

DVD ДАРОМ!



FEDORA 26 + MAGEIA 6

LINUX FORMAT

Главное в мире Linux

Октябрь 2017 № 10 (228)

72 страницы
учебников
и статей!

- » Подкаст с Audacity
- » Восстановим файлы в Ubuntu
- » Автоматизация с Vagrant



Академия кодига:
Решите загадку LXF!

УЧИТЕСЬ С LINUX

Оборудование, проекты, программы и коды,
чтобы стать первым учеником в классе!

- ✓ Open Source в образовании
- ✓ Крутые проекты с Raspberry Pi
- ✓ Как стать отличником по ИТ



ЕСТЬ
**ЭЛЕКТРОННАЯ
ВЕРСИЯ**
ДЛЯ ВАШЕГО
ПЛАНШЕТА!



Провалофон

“ Большинство решений принималось за закрытыми дверями или на закрытых форумах ”

Симон Раффайнер — и его путь разработчика Ubuntu Touch

Безопасность

Простейшие брандмауэры

» Защитите свои устройства самым крошечным аппаратным брандмауэром



Сравнение

Всегда в унисон

» Лучшие инструменты для поддержки актуальности файлов



Открытые телефоны живы!

» Реанимируйте убунтофон, загрузите Lineage OS, и не только!

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

Linux center
www.linuxcenter.ru





iTeleRadio

ИНТЕРАКТИВНОЕ РАДИО&ТЕЛЕВИДЕНИЕ



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: как свободное и открытое ПО помогает вам учиться? Каков ваш любимый инструмент или метод для изучения нового предмета?



Алекс Кокс

Забудьте RTFM: сначала ныряйте головой вниз. Чуть-чуть утоните. Найдите какого-нибудь бедолагу с легкими, наполненными водой, и определите, как он эту воду выкашлял. Нырните снова и повторите процесс, пока метафора утопления себя полностью не изживет.



Нейт Дрейк

Я стал рьяным поклонником мнемонического метода локусов, по которому вы визуализируете объекты, размещая их в знакомых вам местах. Особенно хорошо таким способом запоминать длинные списки информации. При необходимости можно комбинировать это с системой привязки, когда некоторые объекты рифмуются с числами.



Ник Пирс

К сожалению, вынужден признать, что моим любимым инструментом для обучения является DuckDuckGo. 15 лет ответов на вопросы читателей научили меня: дело не в том, что вы знаете, а в том, как задавать правильные ключевые слова (и просивать результаты), чтобы найти искомое.



Лес Паундер

Может, оно прозвучит примитивно, но поиск ответа на свой вопрос — это большое умение. Конечно, можно просто нагуглить, но тогда бывает нужно правильно поставить вопрос. А если ваша проблема связана с конкретной командой Unix, то бесценны map-страницы!



Маянк Шарма

Важная часть процесса обучения — «Застывание». У меня есть «правило получаса»: прежде чем попросить о помощи, я бьюсь над проблемой сам как минимум 30 минут. Это лучше всего работает при обучении программированию, и по-моему, благодаря этому я стал лучшим и более уверенным кодером.



Валентин Синецын

Помню, в детстве мне подарили машинку, в которой можно было покрутить руль — и повернуть колеса, как в настоящей. Довольно скоро колеса перестали поворачиваться, зато я узнал про рулевую рейку. С тех пор методы не слишком изменились: разобрать, посмотреть, попробовать собрать обратно. Свободное ПО этому только помогает. Пойду соберу ядро...



Старые проблемы на новый лад

» Тема создания свободной ОС для смартфонов не нова. Если задуматься, то положение дел очень напоминает состояние двадцатилетней давности на рынке «обычных» операционок. Одна фирма предлагает проприетарную ОС, работающую только на собственном недешевом «железе». Другая разрешает любым авторам смартфонов модернизировать свою систему в ущерб качеству, чем достигает доминирования на рынке. Третья, крупнейший изготовитель аппаратуры, предпринимает уже вторую попытку убедить публику в преимуществах своей ОС, прямо как некогда IBM с OS/2. На этом фоне нередко возникают и столь же быстро угасают проекты создания свободной ОС. Сценарии одинаковы: система «не успевает взлететь» и обрести приложениями, а немногочисленные устройства, на которых она работает, уже снимают с производства.

Вторая проблема связана с самой концепцией смартфона. Функциональность «поговорить голосом» реализована в нем как приложение. Одно из прочих равных. Если какое-то из приложений зависает, устройство перестает выполнять свою самую критичную функцию — принимать звонки. Очень напоминает серверные ОС, 20 лет назад, когда родилась «главная админская заповедь» — выделять отдельный сервер под каждую важную задачу.

Может быть, в архитектуре «смартфонного железа» пора что-то подправить? Сделать ее более стандартной, да и менять пореже.

Кирилл Степанов

Главный редактор

» info@linuxformat.ru

Как с нами связаться

Письма для публикации: letters@linuxformat.ru

Подписка и предыдущие номера: subscribe@linuxformat.ru

Техническая поддержка: answers@linuxformat.ru

Общие вопросы: info@linuxformat.ru

Проблемы с дисками: disks@linuxformat.ru

Вопросы распространения: sales@linuxformat.ru

Сайт: www.linuxformat.ru, группа «ВКонтакте»: vk.com/linuxform

» 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

Зарегистрируйтесь
на сайте
shop.linuxformat.ru



СОМНЕНИЯ

Скачайте бесплатно
архивные PDF-номера
журнала с сайта
linuxformat.ru/archive
и загляните на
wiki.linuxformat.ru



Версия для iPad
и iPhone доступна
в App Store

Выберите вид подписки

PDF-версия
на 6 месяцев

990 ₺

PDF-версия
на 12 месяцев

1800 ₺

Печатная версия
на 6 месяцев

1890 ₺

Печатная версия
на 12 месяцев

3480 ₺

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 2005–2014 гг.
- Порадуются новинкам открытого ПО на DVD-приложении к Linux Format в каждом номере

Способы доставки

- Курьером «ГНУ/Линуксцентра» по Москве и Петербургу
- Курьерской службой СПСР по России
- Почтой по России заказной или простой бандеролью
- Самовывоз из офиса «ГНУ/Линуксцентра» в Санкт-Петербурге
- Через пункты выдачи интернет-магазинов **iml.ru** в 11 городах России: Санкт-Петербург, Москва, Екатеринбург, Калуга, Нижний Новгород, Орел, Ростов-на-Дону, Тверь, Тюмень, Челябинск, Ярославль

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

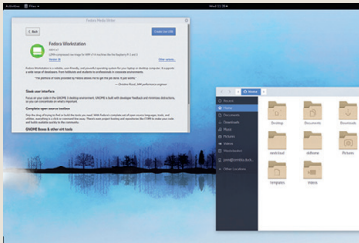
Содержание

«Ибо корень всех зол есть сребролюбие.» 1-е Тимофею 6:10

Обзоры

Fedora 26 Workstation 14

Дистрибутив в голубых тонах заставил автора обзора краснеть — не так просто удумать очередной каламбур про шляпы...



➤ Не программами едиными: Fedora предлагает еще и коллекцию элегантных обоев.

Linux Mint 18.2 16

Шикарный свежий релиз любимца пользователей вполне заслужил престижную награду «Debian среди настольных дистрибутивов».

SwagArch 2017.07 17

Замеряем, далеко ли откатилось яблочко от яблони — точнее, от дистрибутива-прародителя Arch.

Synology DS1817 18

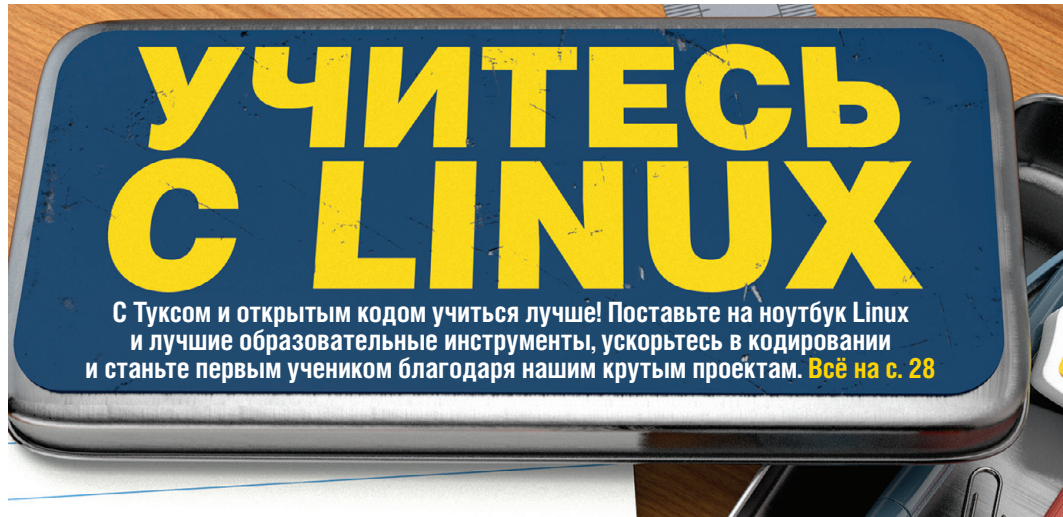
10-гигабитный Ethernet помещает это устройство NAS в ценовую категорию «для малого бизнеса или для людей не бедных» — те, кому такое по карману, получат хранилище с восемью отсеками и супер-скоростную передачу данных.



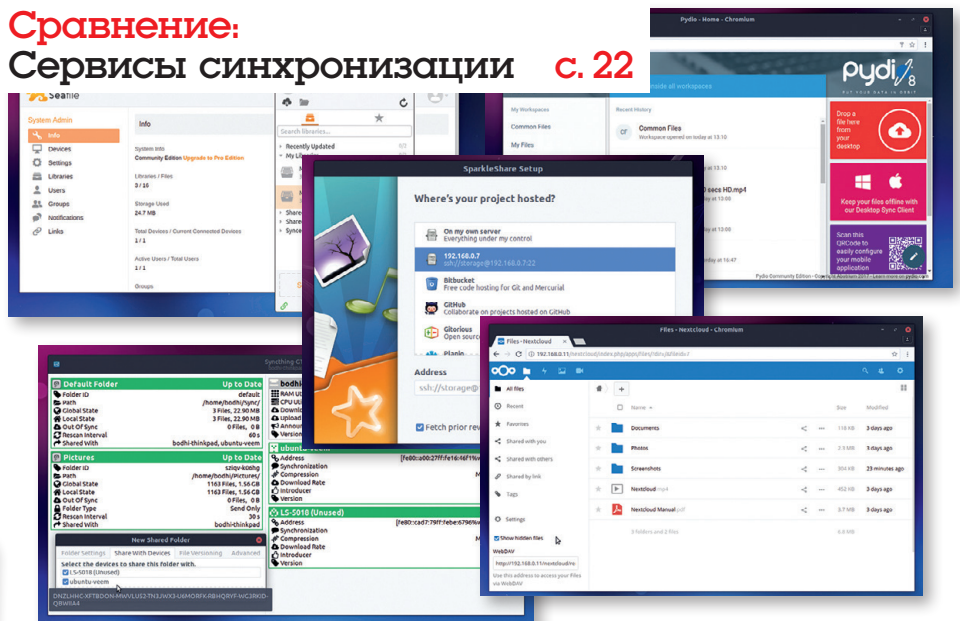
➤ Вот как выглядит солидное хранилище: столько отсеков — это серьезно.

Hollow Knight 19

Герой — насекомое с пустыми глазами — намерен спасти пошатнувшуюся империю, которая держится только былой славой. И это одна из лучших виденных нами игр на открытой платформе.



Сравнение: Сервисы синхронизации с. 22



Интервью

«Какие функции должны войти в следующее обновление OTA, не обсуждалось»

Путь **Симона Раффайнера** — разработчика Ubuntu Touch с. 36

На вашем бесплатном DVD

Побалуйте себя и любимых подпиской на LXF!

Fedora 26
 Передовой дистрибутив с новейшими технологиями:
 » Wayland » Ядро 4.11
 » Gnome 3.24 » GCC 7
 » Python 3.6 + еще куча всего!

Mageia 6.0
 После двух лет разработки — солидное обновление kernels настоящего дружелюбия!

LINUX LIVE-ДИСК: ГОТОВ К РАБОТЕ
 с полных дистрибутива для старта в LINUX

Fedora 26 Gnome 64-битный
Fedora 26 LXQt 32-битный
Mageia 6.0 Xfce 32-битный, **CAINE** 64-битный

» Каждый месяц — только лучшие дистрибутивы

ПЛЮС: HotPicks, код и библиотека с. 98



Доступно в AppStore!



www.linuxformat.ru/subscribe

Пользователям Raspberry Pi



Что слышно про Pi 84

Для Arduino наступило счастье и порядок с торговой маркой, GPIOZero обновился, а Pimoronі взрослеет.

Thunderborg 85

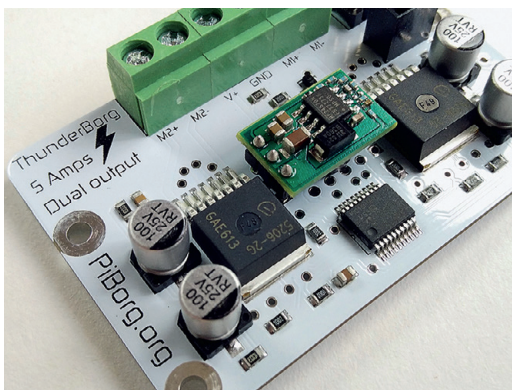
Ну не можем мы удержаться от изготовления гаджетов Skynet — так уж мы запрограммированы.

Сервер печати на Pi Zero 86

Покажем, как с помощью Pi реанимировать старый принтер, сделав его сетевым.

Аппаратный брандмауэр 88

Превратите свой Pi в супер-безопасную точку доступа, да еще и с брандмауэром, чтобы защитить свой ПК от ужасов Интернета.



Ищите в номере

Открытые телефоны 40

Неужели мечту об открытом телефоне похоронили? Ниоим образом! Рассматриваем претендентов.



Академия кодига

Загадка LXF 74

В очередной загадке LXF спрашивается, как всё отсортировать при наличии ограниченной памяти. Ведь мы-то уже и не помним, который нынче день недели...

Технология Serverless 78

Чисто по ненависти к серверам, развернем приложения облачного масштаба всего несколькими строками кода. Из нашей позлащенной небесной колесницы на Amazon Lambda.

```
#!/usr/bin/perl -w

use strict;
use warnings;

my ($in, $out) = @ARGV;

if ( (not defined $in) || (not defined $out) ) {
    die "Need two argument!\n";
}

open(INFILE, '<', $in) or die "$!";
open(OUTFILE, '>', $out) or die "$!";
```

Учебники

Основы терминала

Цепочки команд 46

Сооружаем полезные комбинации из команд, употребляя операторы связывания в последовательность.

Музыка

Multimedia Suite 48

Раскрепощаем ваши таланты и способности к творчеству — займемся музыкальной композицией.

Процессы

Что-то с памятью 52

Изучаем утилиты, которые занимаются виртуальной памятью процессов и еще кое-чем.

Восстановление данных

Scalpel 56

Обойдемся без паники: с таким инструментом удастся спасти все ценные файлы.

Автоматизация

Vagrant для чайников 58

Динамически создаем виртуальные машины и развертываем приложения через Ansible.

Android

Ставим Lineage OS 62

Приобщимся к проекту, который пришел на смену Cyanogenmod.

Аудио

Подкасты с Audacity 66

Объясняем, как создаются идеальные звукозаписи.

Интернет вещей

Теплее... еще теплее... 70

Обеспечиваем в Послушном Доме поддержку комфортной температуры.

Постоянные рубрики

Новости 6

Графика возвращается в процессор, ноутбуки HP худеют, «Базальт СПО» выдала новый релиз, антивирусная фирма нечаянно распространила троянца, Intel разработала самообучаемый процессор, Microsoft поддержала Open Source, вышли жесткие диски на 12 ТБ, а SD-карты выдерживают стресс.

Вести мобильных ОС 20

Грядет полностью свободный смартфон, Raspberry Pi внедряются на автомобили, BlackBerry отказался от физической клавиатуры, а фитнес-браслеты поступили в продажу.

Сравнение 22

При таких замечательных сервисах ни одного файла не будет потеряно: NextCloud, Pydio, Seafile CE, SparkleShare, Synching.

Интервью LXF 36

Симон Раффайнер не жалеет о полученном опыте, но считает методы Canonical малоудачными.

HotPicks 92

Отведайте горяченького! Лучшие в мире новинки свободного ПО: Dead Ascent, Doublespeak, Makehuman, Parlatype, PulseEffects, Qt5ct, Rapid Photo Downloader, Shotcut, Stunt Rally, TiliX, WINE.

Диск Linux Format 98

Содержимое двустороннего DVD этого месяца.

Пропустили номер? ... 100

Да ведь это означает целый потерянный мир — мир виртуальности. Но всё поправимо, если поторопиться.

Школа LXF 103

Изучим влияние перехода на Linux на учебный процесс школы и ВУЗа.

Через месяц 112

Прикинемся супер-шпионами, заведём оборонный комплекс на Pi, защитив всё подряд, и погоняем на корде инструменты мониторинга сети.



В ЭТОМ НОМЕРЕ: Графика вновь в CPU » Самые тонкие » Наши на марше » Вирус от антивируса » Синтетический мозг » Microsoft+Linux = любовь » Накопители большие и малые

БЮДЖЕТНАЯ ГРАФИКА

APU/GPU: вот в чем вопрос!

В конце года AMD выпустит новое поколение процессоров с интегрированным графическим ядром.

Появились результаты тестирования APU архитектуры Raven Ridge, которые AMD представит в декабре. Приводимые WCCFTech оценки Ryzen 5 2500U показывают: по сравнению с нынешними APU архитектуры Bristol Ridge, новый процессор более чем на 50% быстрее в одноядерных приложениях, и примерно на 90% — при многоядерных нагрузках. Его результат — 3625 баллов в одноядерном тесте Geekbench и 9723 балла в многоядерном; для сравнения, результат HP Elitebook 745 G4 с 4-ядерным APU AMD Pro A12-9800B (Bristol Ridge) — 2315 и 5115 баллов соответственно. APU Raven Ridge станут первыми интегрированными устройствами, объединяющими появившуюся в марте

процессорную микроархитектуру AMD Zen и выпущенную летом графическую микроархитектуру Vega. Согласно поступившей в начале года от AMD информации, новые APU обеспечат как минимум 50-процентный рост производительности CPU при снижении потребляемой мощности вдвое. Предполагается, что AMD Ryzen 5 2500U содержит четыре ядра на восемь потоков; процессоры APU Raven Ridge будут устанавливаться в ноутбуки нижнего и среднего классов.

И Zen, и Vega — представители архитектуры с технологией 14 нм, в отличие от своих предшественников с техпроцессом 28 нм. При этом Zen, в форме десктопных CPU Ryzen, уже принята хорошо —

» Согласно тестам, производительность чипсета Ryzen 5 2500U с интегрированной графикой Vega выше, чем показатели Intel Core i7-7660U Kaby Lake с интегрированной графикой Intel Iris Plus 640.

в отличие от Vega: дебютные устройства, оснащенные такими GPU, не обеспечили ожидаемого значительного превышения производительности над соперниками из лагеря Nvidia; кроме того, применение памяти HDM вместо GDDR5 увеличило стоимость видеокарт на базе Vega.



» Рубрику готовил
**АНДРЕЙ
ГОНДАРЕНКОВ**



ПРЕМИУМ-КЛАСС

Восходит призрак золотой

HP обновила свою линейку Spectre.

Второе поколение своих ультрабуков, Spectre 13, HP считает «самыми тонкими сенсорными ноутбуками в мире». В корпусе из алюминиевого сплава этого соперника MacBook заключены процессор Intel Core i5 или Core i7 8-го поколения; до 16 ГБ LPDDR3 RAM и до 1 ТБ PCIe SSD. Время работы от батареи — 11,5 часа (почти на 2 часа больше, чем у Spectre 13 образца 2016 г.), а технология Fast Charge за 30 минут обеспечит 50% заряда. Новый Spectre 13 оснащен дисплеем Ultra HD 4K, полноразмерной клавиатурой edge-to-edge с подсветкой, поддерживает технологию аутентификации Windows Hello

и будет доступен в корпусах Ceramic White и Pale Gold. В UK продажи моделей с Core i5 и Core i7 начнутся в ноябре, за £1399,99 и £1599,99. Обновлен и Spectre x360; у него несколько режимов трансформации и «самое длительное среди 4-ядерных трансформеров время работы от батареи» (обещано до 16,5 часов); он также оснащен дисплеем 13" (вплоть до 4K), CPU Intel 8-го поколения, пером Windows Ink, подключаемым через USB-C, сканером отпечатков пальцев, ИК-камерой Full HD для распознавания лиц; поддерживает Windows Hello. Корпуса — Ash Copper и Pale Rose Gold, цена от £1299,99.

» Энн-Софи Хэдберг [Anne-Sophie Hedberg], глава подразделения Consumer Personal Systems EMEA в HP: «Самые тонкие в мире ноутбуки HP устанавливают новые стандарты автономности, безопасности и дизайна».



ОТЕЧЕСТВЕННЫЕ ДИСТРИБУТИВЫ

Новинки от «Базальт СПО»

«Альт Рабочая станция» и «Engineering» — и для работы, и для развлечений.

Компания «Базальт СПО» представила новый релиз предназначенного для корпоративных рабочих мест и персонального использования дистрибутива K 8.2 ОС Альт Рабочая станция — с рабочей средой KDE 5, широким выбором ПО для работы в Интернете, с документами, сложной графикой и анимацией, для обработки звука и видео, для образования, оснащенный средствами подключения к инфраструктуре и различным сервисам сети подразделения и предприятия.

Рабочая станция интегрирована с сервером Альт Сервер 8 и включена в Единый реестр российского программного обеспечения и баз данных.

Новое в релизе K 8.2:

- » добавлена поддержка мониторов высокого разрешения;
- » исправлена ошибка, препятствующая сохранению сессии в LiveCD;
- » расширен спектр программ, портированных на KF5;

» свежие версии ПО — KDE Plasma 5.10.4, KDE Frameworks 5.37.0, KDE Applications 17.04.3, ядро Linux 4.9.50, Qt 5.7.1, Mesa 17.1.9, Firefox ESR 52.3, WINE 2.16, GIMP 2.8.20, Inkscape 0.92.1, VirtualBox 5.1.24.

А на 14-й конференции разработчиков свободных программ, прошедшей 22–24 сентября в Калуге, был представлен ALT Linux Engineering — неофициальный дистрибутив, предназначенный для знакомства с СПО, которое может использоваться в промышленности. В его составе САПР 2D-черчения (LibreCAD), 3D-моделирования (FreeCAD, OpenSCAD) и проектирования электронных устройств (qucs, KiCAD); системы моделирования изготовления изделий на станках с ЧПУ (CAM) и генераторы g-code (Camotics (бывший OpenSCAM) — симулятор g-code; PyCAM — генератор и симулятор gcode; FlatCAM — генератор и симулятор gcode для изготовления печатных плат; gcodetools — плагин для Inkscape, позволяющий генерировать g-code



» Инженерный дистрибутив ALT Linux Engineering — наш ответ Scientific Linux'у.

из контура; pcb2gcode — генератор gcode из gerb-файла печатной платы); ПО для создания ЧПУ на базе компьютера (LinuxCNC); программы для 3D-печати (для оптимизации 3D-моделей — Meshlab, для нарезки 3D-моделей на слои (слайсеры) — Slic3r, Cura, для отправки gcode на 3D-принтеры — printron); ПО для АСУ ТП (Beremiz, YAPLC, OpenSCADA); а также офисное ПО, поддержка принтеров и сканеров, программы для сканирования изображений. Дистрибутив основан на стартерките LXDE.

ПРОБЛЕМЫ КОНКУРЕНТА

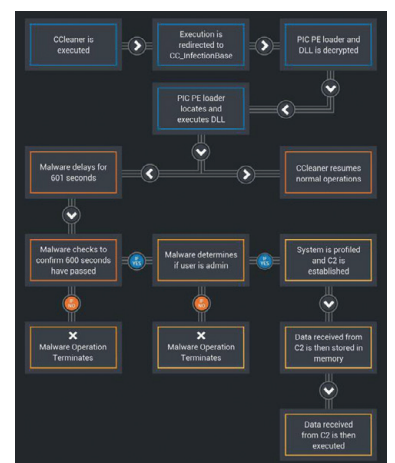
Бонус к оптимизации — бэкдор

Антивирусная фирма Avast непреднамеренно распространила троянца с CCleaner.

Группа Cisco Talos выявила в утилите для оптимизации системы CCleaner вредоносный код. 13 сентября специалисты Cisco Talos обнаружили, что бесплатные версии CCleaner 5.33 и CCleaner Cloud 1.07.3191 содержат «вредоносное дополнение, характерное для Domain Generation Algorithm, а также встроенную функциональность Command and Control» (малварь Floxif). Cisco Talos полагает, что злоумышленник каким-то образом скомпрометировал системы разработки или сборки, использовал цифровой сертификат для подписи вредоносной версии CCleaner 5.33 и подменил ею подлинный вариант; не исключена и помощь со стороны инсайдера. На зараженной машине зловред собирает информацию о системе, запущенных процессах, MAC-адресах сетевых устройств, ID комплектующих, и передает ее на удаленный сервер; он также способен загрузить и запустить дополнительные двоичные файлы.

Avast считает, что зараженные Floxif версии CCleaner успели распространиться на 2,27 млн компьютеров (около 3% пользователей утилиты), однако вышедшие обновления нейтрализовали вредоносное ПО, и ни один зараженный хост второй фазе атаки так и не подвергся. Вице-президент Пол Юнг [Paul Yung] в своем блоге утверждает, что еще 12 сентября компания идентифицировала атаку и приняла соответствующие меры до получения уведомления от Cisco Talos. Атака была ограничена версиями CCleaner и CCleaner Cloud для 32-битных систем, а большинство современных PC всё же работают на 64-битной версии Windows. Для проверки своей системы пользователю необходимо найти в реестре ветку HKEY_LOCAL_MACHINE\SOFTWARE\Piriform\Agomo: наличие в ней элементов MUID и TCID является признаком заражения Floxif. Avast рекомендует пользователям загрузить CCleaner версии 5.34 и выполнить антивирусную проверку

» Представленная Cisco Talos схема действия вредоносной программы Floxif, содержащейся в инфицированной версии CCleaner 5.33.



компьютера. А Cisco Talos рекомендует использовать резервную копию для восстановления PC до состояния на 15 августа, когда была выпущена взломанная версия, или же переустановить всю систему. И это еще один повод перейти на GNU/Linux!

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ

На стыке электроники и биологии

Intel разработала нейроморфный компьютерный чип.

Компания Intel разработала “Loihi” — самообучаемый нейроморфный процессор, имитирующий функционирование человеческого мозга. Названный в честь активного подводного вулкана к юго-востоку от острова Гавайи, Loihi содержит 130 тыс. полупроводниковых «нейронов», связанных с 130 млн «синапсов» (межклеточные контакты между нейронами человеческого мозга). Intel начинает выпуск Loihi уже в октябре, по техпроцессу 14 нм.

Microsoft заявила, что тоже работает над путями создания новых возможностей для альтернативных компьютерных вычислений, включая производство современных чипов и систем, а также разработку программного обеспечения для квантовых компьютеров. Loihi не использует квантовые методы, но преследует аналогичную цель: вместо грубого перебора возможных путей к решению, как делают традиционные чипы, он пытается симитировать параллельную структуру человеческого

мозга, что позволяет гораздо эффективнее найти правильный ответ.

Loihi предназначен для «самообучения» с использованием собственного цифрового массива синтетических нейронов и синапсов; каждое «нейроморфное ядро» содержит «обучающий механизм», который позволяет изменять параметры обучения в соответствии с конкретными потребностями.

«По мере того, как рабочие нагрузки ИИ становятся всё более сложными и разнообразными, они будут испытывать пределы современных доминирующих компьютерных архитектур и ускорять новые прорывные подходы», пишет в своем блоге корпоративный вице-президент и управляющий директор Intel Labs Майкл Мейберри [Michael Mayberry]. Intel считает, что ее реализация Loihi может быть в миллионы раз быстрее, чем другие типичные нейронные сети. «Нейроморфные вычисления предлагают способ обеспечить эксафлопсную производительность в конструкции,



► Loihi — одна из многих новых технологий, претендующих на роль будущего компьютерных вычислений в условиях утраты законом Мура своей актуальности.

имитирующей архитектуру человеческого головного мозга», утверждает Мейберри.

Loihi — не первая реализация «синтетического мозга» на полупроводниках. В представленном в 2016 г. чипе IBM DeepNorth используются 1 млн нейронов и 256 млн синапсов; в 2012 г. исследователи из Google и Стэнфордского университета разработали «мозгоподобную» нейронную сеть для распознавания лиц.

ОТКРЫТОЕ ПО

Времена меняются...

Microsoft присоединилась к Open Source Initiative в качестве Премиум-спонсора.

Мicrosoft сделала очередной шаг в поддержку Open Source, объявив в конце сентября о намерении финансировать деятельность Open Source Initiative (OSI).

OSI, участвующая во многих аспектах движения open source, продвигающая открытое ПО и предоставляющая ему техническую поддержку организация, была основана в 1998 г. — примерно в то же время, когда руководивший тогда Microsoft Стив Балмер [Steve Ballmer] назвал Linux и сообщество Open Source «раковой опухолью». Эта некоммерческая организация наиболее известна своей поддержкой определения Open Source Definition, в котором указаны критерии соответствия лицензии на ПО стандартам Открытого программного обеспечения.

Microsoft, ставшая в 2016 г. платиновым участником Linux Foundation, теперь вошла в состав OSI на правах Премиум-спонсора

(так же как AdblockPlus, Cumulus Networks, Facebook, Google, IBM и др. — всего 15 компаний). Ежегодные корпоративные доходы Microsoft превышают \$250 млн, из которых на деятельность OSI будут выделяться не менее \$20 тыс. Руководитель подразделения Microsoft Open source programs office Джефф Макаффер [Jeff McAffer] считает, что «работа, которую выполняет Open Source Initiative, имеет жизненно важное значение для эволюции и успеха Open Source как первоклассного элемента в индустрии программного обеспечения. Поскольку Microsoft всё более расширяет и углубляет сотрудничество с сообществами Open Source, мы будем рады поддерживать действия Open Source Initiative».

OSI также отмечает уже достаточно длительную историю своего сотрудничества с Microsoft, в частности, релиз Microsoft .NET Framework в 2014 г., реализованную в Windows 10 поддержку Bash/Linux,



► Ряды спонсоров высшей категории некоммерческой организации Open Source Initiative в сентябре пополнила корпорация Microsoft.

расширенную поддержку Linux в Azure cloud, открытие кода Visual Studio Code. Генеральный менеджер и директор совета OSI Патрик Мэссон [Patrick Masson] считает присоединение Microsoft к OSI «значительной вехой для OSI и открытого ПО в более широком смысле. Не может быть лучшего свидетельства зрелости, жизнеспособности и успеха ПО с открытым кодом».

10 и 11 ноября 2017 г.
Набережная реки Ушайки, 12
г. Томск



«ГОРОД IT» — ОДНА ИЗ САМЫХ КРУПНЫХ IT-КОНФЕРЕНЦИЙ ЗА УРАЛОМ

Это масштабное событие соберет около 1000 участников и свыше 60 докладчиков для обсуждения проблем и трендов на рынке и в бизнесе, которые касаются каждого участника, кто использует IT-технологии или сталкивается с цифровыми продуктами в повседневности.

«Город IT» — это:

- Темы, которые интересны и полезны узким специалистам и широкой аудитории.
- Обмен опытом, неформальные встречи и новые знакомства со специалистами из IT.
- Разбор реальных кейсов, советы и рецепты успеха от лидеров отрасли.
- Обсуждение новейших технологий и трендов.
- Приглашенные спикеры из крупнейших компаний.

VII Международный форум Broadband Russia Forum 2017

Инфраструктура для цифровой трансформации

23 и 24 ноября 2017 г.

Отель «Холидей Инн Лесная», Москва, Лесная, 15

ОСНОВНЫЕ ТЕМЫ BROADBAND RUSSIA FORUM 2017:

- Новые требования к сетям ШПД для соответствия требованиям цифровой трансформации.
- Государственная политика развития национальной цифровой инфраструктуры и ликвидации цифрового неравенства.
- Инновации в архитектуре, технологиях и бизнес-моделях развития сетей широкополосного доступа.
- Новые сервисы и контент в сетях широкополосного доступа.
- Стратегии операторов ШПД в применении облачных технологий.
- Умные сети: мониторинг сетевой инфраструктуры и интеллектуальное управление трафиком.
- Виртуализация сетевых функций (NFV) и программно-определяемые сети (SDN): технологии и бизнес.
- Трансформация бизнес-моделей и новые возможности для операторов в цифровой реальности.
- Применение технологий «Интернета вещей» (IoT) в проектах «Умный город» и «Безопасный город».
- Потребности и первые примеры внедрения IoT/промышленного IoT (IIoT) в различных отраслях экономики.
- Новая роль корпоративных данных: Master Data Management, Big Data, превращение данных в цифровой актив, управление данными, монетизация данных.

НАКОПИТЕЛИ БОЛЬШИЕ...

Гиганты для пользовательских систем и NAS

Seagate расширила линейки HDD BarraCuda Pro и IronWolf моделями на 12 ТБ.

Мир огромных жестких дисков пополнился новыми гигантами: компания Seagate представила версии своих HDD BarraCuda Pro и IronWolf на 12 ТБ — самой большой емкости из когда-либо выпущенных в массовом сегменте; и хотя диски уровня предприятия уже давно имеют подобные объемы, для систем обычных пользователей усовершенствованная (и более дорогая) функциональность обеспечения надежности таких приводов избыточна.

Параметры BarraCuda Pro 12 ТБ: форм-фактор 3,5", частота вращения 7200 об./мин., кэш 256 МБ, гарантия 5 лет. На рисунке приведены спецификации новой модели, а также дисков той же серии на 10 и 8 ТБ. Обращает на себя внимание более высокая пропускная способность (260 МБ/с) модели 12 ТБ: дисковая подсистема с новым BarraCuda Pro и кэширующим SSD

Intel Optane обеспечит очень достойную производительность.

Накопитель IronWolf 12 ТБ предназначен для использования в сетевых хранилищах (NAS). В комплект входит программное обеспечение для повышения надежности, в т.ч. инструмент наблюдения за состоянием жестких дисков IronWolf Health Management (IHM) и утилита Error recovery control, а также датчики вращательной вибрации для повышения производительности в многодисковых системах.

По информации PR-агентства Seagate, цена модели IronWolf на 12 ТБ (нагрузочная способность 180 ТБ/год) составит \$470, а IronWolf Pro (300 ТБ/год) — \$540. Рекомендуемая розничная цена производителя для BarraCuda Pro — \$530 (при этом цена модели на 10 ТБ составляет \$400, а enterprise-диск Seagate 12 ТБ доступен на Amazon за \$515).

... И НАКОПИТЕЛИ МАЛЕНЬКИЕ

Приказано выжить

Новые карты SanDisk способны работать в широком диапазоне температур.

Компания SanDisk начала выпуск предназначенных для использования в промышленных устройствах и в автомобильной электронике карт SD и microSD; новинки способны выдерживать стрессовые нагрузки, связанные с широким перепадом температур. Модели обеих линеек, и SanDisk Industrial, и SanDisk Automotive, обеспечивают параметры Class 10 (скорость чтения до 80 МБ/с, скорость записи 50 МБ/с), их диапазон рабочих температур от -40°C до 85°C для карт Automotive SD и Industrial XI SD, и от -25°C до 85°C для карт Industrial SD и Industrial microSD.

Карты Automotive SD соответствуют стандарту AEC-Q100 Grade 3. Карты Industrial обладают номинальным ресурсом до 192 TBW (TeraBytes Written). Диапазон емкостей — от 8 Гб до 128 Гб. Накопители

поддерживают функцию Health status monitor, которая позволяет отслеживать сбои и износ, а технология Enhanced data retention в целях обеспечения надежности автоматически обновляет данные. Предусмотренная в обеих сериях карт функция Power Protection предотвращает повреждение данных в случае прерывания поступления питания во время записи данных на карту. Поддержка функции Host Lock позволяет предотвратить несанкционированное использование данных.

Ознакомительные образцы SanDisk Industrial и SD SanDisk Automotive уже поступили OEM-производителям по всему миру, в октябре к ним добавятся SD-карты SanDisk Industrial Extended Temperature. Серийные поставки запланированы уже на эту осень. **LXF**

Новости короткой строкой

» В Gnome 3.26 реализованы улучшенный поиск, отключение системного лотка, новое оформление конфигуратора, поддержка синхронизации закладок, паролей и истории посещений Eiphanу через сервис Firefox Sync. Источник: mail.gnome.org

» Из очередного LTS-релиза ядра Linux 4.14 поставляемые в каталоге firmware прошивки перемещены в репозиторий linux-firmware, служащий для формирования одноименного пакета. Источник: lkm1.org

» С января 2018 г. браузер Google Chrome начнет по умолчанию блокировать автовоспроизведение видео на открываемых web-страницах. Источник: www.pcworld.com

» Яндекс представил аналог Android Auto и CarPlay — автомобильную интеллектуальную платформу «Яндекс.Авто», которая уже в октябре появится в Toyota Camry Exclusive и Toyota Rav4. Источник: rg.ru

» Intel прекращает разработку своей автономной гарнитуры VR Project Alloy, несовместимой с питанием от ПК. Источник: www.pcworld.com

» Минобороны РФ планирует к ноябрю 2019 г. выпустить новый релиз своей ОС реального времени «Багет 4.0» на что готово потратить 420,7 млн руб. Источник: www.cnews.ru

» В начале октября в РФ стартовали официальные продажи Android-смартфона BlackBerry KEYone с адаптированной под российский рынок физической 35-кнопочной клавиатурой. Источник: www.cnews.ru

» Google обеспечит поддержку LTS-ядра Linux 4.4 на 6 лет (вместо двух), упростив сопровождение прошивок для потребительских устройств на базе Linux и увеличив цикл их жизни. Источник: www.opennet.ru

» Ubuntu 17.10 «Artful Aardvark» вышел с ядром Linux 4.13, рабочим столом Gnome Shell, компонентами Gnome 3.26; и Wayland по умолчанию. Источник: lists.ubuntu.com

» Алиса, голосовой помощник Яндекса, разговаривает голосом актрисы Татьяны Шитовой (дублировавшей Скарлетт Йоханссон) и работает с поиском, картами, музыкой и погодой. Источник: www.vedomosti.ru

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 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 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!

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 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 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!

Новинки программного и аппаратного обеспечения в описании наших экспертов



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

Настройка Maui

Прошлая колонка была посвящена особенностям дистрибутива Maui, которые можно считать достоинствами — и ни слова не говорилось о его недостатках. Впрочем, по мне, недостаток у Maui один: перегруженность штатными приложениями и их довольно субъективный выбор. Трудно себе представить применителя, не занимающегося профессионально графикой, которому нужны были бы одновременно и *GIMP*, и *Krita*. А про игры и говорить нечего: для записного геймера их мало, для простых людей — слишком много.

Напрашивается естественное желание иметь некую облегченную версию, освобожденную от всего, что определяется вкусом, привычками и предпочтениями. Но такой, увы, нет. Что же, еще Ильф и Петров объяснили нам, чьих рук делом должно быть дело спасения утопающих. Тем более что такую Lite-версию Maui легко изготовить самим. Или — воспользоваться той, которую уже сделал автор этих строк. Из нее удалены все игры, почти все мультимедиа и графика (кроме абсолютно необходимой), и даже... *LibreOffice*. В результате чего получился ISO-образ на 1,4 Гб, занимающий после установки 4,8 Гб (оригинал — 2,0 и 6,7 Гб, соответственно). Чисткой системы дело не ограничилось: для обновления ядра был задействован стек HWE, выполнены кое-какие настройки KDE и его штатных приложений. Полный отчет о проделанной работе см. здесь: <http://alv.me/category/manuals/maui-i-kde/>. А зачем это делалось — расскажем в следующей колонке. alv@posix.ru

Сегодня мы рассматриваем:

Fedora 26 Workstation 14

Не прошло и восьми месяцев, как мы обзавелись новой шляпкой! Самое заметное улучшение — графический инструмент для ручного разбиения дисков, да и других тоже немало.

Linux Mint 18.2 16

Дистрибутив долгое время был популярен у новичков в Linux, а вот последняя итерация оснащена инструментом специально для продвинутых пользователей: это инструмент командной строки *mintupdate-tool*.

SwagArch 2017.07 17

Arch, но не совсем — тех же щей, да поживе влей... Наверное, надо бы

выждать, отдавая этому дистрибутиву время на достижение зрелости, но разработчик-одиночка — это, вообще говоря, чревато определенными рисками.

Synology DS1817 18

Да хранит нас это NAS. Всего несколькими щелчками создаются почтовый сервер, сервер Radius, сервер VP, web-сервер *Apache*, платформа *Drupal* и не только.

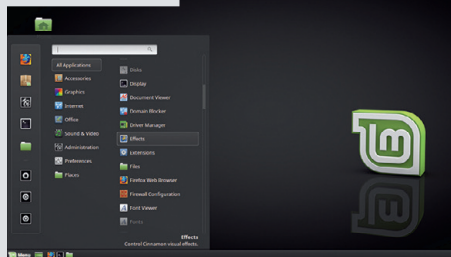
Hollow Knight 19

Рыцарь-насекомое? Это нечто экзотическое, такого мы еще не видели. Но, как и полагается рыцарям, он исправно сражается со злом, скрывающимся в недрах обветшавшей империи.



➤ Это не просто коробка для ваших терабайтников: всё управляется операционной системой.

Linux Mint 18.2



➤ Мрачную черную тему легко заменить на что-нибудь повеселее по вашему выбору.

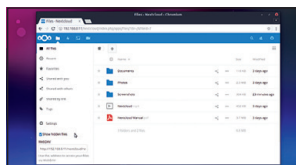
Hollow Knight



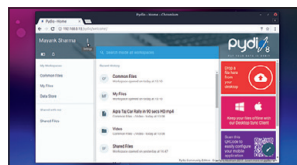
➤ Как и на планете Земля, насекомые здесь — самая многочисленная биопопуляция.

Сравнение: Сервисы синхронизации с. 22

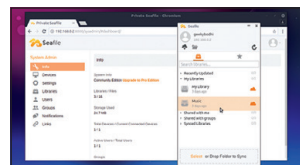
NextCloud



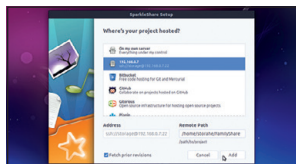
Pydio



Seafile CE



SparkleShare



Syncthing



Цифровая техника порождает огромное количество данных: мы направо и налево щелкаем фотокамерой, скачиваем музыку, снимаем клипы... но ведь всё это надо где-то хранить? И иметь к нему доступ?

Fedora 26

Джонни Бидвелл пасует перед необходимостью выдумать очередную байку про шляпу, достойную этого восхитительного дистрибутива — теперь в версии 26.

Вкратце

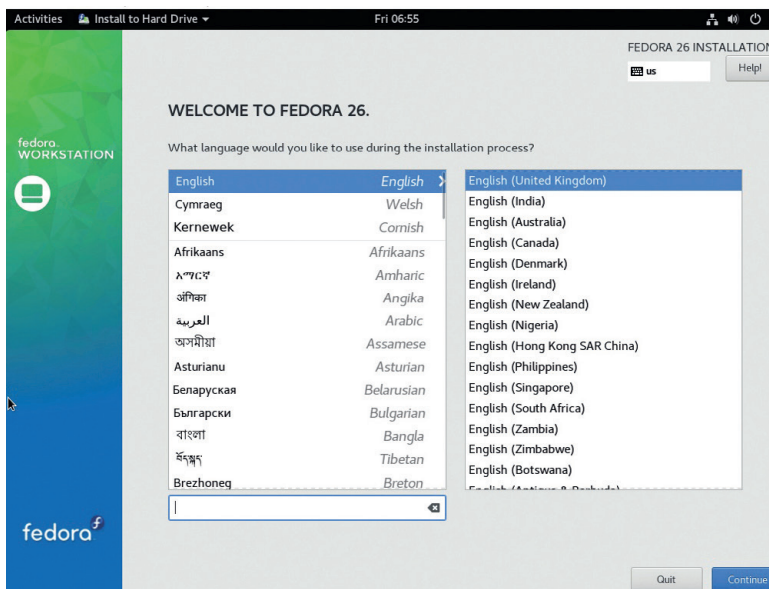
» Последняя версия дистрибутива сообщества, спонсируемого Red Hat (где много зарабатывают, продавая Red Hat Enterprise Linux), набитая новым ПО и функциями и доступная в нескольких редакциях. См. также Debian, Arch Linux, KaOS, CentOS и RHEL

Хорошие вещи стоит подождать, и наконец-то, после нескольких задержек, наше ожидание Fedora 26 завершилось. Прошло около семи с половиной месяцев после выхода отличной Fedora 25 — достаточно для множества захватывающих разработок в мире Linux.. Многие из них вошли в эту версию: Kernel 4.11, *systemd* 233, *GCC* 7.1, *OpenSSL* 1.1, *Gnome* 3.24, *LibreOffice* 5.3... мы могли бы продолжать, но перечни и номера не способствуют качеству обзора.

У Fedora репутация классного дистрибутива, но он подходит для более опытных, а не для начинающих пользователей. Обе оценки справедливы. Новые версии Fedora — отличное место для поиска новых функций и ПО без обращения к PPA (Personal Package Archives), ручной компиляции или рассылки плавающего дистрибутива.

И хотя мы все любим блестящие новые вещи, будьте готовы сделать некоторые настройки сами, и помните: всегда есть шанс, что всё сломается, а исправление может оказаться и того сложнее. А может, и нет; и если вы знакомы с командной строкой и умеете публиковать полезные запросы для поддержки, то это отличная версия, позволяющая повысить уровень Linux-фу.

Так вышло, что мы пропустили обзор Fedora 25. Но это позволило нам протестировать путь обновления от Fedora 24 на нашей обзорной машине, и всё прошло быстро и гладко. Обновление можно выполнить прямо из *Gnome Software* или через командную строку, используя либо плагин *system-*



» Выберите язык, любой язык. Предпочтительно тот, который понимаете.

upgrade, либо опцию *distrosync* нового диспетчера пакетов Fedora, *DNF*. Главным образом мы использовали *system-upgrade*, потому что не знали о родном *distrosync*, и обнаружили, что процесс очень простой.

Изобилие пакетов

Пакеты скачаны, а затем система загружается до «голового» *Systemd*, где и происходит обновление. После коротких секунд задержки система послушно перезагрузилась в новую версию. Запуск *dnf autoremove* удалил несколько оставшихся пакетов и, загадочным образом, заодно

и прекрасную коллекцию обоев (см. снимок экрана внизу справа). Последнюю мы переустановили, поскольку у однотонных рабочих столов [Ред.: — Даже черных?] жизнь слишком коротка.

Мы также сделали чистую установку на Dell XPS13 9360, что снабдило нас улучшениями установщика *Anacanda*. Наиболее заметное из них — новый графический интерфейс *Blivet* для продвинутого разбиения дисков. В Fedora уже давно предусмотрена опция ручного разбиения дисков, но люди, желающие иметь или уже имеющие экзотические настройки (например, LVM поверх RAID), как правило, предпочитают настраивать всё заранее. С *Blivet* эти пользователи могут указать свои требования, вместо того, чтобы догадываться об утилите для ручного разбиения дисков, которая придумывает собственную раскладку, получив информацию о точках монтирования и свойствах.

Всё это, конечно, впечатляет, но большинство людей удовольствуются автоматической утилитой разбиения дисков, которая предлагает выбрать нужный диск или изменить размер существующих разделов, чтобы высвободить место для Fedora. Системным администраторам будут знакомы файлы кикстарт Fedora для автоматизации установок. Они улучшены и предлагают более тонкий контроль, и теперь научены делать снимки предыдущей установки.

Разумная упаковка

Fedora 26 также включает новый менеджер пакетов *DNF* (Dandified Yum) 2.0, который по-прежнему соблюдает синтаксис *yum*. Команда *yum* на самом деле ссылается на *DNF*, но работает точно так же, как и раньше, и если вам неохота изучать новый синтаксис, то и не трудитесь.

DNF умнее и быстрее своего предшественника. Он поддерживает дельта-пакеты (DRPMS), поэтому необходимо загружать только те части пакетов, которые были изменены. Очевидно, что с одними пакетами это работает лучше, с другими хуже, но несколько обновлений, которые мы выполнили, позволили добиться экономии полосы пропускания от 10 до 40 процентов. Пользователи дистрибутивов на базе Debian будут

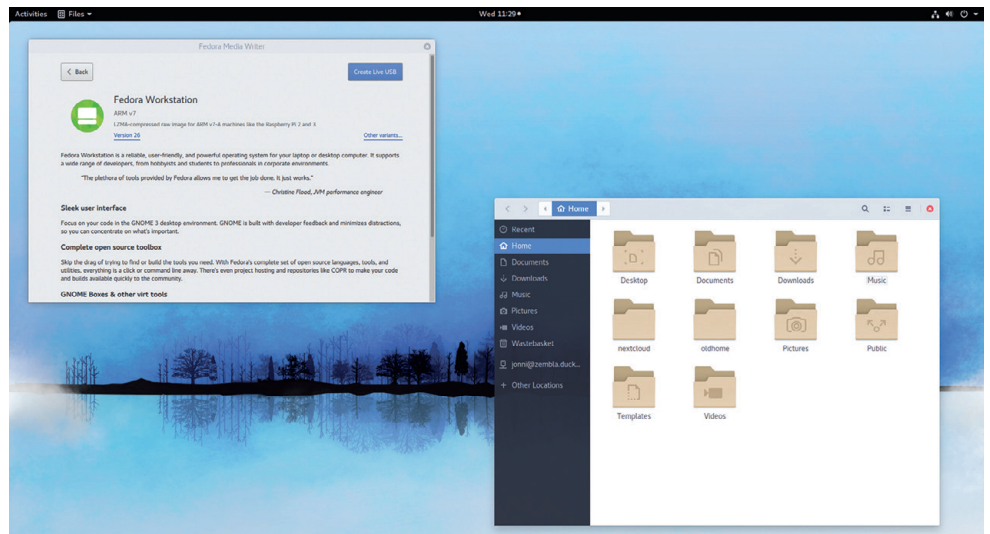
потрясены, услышав, что для обновления системы есть только одна команда — *dnf upgrade*, а не привычное обновление *apt-get update && apt-get*. *DNF* потихоньку обновляет свои списки пакетов в фоновом режиме (по умолчанию — каждые три часа), экономя драгоценные секунды.

Команда *dnf history* особенно полезна для изучения прошлых транзакций пакета. Их можно детально исследовать с помощью *dnf history info*, которая позволяет легко убрать любой беспорядок, созданный теми штуками, которые вы установили ранее, но которые больше не требуются. Также существует команда *dnf update info* для отображения обновлений по категориям, таким как security, bugfix или enhancement.

После установки мы столкнулись с тревожным и повторяющимся сообщением от *Automatic Bug Reporting Tool (ABRT)*, что произошел сбой ядра. Дело явно не в этом, потому что несмотря на сообщения, всё продолжало работать. Системное обновление вроде бы устранило проблему, поэтому мы собираемся приписать ее какой-то неясной ошибке.

Гномолюбивые пользователи Ubuntu 17.04 уже знакомы с Gnome 3.24. Пользователи, пришедшие из Fedora 25, оценят его функцию *Night Light*, для уменьшения синего света, вызывающего напряжение глаз, в ночные часы. Есть еще ряд приятных дополнений, в том числе встроенная информация о погоде, в поле уведомлений [Ред.: — Если выйти на улицу], а также улучшения средств управления мультимедиа. Поддерживается несколько учетных записей онлайн, включая Google, Facebook и Microsoft. Мы обрадовались, что поддержка *ownCloud* переименовывается в *Nextcloud*, но печально, что эта интеграция не работала с нашим экземпляром *Nextcloud 12*, оставив нас с иронически названным, но функциональным настольным клиентом *ownCloud*, чтобы синхронизировать всё для нас.

Помимо стандартной версии Workstation, доступно несколько других редакций («лаборатории» или «спины», на языке Red Hat). Некоторые из них, например, Plasma и Cinnamon, а также Design Lab и Security Lab, существуют не первый день. Эта итерация также получила версию *LXQt*, которой будут рады пользователи более старых машин или просто люди, презирующие раздувание рабочего стола: она обеспечивает рабочий стол на *Qt5* с темой Breeze. Приятно встретить лабораторию Python Classroom, это часть инициативы *Fedora Loves Python* (<https://fedoralovespython.org>). Лаборатория включает несколько версий Python плюс инструмент *tox* для



➤ Тема Arc украшает Gnome, а Media Writer от Fedora помогает создавать в Fedora больше медиа.

переключения между ними. Предусматривается также стек Scientific Python, который включает *Numpy*, *Scipy*, *matplotlib*, оболочку IPython и Jupyter Notebook для web-интерактивных рабочих листов.

Всегда есть место для Pi

Еще одна редакция, заслуживающая упоминания — это версия для Pi. Читатели могут помнить, что когда выпустили Pi, Raspbian OS не было, и рекомендованным вариантом был Pi Remix от Fedora 14 (www.youtube.com/watch?v=RbWE6qF7pIM). С тех пор ситуация изменилась, и исходный Pi больше не поддерживается, но это новое издание для Pi 2 и 3 предлагает множество настольных компьютеров, минимальную версию для использования в проектах IoT и самоделках, да еще и Sugar on a Stick (SoaS) — версию для детей.

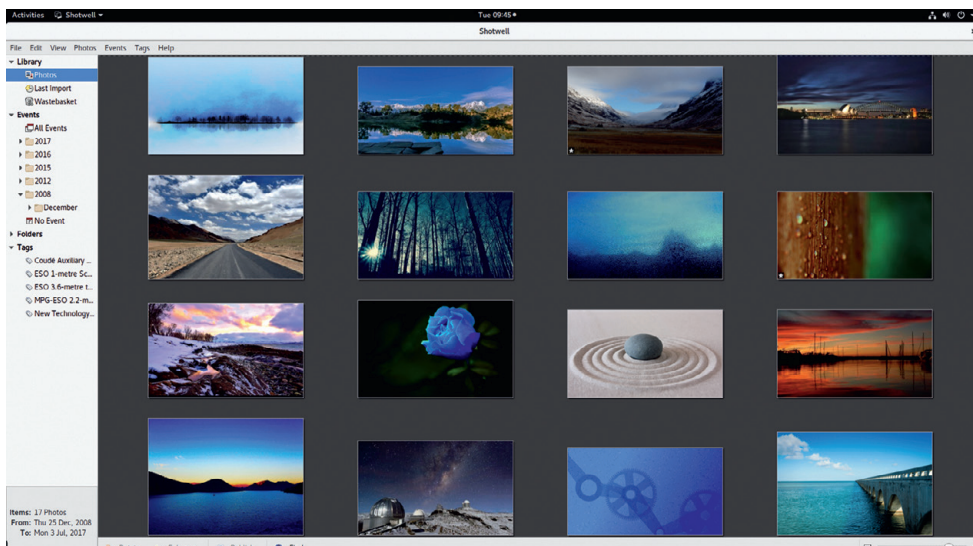
Если вы пришли из дистрибутива на базе Ubuntu, имейте в виду, что идеология Fedora выступает против всех проприетарных и несвободных элементов. Поэтому

установка таких вещей, как *Steam*, *Spotify*, *Flash* [Ред.: — Почему?], драйвер Nvidia и всё, что имеее, потребует больших усилий, чем проставление где-то галочки.

Как бы то ни было, патенты, ограничивающие MP3 в США, истекли в ноябре, и кодирование и декодирование этого почтенного формата теперь работает по умолчанию, несмотря на наличие и других, технических превосходящих форматов. Сторонние репозитории, такие как RPMfusion и negativo.org, позволяют устанавливать проприетарные вставки; а пакет Nvidia, предлагаемый negativo.org, теперь поддерживает ноутбуки с гибридными (Optimus) вариантами графики.

Благодаря новому расширению GLVND и улучшению Prime layer в Mesa, использование в Fedora драйвера Nvidia теперь должно приносить меньше головной боли. Также стоит отметить, что Spotify теперь доступен как Flatpak, поэтому его можно установить только с помощью `flatpak remote-add` и `flatpak install` (<https://blogs.gnome.org/alex/2016/10/26/distributing-spotify-as-a-flatpak/>). **LXF**

➤ Помимо стандартных фонов Gnome, Fedora предлагает на выбор изображения для обоев рабочего стола.



LINUX
FORMAT
Вердикт

Fedora 26 Workstation

Разработчик: Red Hat
 Сайт: www.fedoraproject.org
 Лицензия: Разные FOSS

Функциональность	9/10
Производительность	8/10
Удобство в работе	8/10
Документация	9/10

➤ Отличная версия, хотя и промежуточная. Идеальна для пользователей среднего уровня; более продвинутые могут предпочесть свободу Arch.

Рейтинг
8/10

Mint 18.2 Cinnamon

Восхищенный богатствами новой версии Linux Mint, **Шшанк Шарма** решает дать ему престижную награду «Debian среди настольных дистрибутивов»...

Вкратце

» Дистрибутив накопил историю создания грамотно сделанных версий. Последняя версия LTS будет поддерживаться до 2021 г. и содержит ряд правок и вариантов функций. Его родитель Ubuntu — хороший выбор для пользователей настольных ПК, но, судя по последним событиям, отнюдь не такой здравомыслящий, как Mint. Если вам нужна система управления пакетами RPM, попробуйте Mageia.

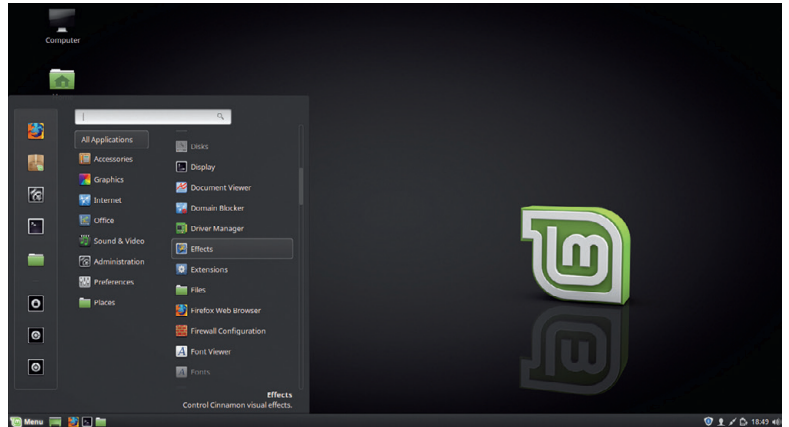
Производные дистрибутивы Linux часто подвержены влиянию того, что происходит с их «родителями». Задержка дистрибутива-источника может сбить расписание потомка. Кроме того, ошибки и основные неудачи родительского дистрибутива могут иногда попасть в продукт ниже по течению. Но Linux Mint, несмотря на то, что он был основан на Ubuntu (а этот дистрибутив Linux вызывает самые полярные мнения), сумел обрести независимость, и на него не повлияли недавние потрясения в лагере Ubuntu.

Глядя на Mint, его расценивают не как клон Ubuntu, а скорее как дистрибутив с уникальной идентичностью: его отличительная черта — долгая история стабильных и симпатичных выпусков.

Приправы

Версия 18.2, названная Sonya, основывается на наборе функций других выпусков 18.x. Теперь можно обновить внешний вид дистрибутива, включив в него различные дополнения, такие как темы, десклеты и апплеты. Исследуйте все эти примочки [addons] с обновленного сайта Spices (<http://cinnamonspices.linuxmint.com>). Если идея их ручной загрузки вам не нравится, используйте соответствующие установщики для каждой из них изнутри System Settings.

Как и многие его аналоги, Mint производит ветки с множеством различных рабочих сред. Однако, чтобы облегчить процесс перемещения своих пользователей между средами, дистрибутив работает над т.н. X-приложениями. Это набор основных приложений, которые начинались как ответвления популярных приложений Gnome,



» Дистрибутив закрепил за собой звание одного из лучших дистрибутивов, дружелюбных к новичкам. Теперь он стал учитывать и опытных пользователей.

но в настоящее время доросли до обеспечения унифицированной работы независимо от рабочего стола, выбранного вами для запуска. Все популярные X-приложения, такие как текстовый редактор (*Xed*), просмотрщик изображений (*Xviewer*), медиа-плеер (*Xplayer*), в этой версии получили ряд изменений и обновлений. *Xreader*, просмотрщик документов, теперь поддерживает сенсорные устройства, жесты прокрутки и щипка.

Дистрибутив долгое время пользовался популярностью у новичков в Linux, но последняя итерация оснащена инструментом, разработанным специально для продвинутых пользователей. *Mintupdate-tool* понравится тем, кто для настройки обновлений предпочитает работать с CLI. Помимо обычного перечисления и установки обновлений, этот инструмент пригоден для написания скриптов или задач планировщика, автоматизируя процесс установки обновлений. Запустите команду `mintupdate-tool --help`, чтобы ознакомиться с ним.

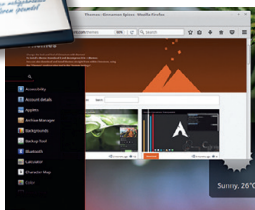
Говоря об обновлениях — Mint имеет несколько иной подход к ним по сравнению с аналогами. Можно выбирать между тремя моделями обновлений, в зависимости от степени знакомства с Linux и уверенности в том, что вы сами справитесь, если обновление нарушит работу системы. Новым пользователям следует придерживаться опции «Просто держать мой компьютер в порядке», а более продвинутые пользователи могут изучить другие варианты. Сознавая, что пользователям может понадобиться помощь для понимания, как система присваивает уровни доступным

обновлениям, эти обновления снабжают обильной и подробной документацией.

В последние годы резкий скачок в дистрибутивах с плавающим релизом убедил автора этого обзора в том, что экосистема Linux медленно, но верно, удаляется от дистрибутивов с фиксированным расписанием выпусков, таких как Ubuntu, Fedora, Mint и другие. Тем не менее, звездные и многофункциональные выпуски, наподобие Linux Mint 18.2, служат напоминанием о том, чего может достичь отдельная команда разработчиков, располагая временем. Это замечательный дистрибутив: он предлагает верное сочетание улучшений по удобству и настраиваемости, чтобы понравиться и новичкам, и опытным пользователям. Не сделавший ни одного неуместного шага, Mint 18.2 является одним из лучших дистрибутивов 2017 г. **LXF**

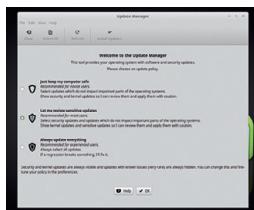


Свойства навскидку



Прямые настройки

Разнообразие тем, десклетов и апплетов поможет вам изменить внешний вид дистрибутива.



LTS-выпуск

Mint 18.2 будет получать обновления до 2021 г. — идеально для любителей постоянства.

LINUX FORMAT Вердикт

Linux Mint 18.2 Cinnamon

Разработчик: Linux Mint
Сайт: www.linuxmint.com
Лицензия: Разные

Функциональность	10/10
Производительность	10/10
Удобство в работе	9/10
Документация	9/10

» Прочно закрепил за собой место одного из лучших дистрибутивов, дружелюбных к новичкам.

Рейтинг 9/10

DiskStation DS1817

Synology называет эту серию устройств NAS бюджетной, но **Марк Пикаванс** решил, что порты 10 GbE и отличный интерфейс выводят их из этой категории.

Спецификация

- » **Процессор** AM4Annapurna Labs Alpine AL-314 (четырёхъядерный, 1,7 ГГц)
- » **Память** 4 ГБ DDR3L, расширяемая до 8 ГБ
- » **Хранилище** Диск SATA 8×3,5" или 2,5"
- » **Порты LAN** 2×1 GbE (RJ-45), 2×10 GbE (RJ-45)
- » **Дополнительные порты** 2×USB 3.0, 2×eSATA
- » **Расширение** через eSATA (два DX517) 10 дополнительных дисков
- » **Габариты** 157×340×233 мм
- » **Вес** 5,31 кг (без дисков)

Снаружи DS1817 практически идентичен DS1815 — устройству, которое это предложение заменяет. Оба вмещают до восьми 3,5- или 2,5-дюймовых приводов SATA, в совокупности потенциально давая 80 ТБ с современными функциями привода. Со внешними блоками расширения (DX517) можно добавить еще 10 дисков и увеличить общую емкость до 180 ТБ.

Чтобы все эти диски работали согласованно, Synology использовала процессор Annapurna Labs Alpine AL-314 — четырехъядерный SoC Cortex-A15 с тактовой частотой 1,7 ГГц. Система поставляется с 4 ГБ памяти DDR3L в одном разьеме, а легко доступный второй сокет позволяет увеличить ее до 8 ГБ. Подключение к сети превосходное — имеются двойные порты LAN, 1 GbE и 10 GbE. Также есть два порта eSATA для подключения блоков внешних накопителей и пара портов USB 3.0 Type-A для подключения переносного накопителя или периферийного устройства USB (принтера), которое вы, возможно, захотите использовать для общего доступа в сети. Будучи моделью серии Value, DS1817 предназначен для продвинутого домашнего пользователя или небольшой рабочей группы. Традиционно эти машины не подлежат обновлению и не имеют высококлассных возможностей. DS1817 бросает вызов всем этим критериям, являясь первым сервером серии Value, в которой Synology одарила пользователей разъемами для памяти с произвольным доступом и сетью 10 GbE.

При одном подключенном клиенте на 1 GbE наблюдаются чтение и запись



» DiskStation может вместить до восьми дисков. Прекрасное зрелище...

на 110 МБ/с. Один клиент с каналом с двумя гигабитными портами, объединенными на обоих концах, повышает производительность до 200 МБ/с. Если верить заявлению Synology, DS1817 способен достигать скорости чтения 1577 МБ/с, а записи — 739 МБ/с.

Хотя для одного подключенного клиента на 10 GbE можно видеть производительность более 350 МБ/с, DS1817 был разработан для совместного использования пропускной способности сети, а не для фокусировки на одном компьютере. В функции обслуживания файлов DS1817 может обслуживать 16 или более активных пользователей при использовании исходящей линии 10 GbE к коммутатору, который соединяет клиентов 1 GbE.

Однако главным для многих людей является расширяемая ОС DSM 6.1. Эта операционная система совместима с линейкой NAS Synology, отличаясь лишь точным выбором приложений, поддерживаемых каждым устройством на основе его аппаратной спецификации. DS1817 умеет работать

с целым рядом приложений и поддерживает широко используемые среды виртуализации, такие, как VMware или Citrix, и сертифицирован на совместимость с Microsoft Hyper-V.

Создание почтового сервера, сервера Radius, сервера VP, web-сервера Apache, платформы Drupal или множества других достигается всего за несколько щелчков. Другие бренды пытались подражать DSM на своих платформах, но ОС Synology выделяется лучшей поддержкой сторонних приложений и отличной базовой функциональностью.

Основанная на ARM, она и не собирается конкурировать с серверами на базе Xeon, но для тех, кто хочет иметь надежный файловый сервер с широкими возможностями или для удовлетворения конкретной потребности, здесь много чего можно рекомендовать. **LXF**



» О сетевых подключениях позаботились хорошо.

LINUX FORMAT Вердикт

DiskStation DS1817

Разработчик: Synology
Сайт: www.synology.com
Цена: £ 899

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

» Устройство Synology предлагает вескую альтернативу желающим иметь высокопроизводительное внешнее хранилище.

Рейтинг 9/10

Hollow Knight

Распадающаяся империя догорает в лучах былой славы, стоившей трудов. **Том Маркс** превратился в смышленного жучка и сбежал в огромный мир фэнтези.

Спецификации

МИНИМАЛЬНАЯ

» **ОС** Ubuntu 16.04 LTS 64-бит

» **Процессор** Intel Core 2 Duo

» **Память** 4 ГБ

» **GPU** Nvidia GeForce 9800GTX, 1 ГБ Vram

» **На диске** 9 ГБ

РЕКОМЕНДУЕМАЯ

» **Процессор** Intel Core i5

» **Память** 8 ГБ

» **GPU** Nvidia GeForce GTX 560

Перед нами экшн-платформер 2D в стиле Metroidvania, а значит, здесь опять предстоит исследовать карту, охотиться за секретами и искать способы попасть на недоступные территории. Что его отличает, так это обширность и насыщенность подземного царства Hollownest. Уже одна структура этого мира выводит игру в лидеры своего жанра.

Мир, в который вы попадаете, давно пережил дни своего расцвета — умирающее королевство, где общество расколото, а его герои пали. В нем нет ничего, кроме пустынного и темного пейзажа на поверхности и извилистых катакомб внизу, откуда веет смертью. Вы управляете бесстрастным рыцарем, чья цель сначала не вполне ясна, и погружаетесь в эту землю, чтобы бороться со злом, которое она таит.

Вот только на самом деле вы симпатичный маленький жучок, который сражается и заводит дружбу с другими симпатичными жучками. Хотя предполагается, что *Hollow Knight* погружает вас в фэнтезийный мир средневековых королевств, тут во всем царит жукообразие. Груз опустошенности и утраты довлеет в истории Hollownest, но благодаря столь милой графике, атмосфера здесь никогда не бывает давящей или удручающей. Самые тяжелые темы подаются легко, без впадения в пародию.

Неигровые персонажи, с которыми вы встречаетесь — забавные и мультяшные, но тоже ощущаются как часть этого мира. Особенно полюбились нам пухлый лавочник, который вечно хихикает, любезный бродяга-фехтовальщик, потерявший память, и крошечный, но шумный Зот [Zote] Могучий. Они также помогают



» При такой роскошной графике, исследовать этот мир — сущее удовольствие.

собрать воедино сюжет *Hollow Knight*, что довольно непросто.

Тактика боя тут незамысловатая: в основном вы снуете вокруг врагов, пытаетесь поразить их с помощью «гвоздя» (вашего меча), одновременно уклоняясь от атак. Удары ощущаются весомыми и реальными, и каждый раз отбрасывают вас назад. Когда вы наносите удар противнику, это тоже чувствуется. Но битвы по ходу игры в *Hollow Knight* нас особо не удивили. Вы всё время сражаетесь одним и тем же оружием и, по сути, одним и тем же способом.

Заклинания

Амулеты наделяют игрока особыми способностями, но число пазов-слотов, куда вы можете их вставить, ограничено. С помощью амулетов можно проделывать всякие простые вещи — например, увеличивать количество Душ, получаемых, когда вы наносите удары врагам, или улучшать свою атаку. Чем сильнее амулет, тем большим количеством пазов он располагает.

Чтобы биться с проблемными боссами, часто приходится искать новые комбинации заклинаний, которые смогут дать вам преимущество. Для каких-то нужен больший радиус, для некоторых — больше здоровья. Амулеты открывают также новые способности — по большей части в плане движения: например, рывок, двойной прыжок и умение передвигаться по стенам.

Хорошо, что в *Hollow Knight* движение ощущается так круто, ведь пространство очень велико. Даже заработав рекорд Steam за разблокировку всех карт игры,

мы обнаружили еще больше скрытых областей: каждая со своей темой, врагами и историей в Hollownest.

Чем больше усилий вы вкладываете, тем больше захватывающих вызовов и прекрасных, овеянных легендами пещер открывает вам *Hollow Knight*. Но сколько всего еще можно случайно упустить или проигнорировать! Целые части карты скрыты за простыми разрушаемыми стенами, и до одного из трех вариантов концовок можно дойти, даже не ступив на большую часть дополнительных областей игры.

Всего, что можно, просто не переделать. Примерно через пять часов карта открывается, и вы уже не знаете, за что взяться. Это потрясающе, но это также означает, что почти никогда не бывает неправильного выбора в том, куда идти. **LXF**

LINUX FORMAT Вердикт

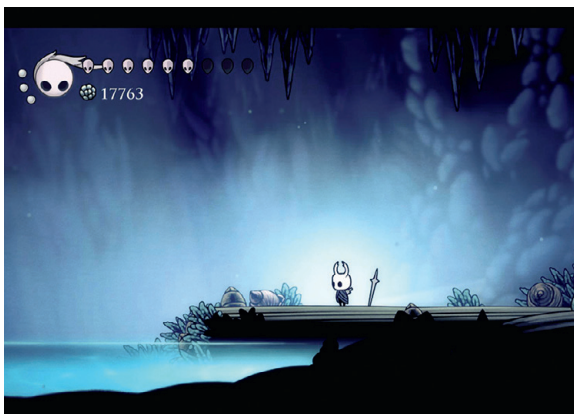
Hollow Knight

Разработчик: Team Cherry
Сайт: <http://hollowknight.com>
Цена: £ 10,99

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

» Спустя 35 часов, мы можем уверенно сказать, что *Hollow Knight* — одна из лучших игр, которые нам доводилось пробовать.

Рейтинг 9/10



» В промежутках между боями можно и о жизни подумать.



Мобильные НОВОСТИ

СОТРУДНИЧЕСТВО

Librem 5: Свободен и открыт

KDE и Purism создают Linux-смартфон с рабочим столом Plasma Mobile.

Для сбора средств на создание смартфона с полностью свободным ПО компания Purism, производитель высококачественных и уважающих свободу пользователя устройств, запустила в конце августа краудфандинговую кампанию; планируется привлечь \$1,5 млн, из которых к середине сентября было собрано около \$400 тыс. Сначала для смартфона, получившего название Librem 5, планировался рабочий стол GNOME, но он не для мобильных устройств, и окончательный выбор пал на Plasma Mobile. Свободная, открытая и полнофункциональная графическая среда для мобильных устройств KDE Plasma Mobile уже не раз протестирована на различном существующем «железе». Однако большинство смартфонов содержат аппаратные средства, требующие

проприетарного ПО; это против принципов свободы и открытости KDE и затрудняет сборку, препятствуя полному доступу к защищаемым коммерческой тайной компонентам. «На рынке мобильных ОС полностью свободная платформа Plasma Mobile станет серьезным игроком, разрушающим существующую ныне дуополию», говорит CEO компании Purism Тод Уивер [Todd Weaver]. Librem 5 позиционируется как сверхзащищенный смартфон с упором на сохранение приватности. По словам разработчика, это будет первое мобильное устройство с нативным IP, использующее децентрализованную коммуникацию, защищенную сквозным шифрованием. Librem 5 получит экран 5", поддержку 2G/3G/4G, GSM, UMTS и LTE, аппаратные выключатели для камеры, микрофона, Wi-Fi/Bluetooth и Baseband.



► По умолчанию, Librem 5 будет поставляться с предустановленной ОС PureOS, которую пользователь сможет заменить на другой дистрибутив GNU/Linux.

BLACKBERRY

Ежевичная поляна

Меняем физическую клавиатуру на гигантскую батарею и водонепроницаемость.

На конференции GITEX Technology Week в Дубае BlackBerry Mobile представила смартфон BlackBerry Motion. Смартфон BlackBerry Motion оснащен полноцветным антибликовым 5,5-дюймовым дисплеем Full HD, процессором Snapdragon 625, 4 ГБ ОЗУ, камерой 12 МР с апертурой f/2.0. Место физической клавиатуры под экраном занял встроенный в кнопку «Домой» сканер отпечатков пальцев, украшенный логотипом BlackBerry. Встроенная система безопасности включает защищенную ОС и проприетарную технологию BlackBerry Limited для создания корневого центра доверия и добавления ключей безопасности к процессору. Это первый телефон BlackBerry, соответствующий стандарту IP67 по защите от влаги и пыли; также впервые у телефона BlackBerry настолько емкая батарея —

4000 мА·ч, а технология Quick Charge 3.0 обеспечивает 50% заряда за 40 мин.

Клавиша Convenience дополняет функциональность и контекст клавиши быстрого доступа: 4 настраиваемых профиля (дом, автомобиль, офис и пользовательский профиль) с тремя триггерами, вызываемыми домашней сетью Wi-Fi, автомобильным Bluetooth, синхронизируемыми с BlackBerry Hub встречами или чаще всего используемыми приложениями. Motion поставляется с Android 7.1 Nougat с доступом к Google Play, и полным набором приложений BlackBerry. Функция Locker Mode позволяет клиентам хранить документы и фотографии во внутренней области памяти устройства, для доступа к которой требуется введение PIN-кода или отпечатка пальца владельца устройства; режим "stealth photo" позволяет клиентам снимать изображения



► Дизайн BlackBerry Motion имеет упрочненную алюминиевую раму и мягкую текстурированную заднюю часть с мягким захватом для обеспечения большей долговечности, прочности и защиты в повседневном использовании.

и напрямую отправлять эти фотографии во внутреннюю область памяти, без автоматического сохранения в облаке.

ДЕВАЙСЫ И ГАДЖЕТЫ

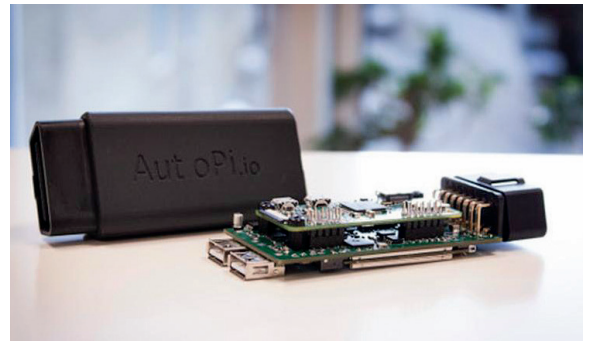
AutoPi, донгл датский...

Добавьте интеллекта своему автомобилю с помощью устройства на Raspberry Pi.

На основе платы Raspberry Pi Zero датский стартап AutoPi.io разработал приставку [dongle], которая подключается к диагностическому разъему OBD II и может быть запрограммирована для отправки предупреждений, отслеживания местоположения автомобиля, контроля за его эксплуатационными параметрами. Продукты типа Verizon Hum расширили типовые возможности использования OBD II для мониторинга в режиме реального времени и автоматизации технической помощи на дороге; AutoPi — еще один шаг в этом направлении, позволяющий разработчикам создавать собственные функции, такие как программируемые триггеры типа IFFT. Концепция датчан заключается в организации взаимодействия расположенного в автомобиле донгла и облачного сервиса AutoPi Cloud; для сбора необходимых средств стартап запустил кампанию на Kickstarter, планируя собрать \$76 тыс.

Предлагаемые приложения обеспечивают дистанционный запуск двигателя, помощь при парковке, запись видеoinформации, обнаружение сбоев, родительский контроль, автоматическую блокировку и разблокировку. Панель управления AutoPi доступна через Web, и отображает ключевые показатели производительности и условия эксплуатации, включая скорость в различные моменты времени, текущую скорость, температуру двигателя, местоположение автомобиля. Предложены виджеты для отображения отдельных параметров; ведется журнал всех поездок и связанных с ними данных; существует библиотека программных расширений — в частности, для помощи при парковке, но для его функционирования необходимо подключить датчик приближения к одному из двух USB-портов приставки.

Приставка содержит интегрированный сканирующий чип OBD-II от ScanTool, GPS,



► Приставка AutoPi для OBD II, выполненная на Raspberry Pi Zero, делает автомобиль чутьочку «умнее».

акселерометр, динамик для воспроизведения голосовых уведомлений, карту microSD на 8 Гб, 18 контактов GPIO, порт HDMI, Bluetooth, Wi-Fi, и, в зависимости от версии, встроенный модем 4G. Устройство работает под управлением Raspbian OS и поставляется с предустановленным ПО AutoPi.

УСТРОЙСТВА ДЛЯ ЗОЖ

Если хочешь быть здоров

В России начались продажи фитнес-браслета Samsung Gear Fit2 Pro.

Фитнес-браслеты Gear Fit2 Pro — универсальные устройства для активных людей. Они не только великолепно справляются со стандартными функциями фитнес-трекеров — отслеживанием физической активности, напоминаниями о правильном питании и режиме сна, — но и обладают рядом других полезных функций. В частности, устройства получили защиту от воды (и прекрасно подходят для использования в бассейне), а также более точное автоматическое определение активности пользователя и усовершенствованный датчик сердечного ритма.

Реализована поддержка популярных приложений для спорта и здорового образа жизни — Under Armour Record, MyFitnessPal, MapMyRun и Endomondo, а программа для водных тренировок Speedo On от знаменитого производителя экипировки для плавания Speedo позволяет отслеживать основные параметры тренировки в бассейне или открытом водоеме: длину заплыва,

среднюю скорость, стиль плавания и другие характеристики.

Новый Gear Fit2 Pro получил усовершенствованные возможности аналитики активности: после занятий он покажет графики и карту территории, на которой проходили занятия, и проанализирует, насколько эффективно шла тренировка на каждом участке пройденного маршрута.

Характеристики Samsung Gear Fit2 Pro:

- **Дисплей** 1,5" изогнутый Super AMOLED 216×432, 310 ppi, Corning Gorilla Glass 3
- **Процессор** 2 ядра, 1 ГГц
- **ОС** Tizen
- **Габариты** 25,0×51,3 мм (Ш×Д)
- **Вес** 34 г (размер L), 33 г (размер S)
- **Ремешок** Размер S (размер запястья: 125–165 мм); Размер L (размер запястья: 158–205 мм)
- **Память** 4 Гб встроенная, 512 Мб оперативная
- **Связь** Bluetooth v4.2, Wi-Fi b/g/n, GPS/GLONASS/Beidou

► **Samsung Gear Fit2 Pro**, фитнес-браслеты с водонепроницаемостью, GPS и продвинутым ПО уже продаются в фирменных магазинах Samsung и у его официальных партнеров.



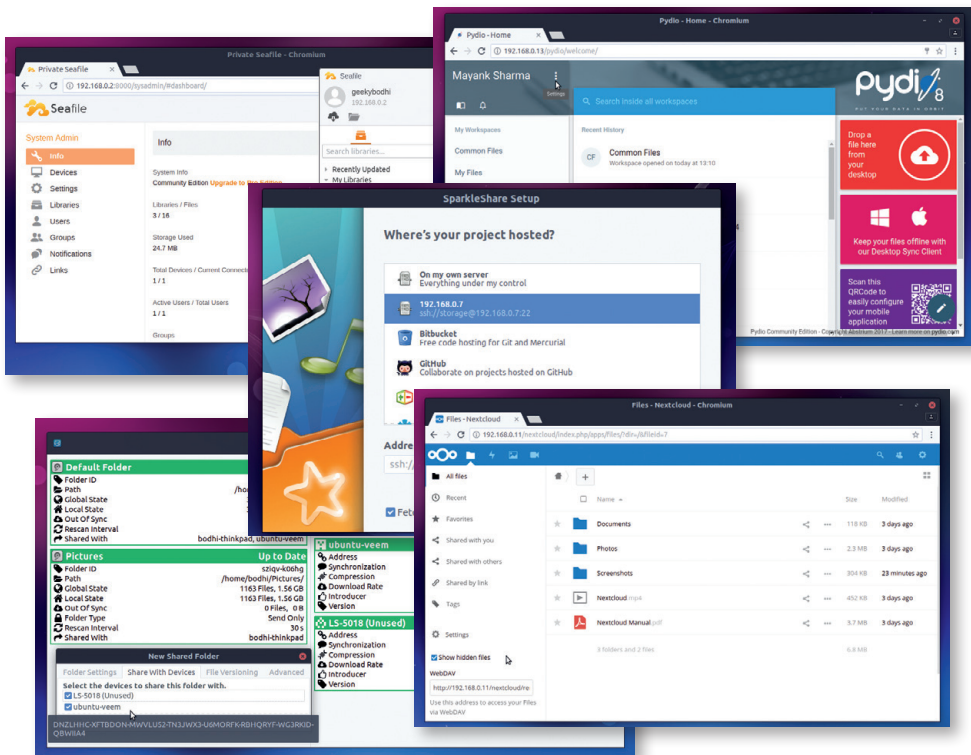
- **Датчики** Акселерометр, гироскоп, барометр, датчик сердечного ритма
- **Батарея** 200 мА·ч, зарядка Тип Pogo
- **Водостойкость** 5 ATM (до 50 метров)
- **Совместимость** Android 4.4 или более поздние; Q3U 1,5 Гб и выше, iPhone 7, 7 Plus, 6S, 6s Plus, SE, 5 *iOS 9.0 или более поздние версии. LXF

Сравнение

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

Сервисы синхронизации

Зачем оплачивать какой-то компании присмотр за вашими данными, вопрошает Маянк Шарма, когда есть масса безопасных открытых решений?



Про наш тест...

Одни наши решения проще в реализации, другие требуют точного управления доступом, а третьи ставят безопасность превыше удобства. Однако несколько основных функций общего использования и синхронизации файлов присутствуют повсеместно, и при оценке имеющихся опций мы обратим на них внимание. Шифрование данных вместе с безопасностью и управлением доступом помогают контролировать данные. Вместо они являют собой основную причину настройки своего сервера хранения.

Еще одна причина создания специального хранилища — возможность управления версиями для восстановления старых версий измененных или случайно удаленных файлов. Инструменты с мобильными клиентами ценятся выше, чем те, у которых клиенты только настольные: ведь мобильные устройства всё больше становятся компьютерной платформой по умолчанию.

Практически все мы употребляем компьютеры для добычи и накопления данных. Проведите часок в Сети, и не успеете вы оглянуться, как ваша папка **Downloads** затреплет по швам от изображений и видеофайлов. Слегка позанимайтесь кодированием, проходя учебник по программированию, и ваша папка **Documents** забьется до отказа. Возьмите на прогулку телефон с камерой, и на 32-ГБ SD-карте мигом закончится место по мере съемок окрестных пейзажей. В свете этого мысль про постоянно включенный центральный репозиторий данных, позволяющий получать легкий доступ к вашим данным отовсюду, вовсе не кажется перебором — даже обычному

домашнему пользователю. Это объясняет экспоненциальный рост Dropbox и других сервисов, которые берут деньги за управление вашими данными. Однако эти сервисы подходят не каждому. К счастью, немного DIY (Do It Yourself — Сделай Сам) может дать вам удобство сервиса хранения типа Dropbox, не заставляя делиться вашими личными и данными с посторонними.

В Сравнении этого месяца мы рассмотрим наиболее популярные инструменты

с открытым кодом для размещения ваших данных и разделения их с другими на ваших условиях. Стоимость хранения сейчас приняла расти, и в этом есть большой смысл. Вы можете установить их на старом неиспользуемом оборудовании и перепрофилировать его под сервер с одним из наших решений, при скромнейших системных требованиях. На самом деле, кое-кто даже выкладывал образы сервера для Raspberry Pi!

Наша подборка

- » NextCloud
- » Pydio
- » Seafile CE
- » SparkleShare
- » Syncthing

Постоянно включенный репозиторий и легкий доступ к данным вовсе не кажутся перебором.

Хранение и синхронизация

Это их задача; и как наши сервисы достигают своих целей?

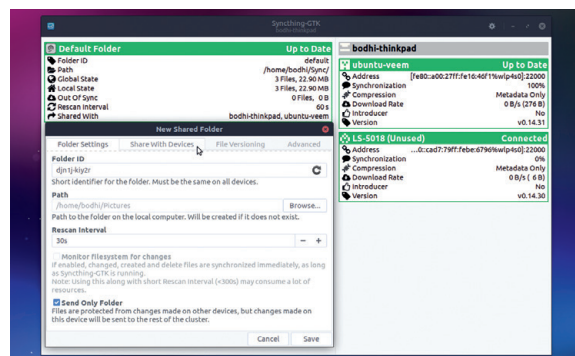
Seafile позволяет делиться библиотеками и папками с пользователями или группами, с разрешениями только для чтения или для чтения и записи. Можно также поделиться файлами со сторонними пользователями через ссылки, и они будут защищены паролями и сроком хранения. Когда библиотека скачается клиентом, после любых изменений, самая последняя версия загрузится на сервер и затем синхронизируется со всеми компьютерами.

В *Pydio* файлы делятся в папках между рабочими областями, которые могут оставаться частными или открытыми. Чтобы поделиться файлом, вы можете создать ссылку и даже пригласить сотрудников для работы над документом.

NextCloud работает очень похоже на *Pydio* и *Seafile*: файлами и папками можно поделиться с другими пользователями в системе или открыто через URL. Сервис обеспечивает синхронизацию изменений в файлах общего доступа для всех пользователей. Кроме того, в *NextCloud* можно поделиться файлами с пользователями

через другие установки *NextCloud* и даже с пользователями на серверах *ownCloud* или *Pydio*. *NextCloud* также умеет получать доступ к файлам, хранящимся на разных облачных сервисах, например, Amazon, Google и Dropbox.

Чтобы *Syncthing* смог синхронизировать файлы с другим устройством, он должен быть спарен с этим устройством через обмен ID устройств, это уникальные криптографически безопасные идентификаторы. Добавление файлов в директорию общего доступа на любом устройстве будет синхронизировать эти файлы с другой стороны. Но вы можете велеть *Syncthing* не изменять файлы в исходной папке, когда они изменяются на удаленном компьютере. Тогда только изменения, внесенные в файлы на исходном компьютере, будут отправляться на ресурс с общим доступом, но не наоборот. Вы можете использовать *Syncthing*, чтобы поделиться папкой с указанными вами компьютерами. Возможно также исключить из синхронизации определенные файлы в директории общего доступа.



» *Syncthing* умеет интегрироваться с популярными менеджерами файлов, такими как *Nautilus*, *Caja* и *Nemo*.

Аналогично, в *SparkleShare* вы создаете папки общего доступа на хосте, и все файлы, добавленные в папки проекта на клиенте, автоматически получают общий доступ с сервером и остальными подключенными клиентами. *SparkleShare* также позволяет информировать все клиенты об этих папках общего доступа. Например, к папке **FamilyShare** разрешено подключаться всем компьютерам в сети, тогда как доступ к папке **Work** будет предоставлен только вашему рабочему ноутбуку и персональному ноутбуку.

Вердикт

NextCloud
★★★★★
Pydio
★★★★★
Seafile
★★★★★
SparkleShare
★★★★★
Syncthing
★★★★★

» Все решения неплохи, но NextCloud еще умеет работать с исходными базами данных.

Управление доступом

Не позволяйте посторонним окунать ноги в ваш бассейн данных.

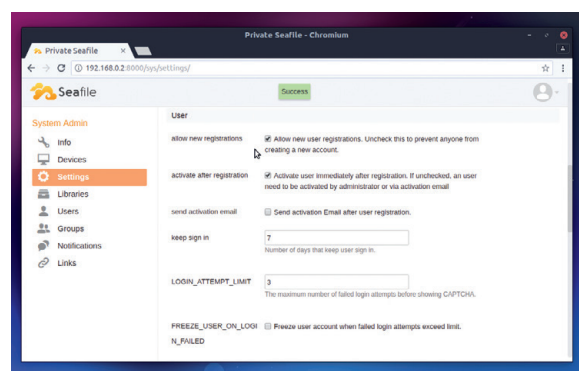
Недостаток постоянной доступности данных в том, что вы должны предпринять дополнительные предосторожности, чтобы обеспечить эту доступность исключительно для целевой аудитории. Все тестируемые здесь решения предлагают похожий подход для блокировки доступа к данным общего пользования.

Двумя исключениями являются *Syncthing* и *SparkleShare*. Они оба изначально

предназначены для синхронизации папок на разных компьютерах, и не предлагают таких же функций управления доступом, как другие решения. Таким образом, единственным способом ограничения потока данных в этих двух сервисах является возможность выбора компьютера, с которым вы хотите делить выбранную папку.

Зато в трех других решениях предлагаются сразу несколько опций обеспечения

безопасности ваших ресурсов совместного пользования. Все три позволяют защитить их паролем и задать автоматический срок отключения ссылки — по дате или по числу скачиваний. В *Pydio* можно также установить права доступа к ресурсам совместного пользования и делиться целиком папками, либо в качестве независимых рабочих областей, либо через ссылку. *Seafile* поддерживает ролевой механизм управления функциями. Будучи администратором, вы можете определять роли в *Seafile* и указывать функции, отводимые каждой роли: например, в роли гостя запретить удаление библиотек. *NextCloud* также позволяет делиться ссылками общего доступа только на чтение или с возможностями редактирования. Есть также плагин File Access control [Управление доступом к файлу], пригодный для задания групп правил, которые затем будут сопоставляться с запросом на файл: например, можно запретить пользователям из студенческих групп загрузку файлов более 100 МБ.



» В более крупных сетях можно позволить пользователям регистрироваться на сервере *Seafile*.

Вердикт

NextCloud
★★★★★
Seafile
★★★★★
Pydio
★★★★★
SparkleShare
★★★★★
Syncthing
★★★★★

» Функции контроля в *Seafile* и *NextCloud* некоторым покажутся очень привлекательными.

Развертывание и работа

Квадратный штырь в круглом отверстии? Не совсем, но настроить нелишне...

Не позволяйте дружелюбным к пользователю функциям и интуитивным интерфейсам себя одурачить. Кроме *Syncthing*, все наши конкурсанты — тщательно созданные серверные программы, и их развертывание в сети потребует усилий. Хотя по большей части процесс

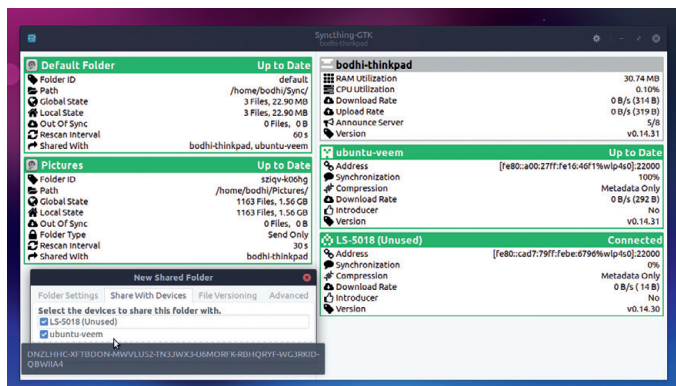
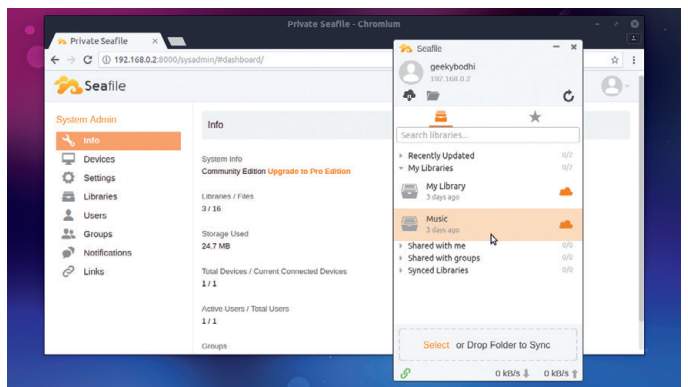
подробно разъяснен, готовьтесь работать с неожиданными зависимостями и ошибками разрешения. И хотя они должны работать на любом типе сервера Linux (мы настраивали их все на Ubuntu Server), вам придется подготовить сервер; впрочем, читатели **LXF** со стажем способны сделать

это даже с завязанными глазами. И, как и со всеми остальными серверными программами, мы предпочитаем те, что умеют адаптироваться к масштабу внедрения.

Итак, в данном разделе мы рассмотрим, чего стоит заставить эти программы работать.

Seafile ★★★★★

Настройка *Seafile* не требует особых усилий. Можно употреблять разные базы данных, в зависимости от числа обслуживаемых пользователей. Для простых развертываний сойдет база данных *SQLite*; можно развернуться на существующих установках *MySQL* или *PostgreSQL* и таких web-серверов, как *Nginx* или *Apache*. Создав базу данных, запустите из командной строки сервис *Seafile* и интерфейс администратора *Seahub*; скрипт предложит создать для сервера *Seafile* учетную запись администратора. Начните с создания библиотеки (ее можно зашифровать), а затем добавляйте в нее файлы с вашего компьютера. Библиотеки можно синхронизировать и выдавать к ним доступ по отдельности. Наведите мышью на библиотеку — и появятся рабочие ссылки. Настроив сервер, установите клиентские приложения на другие компьютеры в сети и привяжите их к серверу для синхронизации контента папок с общим доступом.



Syncthing ★★★★★

Сервер *Syncthing* устанавливать не обязательно: его можно просто скачать, распаковать и запустить. Но есть официальные пакеты для Debian/Ubuntu. При запуске *Syncthing* из терминала обычным пользователем создаются настройка по умолчанию и ключи идентификации. По умолчанию web-интерфейс доступен только с того же компьютера, что несколько раздражает; но это исправимо. В отличие от других, интерфейс *Syncthing* не позволяет напрямую получать доступ к файлам общего пользования, это скорее средство настройки работы и просмотра настроек. Для подключения устройств надо добавить ID компьютера (или отсканировать его QR-код через приложение *Syncthing*). Все клиенты *Syncthing* создают по умолчанию синхронизируемую папку с идентичным ID папки, и они уже будут иметь доступ к содержимому этой папки. Потом можно добавить папок и выбрать компьютер, с которым вам требуется общий доступ.

Документация и поддержка

Кого призвать, когда нужна помощь?

Как и во всех серверных программах, достойная инфраструктура поддержки играет ключевую роль в их внедрении. Кому же захочется остаться без помощи в трудную минуту?

SparkleShare здесь является исключением: его документация и поддержка бледнеют по сравнению с конкурентами. Правда, *SparkleShare* — единственный, у которого на главной странице предлагается инструкция по установке; но информация, которая помогла бы начинающему пользователю, крайне скудна. На сайте проекта есть wiki,

где информации ровно столько, сколько хватило бы пользователю Git.

А вот на сайте интуитивного *Seafile* — подробная документация, которая расскажет обо всех его функциях и задачах. Точно так же, *Syncthing* предлагает документацию, охватывающую всё, от базового использования до расширенной настройки. Есть также информация об утилитах CLI и подробный ЧаВо (FAQ). Если в документации нет ответа на ваш вопрос, можете разместить его в соответствующем разделе дискуссионного форума *Syncthing*.

На сайте *Pydio* тоже нет недостатка в документации. Помимо подробного руководства администратора, есть база знаний, охватывающая разные проблемы углубленной настройки. Информация о релизах выкладывается в блоге, и вы можете решить свои проблемы на активных ветках форумов. *NextCloud* идет на шаг дальше и включает подробные руководства для администраторов и пользователей и на своем канале YouTube регулярно выкладывает скринкасты. Вы также можете попросить о помощи на форумах канала IRC.

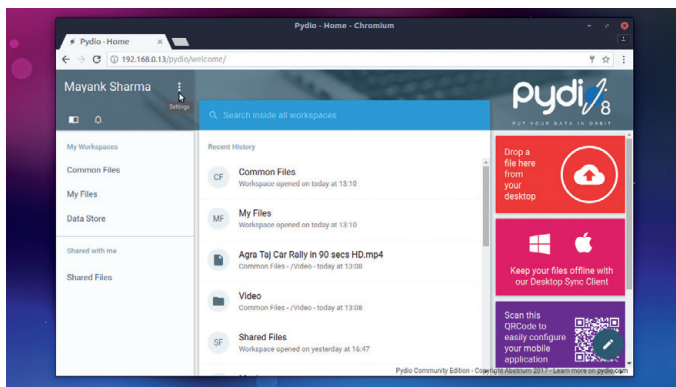
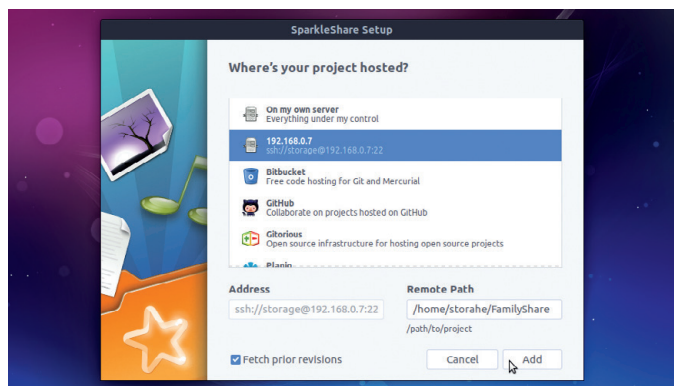
Вердикт

- NextCloud ★★★★★
- Pydio ★★★★★
- Seafile ★★★★★
- Syncthing ★★★★★
- SparkleShare ★★★★★

» NextCloud также предлагает платную поддержку пакетов, по тарифам от € 1693/год.

SparkleShare ★★★★★

SparkleShare — одно из простейших в установке решений. Потребуется всего лишь скачать и запустить скрипт. Однако с настройкой клиентов для сервера, чтобы они разговаривали с ним и потом подключались к директории общего доступа, придется поработать. Клиент можно установить из репозитория своего дистрибутива или скомпилировать самостоятельно, следуя инструкциям на сайте. Когда вы будете добавлять клиентов, вас спросят об открытом ключе SSH для сервера, так что следует изыскать способ безопасной его передачи удаленным клиентам. Аналогично, при создании директории с общим доступом *SparkleShare* появится SSH-адрес хоста и местоположение папки с общим доступом, создаваемой в его собственном пользовательском хранилище `/home/`. И надо сообщить местоположение папки с общим доступом всем клиентам, которых вы хотите синхронизировать с сервером.

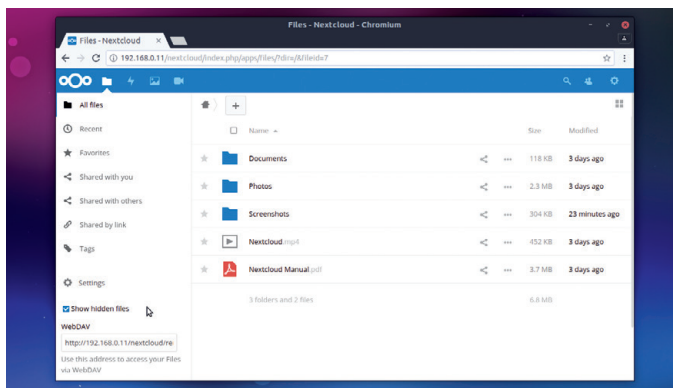


Pydio ★★★★★

Вы можете установить *Pydio*, добавив его репозиторий к своему серверу RPM- или Deb-пакетов, и развернуть его вручную, следуя инструкциям. Затем направьте свой браузер на установку и прогоните его через инструмент диагностики и мастер установки для создания пользователя Администратор и основной настройки, например, подключения к серверу базы данных. Сетевой интерфейс *Pydio* полнофункционален и интуитивен. Он пригоден для загрузки файлов — или перейдя в определенную папку, или просто перетаскивая файл с компьютера в интерфейс. Возможно, вам также понадобится перейти в панель администратора для добавления пользователей, рабочих областей и прочих настроек. Настроив всё, установите клиентские приложения *Pydio* на все компьютеры и мобильные устройства вашей сети для взаимодействия с этим центральным репозиторием, загрузки данных или их потоковой передачи.

NextCloud ★★★★★

Установка *NextCloud* сложнее, чем у других решений, но процесс хорошо документирован. Когда все компоненты будут на месте, перейдите в браузер для завершения настройки. Основанный на браузере интерфейс *NextCloud* оказывается несложным в работе даже для новичков, и это радует, с учетом множества предлагаемых функций. Можно сразу начать загружать файлы — или через web-интерфейс, или установив и подключив настольный и мобильные клиенты. Опытные пользователи, вероятно, захотят посетить раздел администрирования и настроить некоторые функции для подключения *NextCloud* с другими соответствующими сетевыми сервисами, такими как сервер директории, для импорта пользователей. Вам также, скорее всего, захочется включить и настроить некоторые из официальных плагинов *NextCloud*, по умолчанию отключенных.



Поддержка управления версиями

Вы не потеряете ни файла, регулярно делая моментальные снимки данных.

Явным преимуществом использования специального файлового хранилища является его способность хранить предыдущие версии измененных или удаленных файлов. *SparkleShare* основан на системе управления версиями Git. Его значок в системном лотке отслеживает изменения в файлах и применим для восстановления более старой версии. Обратная сторона — то, что вам не дозволяется настраивать свою работу. *Seafile* тоже периодически делает моментальные снимки файлов, папок и целых библиотек, которые

легко восстановить одним щелчком. Хотя по умолчанию хранится вся история, для каждой библиотеки можно указать срок хранения старых файлов. Можно просмотреть историю файла и восстановить содержимое файла до предыдущей версии. Что касается его разработчиков, *Seafile* использует технологию удаления дубликатов, чтобы освободить место, занимаемое старыми версиями. Аналогично, *Syncthing* поддерживает стратегию нескольких версий, охватывающую всё, от безопасного хранения только последней версии файла

до их вечного хранения. Можно также выбирать разные стратегии для разных папок. Единственная проблема, возникшая у нас с системой управления версиями *NextCloud*, была в том, что хранятся следы даже незначительных изменений, и список более старых версий может разрастись до неуправляемых размеров. В отличие от остальных, *Pydio* поддерживает систему управления версиями на основе Git; однако в наших тестах она не работала. Похоже, расширение Git пока что на стадии альфа, и работа над ним давненько не велась.

Вердикт

- Seafile ★★★★★
- Syncthing ★★★★★
- NextCloud ★★★★★
- SparkleShare ★★★★★
- Pydio ★★★★★

» Отсутствие функционального управления версиями в Pydio неприятно удивляет.

Настольные и мобильные опции

Универсальный доступ к вашим данным на любой платформе.

Загрузка данных на сервер — это только часть истории. Вам также нужен выбор кросс-платформенных клиентов для взаимодействия с сервером — в идеале, совершенно гладкого.

SparkleShare в этой области ограничен: у него есть настольные клиенты для Linux, Windows и Mac OS X, но нет функциональности мобильного клиента. Он сидит в системном лотке и отображает уведомления об операциях синхронизации и список изменений в директориях общего доступа.

Syncthing, по сути, является утилитой CLI с надстройкой из основанного на браузере интерфейса. Но в проекте также имеется кросс-платформенный графический интерфейс на Python и значок области уведомлений, который показывает все функции *Syncthing*, доступные через браузер. Кроме того, есть работающее мобильное приложение для Android.

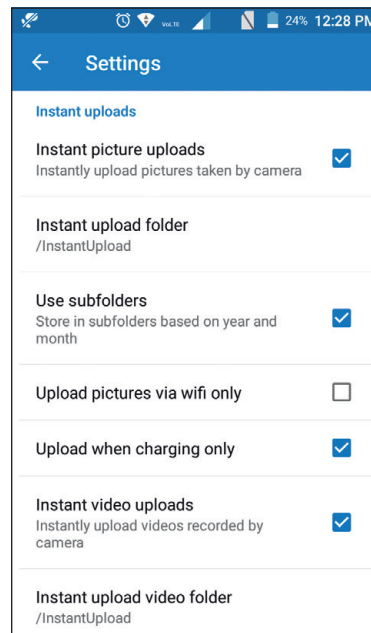
Seafile тоже, помимо Deb и RPM, имеет клиенты для Windows и Mac OS X. У каждого настольного клиента *Seafile* есть уникальный частный ключ. При соединении

клиента и сервера они обмениваются открытым ключом и обговаривают ключ сессии. Этот ключ сессии потом используется для шифрования передачи данных. Настольный клиент сидит в системном лотке и отображает уведомления об операциях синхронизации.

У *Seafile* тоже есть клиенты для Android и iOS. Самая последняя версия клиента Android поддерживает шифрование клиентской стороны для зашифрованных библиотек и двухфакторной аутентификации.

Pydio имеет настольные и мобильные приложения для всех платформ, что помогает доступу и синхронизации документов с устройств на сервере *Pydio*. Настольное приложение синхронизирует файлы по всем компьютерам, а мобильное приложение может потоково передавать аудио и видео прямо с сервера *Pydio*.

В том же русле, клиенты *NextCloud* для Android, iOS и настольной системы допускают синхронизацию и общий доступ к файлам абсолютно безопасным способом через зашифрованное подключение.



» Мобильный клиент *NextCloud* можно использовать для загрузки изображений и видео с телефона на сервер.

Вердикт

NextCloud ★★★★★

Pydio ★★★★★

Seafile ★★★★★

Syncthing ★★★★★

SparkleShare ★★★★★

» Только *SparkleShare* не хватает работающего клиента для всех платформ.

Контроль администратора

Кто вообще за это всё отвечает?

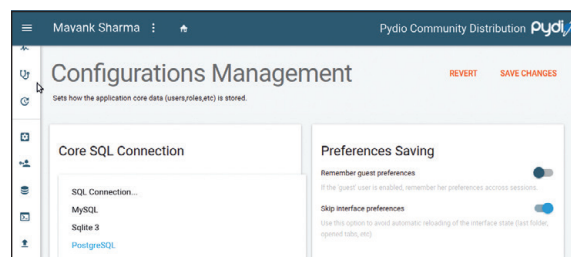
Искушенность — критический аспект любого серверного ПО. Нам нравится, когда у нашего ПО разумные настройки по умолчанию, чтобы начать использовать его сразу после развертывания. Но в долгосрочной перспективе хотелось бы получить достаточную степень управления, чтобы внедрить его в существующую инфраструктуру сети и настроить его работу в соответствии с нашими изменчивыми требованиями.

И снова тут выделяется *SparkleShare*, поскольку требует правки файлов настроек в текстовом редакторе. Можно обойти некоторые опции настройки по умолчанию, поместив их в `config.xml`, находящийся в скрытой папке `~/config/sparkleshare/`. *Syncthing* чуть лучше: его демон предлагает кое-какие общие настраиваемые опции. Можно, скажем, заблокировать доступ к веб-интерфейсу через имя пользователя и пароль, ограничить скорость передачи и применить для веб-интерфейса HTTPS.

Помимо названных опций, остальные три конкурента предлагают расширенные настраиваемые параметры. Из раздела

администратора *Seafile* можно вести мониторинг различных аспектов установки, включая информацию о подключенных настольных и мобильных клиентах. Он демонстрирует ровно столько настроек, чтобы не отпугнуть начинающего пользователя, но при этом удовлетворить потребности администратора сети достойного размера. Можно изменить логотип, определить индивидуальную политику паролей, разделить пользователей по группам и отправлять системные уведомления всем пользователям.

Pydio предлагает категоризированный интерфейс настройки. Будучи администратором *Pydio*, вы можете просматривать логи, обновлять установку до самой свежей версии и устанавливать новые плагины. Можно также настраивать сопутствующие задачи серверов, например, возможность отправлять электронную почту и планировать задачи *Cron*. Одна из уникальных функций *Pydio* — роли контроля доступа. Вы можете редактировать существующие роли и создавать новые из раздела администрирования.



» Для меньших установок, почему бы не подумать об использовании *Seafile*, *Pydio* и *NextCloud* с *SQLite3*?

В *NextCloud* настраиваемые параметры логично разделены на десятки категорий. Это число растет по мере добавления вами приложений, у которых имеются собственные настраиваемые параметры. Вы можете делать всё — от индивидуальной настройки внешнего вида вашего *NextCloud* до тонкой настройки общего доступа и определения политики паролей. В самой последней версии администраторы теперь могут играть роль другого пользователя благодаря приложению *Impersonate App*, что помогает решать проблемы пользователей.

Вердикт

NextCloud ★★★★★

Pydio ★★★★★

Seafile ★★★★★

Syncthing ★★★★★

SparkleShare ★★★★★

» У всех решений хорошие настройки по умолчанию, чтобы вы продвигались после установки.

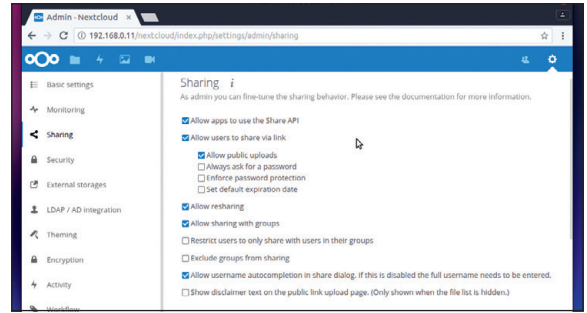
Синхронизация файлов и серверы хранения

Вердикт

Выбор сервера синхронизации и хранения зависит от ряда факторов. По нашему мнению, двумя важнейшими факторами являются тип данных, доступ к которым вы будете открывать, и ваш уровень комфорта при внедрении сетевых программ.

Если для вас они столь же важны, предлагаем использовать *Pydio*. Он прост в установке и имеет приятный и интуитивный интерфейс пользователя. Он также набирает баллы за определенные функции администрирования пользователей, такие, как роли, и нам нравится его способность отправлять уведомления, когда файлы с общим доступом просматриваются или изменяются. Однако если вы готовы пожертвовать удобством во имя безопасности, рассмотрите *Seafile*. *Seafile* предлагает шифрование клиентской стороны и функциональную систему управления версиями. Вы можете использовать его web-интерфейс для просмотра видео, аудио, PDF, изображений и текстовых файлов. Более того, у *Seafile*

есть богатые функциями настольный и мобильный клиент, которые свое дело делают. Также предусмотрена профессиональная версия для корпоративного развертывания, и он может подключаться к серверу Collabora Online и Microsoft Office Online Server. Сервер также заслужил наше предложение за создание специальной версии для Raspberry Pi. С точки зрения безопасности нет равных *SparkleShare*. Он очень безопасен, потому что использует SSH, но недостаточно интуитивен для обычного пользователя настольного ПК, поскольку подключение клиента к серверу требует ручного вмешательства, и вы должны изыскать способ безопасной передачи ключей SSH от клиента хосту. По своей природе, и *SparkleShare*, и *Syncthing* более полезны для обобществления данных между вашими собственными компьютерами, например, офисным ноутбуком и домашним настольным ПК. Хотя их можно заставить эмулировать функции общего доступа к данным от других опций, они на самом



деле не являются подходящими инструментами для резервного копирования или создания общего доступа к семейным отпускным фотографиям, например.

Но в конечном счете, по предлагаемым функциям ничто не сравнится с *NextCloud*. Единственное, где бы мы не рекомендовали его использовать — обычный домашний компьютер с несколькими пользователями. Все остальные оценят гибкость *NextCloud*, способного легко управлять сетью из 10 или даже из 100 пользователей. Подключите парочку дополнений, и у вас будет универсальное решение для всех ваших потребностей, связанных с данными и совместной работой. Сервер предлагает солидный список функций, благодаря чему идеален для разных задач.

» У *NextCloud* широкие опции общего доступа, и легко обмениваться данными со сторонними пользователями.

I

NextCloud ★★★★★

Версия: 12.0 Сайт: www.nextcloud.com Лицензия: AGPL v3

» Предлагает функциональность, превосходящую даже серверы коммерческого хранения.

IV

Syncthing ★★★★★

Версия: 0.14.31 Сайт: www.syncthing.net Лицензия: MPL 2.0

» Применяйте для синхронизации файлов и папок между машинами у вас под контролем.

II

Pydio ★★★★★

Версия: 8.0 Сайт: www.pydio.com Лицензия: AGPL v3

» Лучший для некорпоративных пользователей, не требующих управления версиями.

V

SparkleShare ★★★★★

Версия: 1.5 Сайт: www.sparkleshare.org Лицензия: GPL v3

» Пользователям Git годен для общего доступа и отслеживания часто изменяемых файлов.

III

Seafile ★★★★★

Версия: 6.1.1 Сайт: www.seafile.com Лицензия: AGPL v3

» Менее дружелюбен к пользователю или функционален, чем Pydio, но легко развернуть.

Обратная связь

Слухи о смерти *ownCloud* преувеличены? Система управления версиями — ключевая функция для сервера хранения? Пишите: ixf.letters@futurenet.com.

Рассмотрите также...

Помимо серверов, рассмотренных в нашем Сравнении, есть множество опций, пригодных для совместного доступа к данным с нескольких компьютеров и устройств. Для начинающих есть *ownCloud*, прародитель *NextCloud*. Он всё еще в силе, несмотря на уход его основных разработчиков; но разговоры о неопределенности

его будущего не вполне затихли. Затем есть *Git-annex*, который синхронизирует файлы с помощью Git. Еще одна интересная опция — *StackSync*, который хранит настоящие зашифрованные данные на общественном облаке типа OpenStack Swift или Amazon S3, но держит привязанные к ним метаданные внутри вашей сети. В порядке альтернативы,

если вы не враг проприетарного ПО, имеется *Resilio Sync*, с модифицированной версией протокола BitTorrent. А можно выбрать решение NAS, которое выполняет в основном ту же функцию по обеспечению доступности ваших данных в сети. Этот класс серверного ПО включает такие хорошо известные опции, как *OpenMediaVault*, *NAS4Free* и *FreeNAS*. **LXF**

УЧИТЕСЬ С LINUX

Джонни Бидвелл показывает, как FOSS может облегчить возвращение в оторванный от реальности мир академического образования.

✓ Покрути спиннер

✓ Raspberry Pi

✓ Переделай ноутбук

✓ Изучи Python

✓ Поставь Linux

✓ Linux Format

Солнце заходит раньше, и родители могут почти физически ощутить 6,5 часов свободы, гарантированные им благодаря возврату их чада в школу. А те, чьи отпрыски идут в университет, могут рассчитывать на еще больше свободного времени. Однако подумайте о тех студентах, которые сталкиваются с давлением общества, напряженным расписанием и средой, которая без нужды толкает их к проприетарным экосистемам.

Пересмотр образовательной программы в Великобритании в 2014 г. заменил микки-маусовский курс Информационных и коммуникационных технологий, заточенный под *Microsoft Office*, на более солидный курс Компьютерных технологий. В некоторых школах дети обучаются основам

программирования с пяти лет. Это создает трудности не только ученикам, но и родителям, которым вопросы о Python могут показаться ужаснее вопросов о Пифагоре. А у нас есть несколько отличных проектов для Pi, чтобы дети и взрослые могли

Родителям вопросы о Python могут показаться ужаснее вопросов о Пифагоре.

учиться вместе. Идеальное введение в программирование без засилья специального жаргона — это Scratch, а когда вы созреете для подъема на уровень выше, у нас есть отличные советы по Python.

Немало отличных инструментов с открытым кодом способны выполнять ваши задачи не хуже —

а иной раз и лучше — своих проприетарных аналогов. Пакеты приложений Gnome и KDE имеют целую категорию образовательных инструментов, однако большинство дистрибутивов не включают их по умолчанию — иначе большинство дистрибутивов были бы значительно увесистее. Если вы намерены попрактиковаться в японском иероглифическом письме, улучшить знание геометрии или научиться печатать, как демон [Ред.: — А разве у демонов нет секретарей?], для вас в этих пакетах что-нибудь да найдется.

И, естественно, вы обнаружите множество инструментов с открытым кодом, не связанных со средой рабочего стола. Кроме того, также существует ряд онлайн-сервисов — и для студентов, и для преподавателей.

Учеба с ноутбуком

Ноутбук с Linux станет предметом зависти одноклассников.

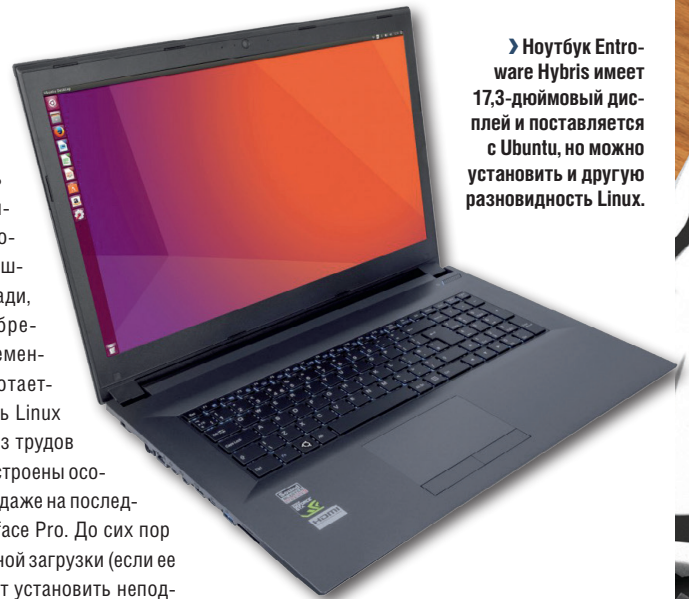
Всего за несколько лет настольный Linux прошел долгий путь, и сейчас большинство наших читателей согласится, что повседневные задачи можно с таким же успехом — если не лучше — выполнять с помощью Ubuntu, LibreOffice и <вставьте сюда название своего любимого браузера FOSS>, чем в Windows, MS Office и Internet Explorer/Edge. Все старые аргументы насчет менее раздутых программ и вирусов по-прежнему актуальны, а теперь у нас появились еще и новые, касающиеся конфиденциальности.

Конечно, есть препятствия, как реальные, так и воображаемые, не позволяющие кому-то уйти на Linux навсегда: скажем, расширенные функции Photoshop или зависимость от спецпрограмм, для которых (пока) нет порта или достойной альтернативы в Linux. Ну, а как Linux и FOSS в целом могут позаботиться о студентах и школьниках, наших будущих руководителях, которые месяц назад возвратились в классы и аудитории?

Многие студенты и школьники жаждут получить к началу учебного года новый ноутбук, и здесь определенно можно найти выгодные стороны. Навороченные процессоры i9 или Ryzen 9 нужны только продвинутым геймерам или тем, кто редактирует видео; для обычной классной работы прекрасно подойдут малобюджетные варианты. Поскольку ноутбуки i3 настолько дешевы, мы предостерегаем вас от приобретения подержанных. Кто его знает, сколько там внутри скопилось пыли, и всегда лучше получить гарантию на товар и не использовать ее, чем наоборот. Тем не менее, легко вдохнуть жизнь в старый компьютер, добавив ОЗУ или SSD (и убавив Windows). Однако избегайте 32-битных машин, потому что дистрибутивы постепенно сворачивают их поддержку.

Исторически ноутбуки и Linux ладили не очень хорошо. Нас отягощают воспоминания о том, как сомнительно интегрированная графика не работала с драйверами, обязанными ее поддерживать, как

было не выйти из спящего режима, как коротка была жизнь батареи... и о множестве всяких других вещей, которые по-прежнему посещают нас в кошмарах. К счастью, эти дни позади, и практически любой приобретаемый вами ноутбук с современными дистрибутивами сработается. Вы даже можете поставить Linux на свой MacBook, хотя и не без трудов (см. LXF225), или, если вы настроены особенно авантюрно и бунтарски, даже на последние реализации Microsoft Surface Pro. До сих пор существует проблема безопасной загрузки (если ее не отключить, она не позволит установить неподписанную ОС), но даже если эта функция неотключаемая, всё равно пригодны производные Ubuntu и Fedora. Хотя OEM теоретически не позволяют отключить безопасную загрузку, мы никогда не сталкивались с конкретным примером. А столкнулись мы с теми случаями, когда настройки UEFI изменяются только при задании пароля администратора, или эти настройки доступны через меню Выключ-



► Ноутбук Entroware Hybris имеет 17,3-дюймовый дисплей и поставляется с Ubuntu, но можно установить и другую разновидность Linux.

Сейчас есть места, где можно купить ноутбук с предустановленным Linux, и это гарантирует, что оборудование будет нормально работать с Linux, и означает лучшую поддержку на случай, если что-то пойдет наперекосяк. В Великобритании имеются компании Entroware и Starlabs Systems, которые продают ноутбуки и настольные ПК. В США есть

Практически любой приобретаемый вами ноутбук с современными дистрибутивами сработается.

чение Windows 10, если удерживать Shift и выбрать Restart > Troubleshoot > Advanced Options — всё это вполне очевидно. Многие пользователи (ноутбуков и настольных ПК) жалуются, что экран загрузки с сообщением, какую кнопку надо нажать для доступа к настройкам UEFI, исчезает слишком быстро. Это может варьироваться, но попробуйте F1, F2, F10, F12 или Esc, или введите в Google модель вашего ноутбука или материнской платы.

System76, и растет число розничных магазинов поменьше. Лидером на этом рынке является Dell, продающий ноутбуки с Ubuntu с 2012 г. под именем Developer Edition (вообще-то они начали продавать свои первые компьютеры с Linux в 2007 г., но в те времена особого успеха не добились). Нашему научному редактору нравится XPS13, однако для работы в классе это будет перебор, особенно версии QHD.

Хромбуки

Одна из причин популярности хромбуков в том, что они активно используются в учебе. У них невысокая цена (за исключением особо навороченных), впечатляющая длительность заряда батареи и при этом достаточно мощности для работы с текстом, работы в Сети и всего остального, чем могут заниматься дети. Учащихся не пугает его небольшое хранилище, поскольку стоит вам привыкнуть сохранять всё в облаке, как это становится второй натурой и избавляет от старых проблем отслеживания кучи версий документа или необходимости сохранить его

на сомнительное устройство USB и потом потерять навсегда. Не беспокоит их и то, что их устройства бессмысленны в местах, где нет доступа к Интернету, потому что они не так уж часто бывают в подобных местах. У администрируемой из облака ОС тоже есть свои преимущества — как правило, она не ломается, а тот факт, что на нее смотрят свысока, снижает риск атак вредоносным ПО. Если что-то испортится, устройство можно спокойно перезагрузить на заводские настройки, и пользователь может начать по новой. Почти все хромбуки применяют

для загрузки прошивку Coreboot, которая, хоть и использует некоторые двоичные фрагменты, всё же устраняет часть рисков проприетарных прошивок. Конечно, Chrome OS не относится к открытому коду (хотя она создана из Chromium OS на базе Gentoo), но есть отличная, стильная альтернатива — Gallium OS, и вдобавок без падения производительности из-за Crostini для настройки Linux внутри chroot Chrome OS. Правда, работа в дистрибутиве Linux при скромном объеме хранилища требует определенной дисциплины.

Открытый код для учебы

Свободные и открытые программы помогут вам в учебе, и вот наша подборка лучших инструментов и дистрибутивов для гарантии успеха в классе.

Какие бы предметы ни изучали вы или ваши отпрыски, вам обязательно понадобится редактировать документы, возможно, возиться с электронными таблицами и, вероятно, делать презентации. И здесь вашим лучшим другом будет **LibreOffice**.

Об этом прекрасном ПО ходит немало клеветы, и нам бы очень хотелось восстановить справедливость. Во-первых, *MS Office* и *LibreOffice* совместимы по формату. Файлы *Word* будут открываться в *LibreOffice Writer*, и наоборот. Любой, кто скажет вам что-то другое, лжет; *MS Office* поддерживает тип файлов OpenDocument с 2007 г. Даже правительство Великобритании утвердило OpenDocument как стандарт формата файлов для официального делопроизводства. И всё-таки *LibreOffice* — не *Microsoft Office*, и периодически вы будете наткнуться на документ, который в *Writer* выглядит

не совсем так же, как в *Word*. На то есть несколько причин: усложненное или неряшливое форматирование; какой-нибудь жуткий встроенный макрос; или, чаще всего, отсутствующие шрифты. Хотя в своей системе никому не нужен Comic Sans MS, может быть небесполезно обзавестись другими шрифтами, особенно если у вас общий доступ к документам с пользователями Windows. Требуемые шрифты можно установить на Ubuntu с помощью

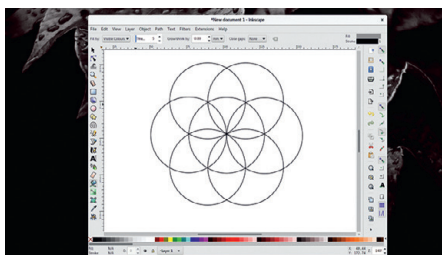
```
$ sudo apt-get install ttf-mscorefonts-installer
```

Пока мы писали, вышел *LibreOffice 5.4* с потрясающими новыми расширениями и улучшениями. Некоторое время он не будет доступен в репозиториях большинства дистрибутивов, но когда вы будете это читать, самые смелые смогут установить его из Flatpak (www.libreoffice.org/download/flatpak). Помимо слов, многие задачи также требуют наличия графики. *GIMP* отлично подойдет для боль-

способен позаботиться обо всем по части векторной графики. И, хотя это выходит за рамки данной статьи, подумайте также о настройке собственного экземпляра *NextCloud*. Это даст вам хранилище типа Dropbox, возможности совместного редактирования документов и календарь для отслеживания множества задач.

Какой взять дистрибутив?

Новичков в Linux часто обескураживает изобилие разных дистрибутивов. Мы и с этим поможем. В первую очередь следует упомянуть два специальных образовательных дистрибутива: *Edubuntu* и *Uberstudent*. Оба основаны на Ubuntu и предлагают всевозможные предустановленные прелести. *Uberstudent* позиционируется как «унифицированная система для обучения, преподавания и достижения успеха в образовании», и предла-



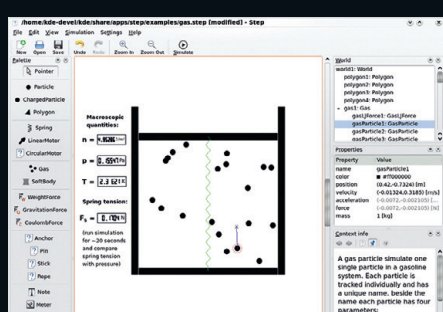
► *Inkscape* помогает рисовать фигуры, диаграммы, сакральную геометрию и прочее, последовательность Фибоначчи, и т.д. и т.п.

Следует упомянуть два образовательных дистрибутива: Edubuntu и Uberstudent.

шей части ваших обычных задач с изображениями, будь то кадрирование фотографий или добавление линз. Это не *Photoshop*, однако большинству не нужны ни все эти навороты, ни брешь в кошелке, которая следует за регистрацией на Creative Cloud. *Inkscape*, больше похожий на *Adobe Illustrator*,

дает инструменты не только для решения уравнений, но также и для организации рабочего времени [Ред.: — Вот некоторые тут еще пишут, а нам завтра в типографию сдаваться]. *Edubuntu* задуман для обеспечения преподавателей всем необходимым по умолчанию. Вы не только получаете

Фавориты FOSS



Step

Часть образовательного пакета KDE Edu, *Step* является физическим симулятором. Вы можете разместить частицы, пружины, моторы, самые разные вещи, затем нажать Go, и ситуация будет развиваться в соответствии с законами физики. *Step* научит вас Ньютонской механике и динамике мягких тел и будет куда интереснее невразумительного перечня формул.



GCompris

Образовательный пакет *GCompris* изначально предназначался для Gnome, но сейчас переписан на Qt. Здесь более 130 'activities' (игр), предназначенных для детей в возрасте от 2 до 10 лет, но некоторые из них и для взрослых тоже будут интересны. *GCompris* учит самым разным вещам, от основ работы на компьютере до составления пазлов, в том числе из картин Ван Гога.



Scratch

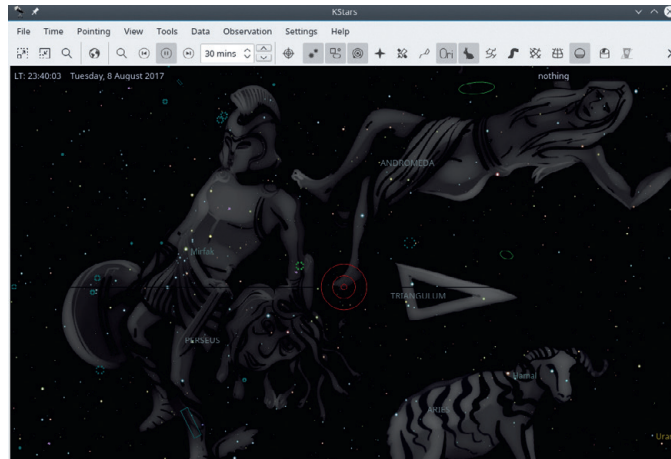
Разработанный в MIT, Scratch обучает программированию визуально, посредством соединения блоков. Это намного нагляднее, чем написание кода вручную (и избавляет от воплей при виде синтаксических ошибок), а косвенно вы еще и обучаетесь. Scratch очень хорошо работает на Raspberry Pi, но ничто не мешает использовать его на любой разновидности Linux.

всё полезное из образовательного пакета KDE, и *Gnome GCompris* (см. врезку внизу), но здесь имеется даже *Linux Terminal Server*, позволяющий настроить компьютер для нескольких простых клиентов (например, Raspberry Pi) для подключения и удаленного запуска рабочего стола. К сожалению, у *Uberstudent* не было нового релиза более 18 месяцев, да и с *Edubuntu* ситуация немногим лучше: последний релиз был в прошлом августе. Оба основаны на почтенном Ubuntu 14.04 LTS и будут продолжать получать отладки безопасности до 2019 г., но можно (и, вероятно, предпочтительнее) установить более новые версии инструментов, реализованных этими дистрибутивами, на более новый и симпатичный внешне дистрибутив.

Мы рекомендуем Ubuntu 16.04.5 (ну да, есть релиз 17.04, но он поддерживается только до января 2018 г., и тогда вам придется делать обновления, а это время лучше потратить на учебу), или Linux Mint (он основан на Ubuntu и недавно выдал релиз 18.2) в качестве ОС для начинающих пингвинофилов. Ни один из обоих не требует ничего особенного от CPU, ОЗУ или емкости диска. Mint удобнее для тех, кто переходит с Windows, и он исправил многое из того, что подорвало популярность Ubuntu. Более рискованные могут попробовать Ubuntu Gnome. Этот рабочий стол в следующем году станет стандартом для Ubuntu, поэтому попытка привыкнуть к нему уже сейчас может быть мудрым шагом. Поскольку *Edubuntu* — официальная разновидность Ubuntu, то предлагаемые им инструменты (более новые версии, никак не меньше) доступны в его официальных репозиториях и удобно разделены на категории для дошкольного, начального, среднего и высшего образования. Установите их, например, с помощью

```
$ sudo apt install ubuntu-educ-primary
```

KDE, другая крупная среда рабочего стола в экосистеме Linux, предлагает свой собственный



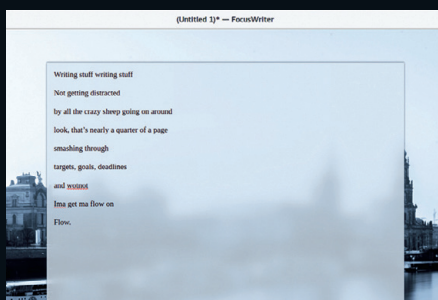
➤ *Kstars* позволит изучать небо прямо на экране. Здесь есть самые разные дополнительные материалы для скачивания, например, изображения дальних галактик.

образовательный пакет (www.kde.org/applications/education/) с отличными инструментами, которые превратят учебу в удовольствие. Здесь есть *Marble*, аналог *Google Earth*, для улучшения знаний по географии; *Kturtle* для программирования с помощью черепаший графики, где черепашка с карандашом управляется простыми командами; *KmPlot* позаботится обо всем, что связано с графикой, для увлеченных математиков; а *Parley [англ. переговоры]* (помимо того, что является вашим неотъемлемым правом, если вас захватили пираты) может помочь с освоением слов при помощи карточек.

Учитесь кодировать

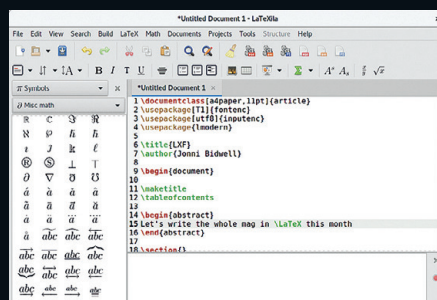
Многие считают программистов своего рода современными шаманами и жрецами, употребляющими странные заклинания и таинственные знаки, чтобы добиться почти невозможного. Да, в программировании есть своя магия, но ее могут освоить все, и мы надеемся, что новая учебная программа по кодированию поможет молодому поколению в этом. В следующей статье у нас перечисляются

несколько отличных аппаратных проектов и прекрасных ресурсов. *Scratch* (см. внизу) и *App Inventor* от MIT (<http://appinventor.mit.edu>) — отличные способы научиться кодировать визуально, но есть масса других инструментов, которые поспособствуют также и полноценному, взрослому кодированию. *Codecademy* (<http://codecademy.com>) предлагает и платные, и бесплатные курсы; они помогут вам кодировать практически на любом языке, который вам нравится — как на «простых» языках типа Ruby, PHP и Python или языках web-разработки типа JavaScript, CSS и HTML, так и на чем-то покруче, типа Java. Для сдающих экзамен по ИТ есть также отличные ресурсы от BBC (www.bbc.co.uk/education/topics/zq6hvcw). Движение Code Club выросло в мировой феномен (таким образом программированию обучилось более 150 000 детей), который объединяет добровольцев, преподавателей и учеников. Только в Великобритании имеется около 6000 клубов, где дети в возрасте от 9 до 13 лет могут заниматься бесплатно. Узнайте об этом больше на www.codeclub.org.uk. LXF



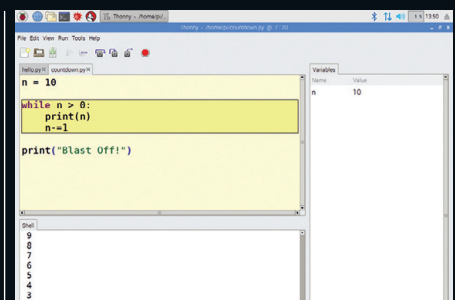
FocusWriter

Внимание некоторых людей (а детей в особенности) легко отвлечь, и в этом нет ничего страшного, но *FocusWriter* может помочь им просто сесть и писать. У него изумительно минималистичный интерфейс, где нет ничего, кроме вашего документа, пока вы не коснетесь края экрана. Предлагается расширенное форматирование и живая статистика, и можно установить план на каждый день.



LaTeXila

Форматировать формулы непросто, и редактор уравнений *LibreOffice* не всегда с этим справляется. Альтернативой является язык разметки LaTeX, однако его не легко освоить. Приложение *LaTeXila* для Gnome поможет вам освоить этот язык без слез и обеспечит ровную работу. Ваши Эйнштейновские суммирования будут смотреться просто потрясающе.



Thonny

Интегрированная среда разработки [Integrated Development Environment, IDE] для Python, *Thonny* связана с Raspbian. Переключение из текстового редактора в терминал быстро выходит из моды, а IDE обеспечивает куда более ровный способ работы. *Thonny* предлагает выделение синтаксиса, встроенную оболочку Python и мощный, но тем не менее простой для понимания отладчик.

«ЖЕЛЕЗО» В ШКОЛЕ

Сделайте обучение приятным! Поможет классное оборудование: Raspberry Pi, micro:bit — и Лес Паундер, который объяснит, что вам требуется.

Говорят, лучший способ научиться — это практика; а что может быть приятнее, чем создание нескольких проектов? Мы выбрали три проекта, которые не только помогут вам научиться программированию и улучшить свои знания по электронике, но и познакомят вас с самыми выдающимися и популярными устройствами, используемыми в школах и в современном мире. Это не просто скучное обучение: это настоящие инструменты, меняющие планету.

Мир одноплатных компьютеров и микроконтроллеров набит разработками, каждая из которых

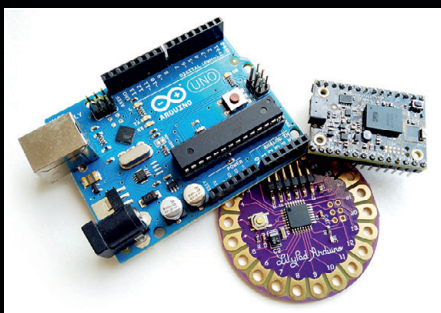
обещает стать следующим шагом эволюции в создании и обучении. Но если вы новичок в этой области, вам стоит просто придерживаться «большой тройки»: Arduino, micro:bit и Raspberry Pi.

Поддержка (и открытость) — решающий фактор для этих плат. У Arduino есть множество членов сообщества и сообщений в блоге, демонстрирующих изобилие созданных проектов — например, для micro:bit, который является относительно новым в этой области, но при этом стремительно создает вокруг себя сообщество создателей, и все они стремятся выжать свои платы до предела. Самое

сильное сообщество у Raspberry Pi: число сообщений в блоге, форумов и видео по этой плате просто феноменально, в большой степени благодаря сотрудничеству проекта со школами и такими организациями, как Computing at School. Здесь Raspberry Pi доминирует на рынке: она влияет на современное восприятие информатики [Computer Science] для старшеклассников.

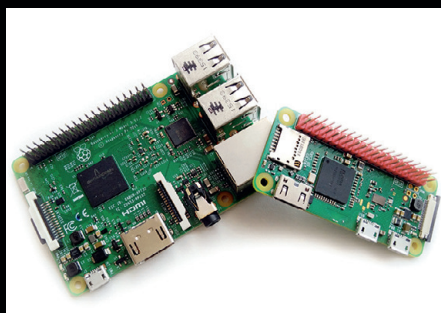
Каждая из этих плат способна учить основам информатики. Мы рекомендуем вам попробовать каждую и понять, какая из них совместима с проектами, которые вы хотите создать.

Платы всевластия



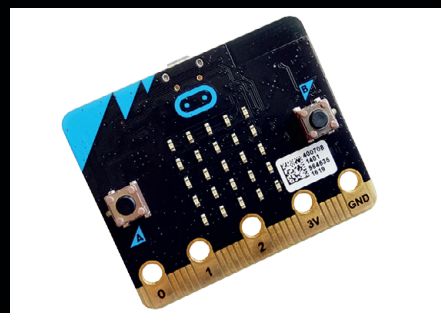
1 Arduino

Этот проект с открытым кодом зародился в Interaction Design Institute Ivrea в Италии. Созданный, чтобы помогать студентам добавлять в свои работы интерактивные функции, Arduino стоил «как хорошая еда с бокалом вина» [Ред.: — Но не у нас в Бате], чтобы быть студентам по карману. Используя микроконтроллер ATMEGA, Arduino питает тысячи проектов.



2 Raspberry Pi

Raspberry Pi взорвал мир одноплатных компьютеров — при более 14 миллионов проданных экземпляров с 2012 г. Этот компьютер за \$35 прошел путь от простого курьеза до глобального феномена, вызывающего желание учиться и творить у детей и взрослых. В сердце его — маломощный ARM CPU и легковесная ОС на базе Linux.



3 micro:bit

Проект micro:bit появился в конце 2015 г.; это плата микроконтроллера на базе ARM. Ее можно запрограммировать на нескольких языках, основанных на тексте и блоках. Миллион экземпляров распространился среди детей Великобритании еще до выхода в продажу. В плату встроены датчики направления, ориентации и ускорения.

Шейкер решений

micro:bit

Предоставьте все важные решения в жизни вашему компьютерному другу, создав свой собственный движок принятия решений на micro:bit.

Контроллер micro:bit — это простая в использовании плата, разработанная для детей, которые делают свои первые шаги в работе на компьютере. В нашем руководстве мы используем micro:bit для создания управляемого жестами «Шейкера решений»: будем задавать micro:bit вопрос, встряхивать ее и наблюдать, как ответ появляется на ЖК-экране.

Проект мы начнем с открытия браузера и посещения <https://makecode.microbit.org>: загрузится основанный на блоках язык, который мы в дальнейшем будем применять для программирования проекта. Экран поделен на три части: первый столбец — это виртуальная micro:bit для симуляции нашего кода; второй столбец показывает палитру — набор блоков, которые можно выбирать для написания кода; и последний столбец, он же и самый большой — это область кодирования, куда помещаются блоки.

В области кодирования уже виднеются несколько блоков. Чтобы удалить их оттуда, перетащите

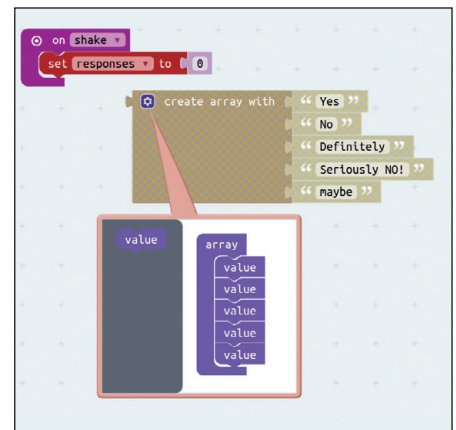
to 0 [Установить item в 0] и поместите ее внутри блока on shake. Измените 'item' на 'responses'.

Далее мы создадим массив для хранения возможных ответов на наши вопросы. Массивы можно найти в палитре Advanced. Нам требуется нижний блок create array with [создать массив с]: перетащите его и соедините с set responses так, чтобы он подходил к верху '0'. Вы увидите рядом с create array with шестерню: это означает, что в массиве можно изменять количество предметов. Нажмите на шестерню, и появится новое меню. Теперь надо перетащить value [величина] и поместить это в самый верхний слот в массиве сразу справа. Чтобы закрыть меню, нажмите на шестерню, и вы вернетесь к кодированию. Затем добавьте в пустые слоты ряд ответов на вопросы (справа показано несколько примеров).

Закончив, перейдите в палитру Basic и перетащите блок show string [показать строку], чтобы поместить его в on shake под только что созданным массивом (дождитесь, чтобы маленькая стрелочка

Вам нужны

- » micro:bit
- » USB-кабель
- » Набор батарей micro:bit или набор батарей USB
- » Код: <https://github.com/lesp/LXF228-Back-To-School/archive/master.zip>



» Создание массива в браузере может вас запутать: просто не забывайте, что новые значения добавляются в верхнюю часть массива, а не в нижнюю.

Мы используем micro:bit для создания управляемого жестами «Шейкера решений».

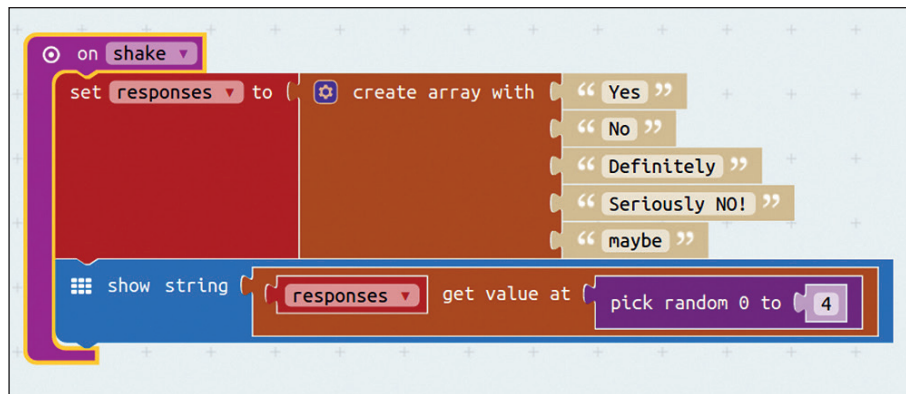
их на палитру во втором столбце и оставьте. Они исчезнут. Теперь давайте перейдем к палитре Input и перетащим on shake [при встряхивании] в область кодирования. Это послужит триггером для запуска нашего проекта.

Затем следует создать переменную, которая будет хранить возможные ответы от нашего шейкера решений. Нажмите на Variables [Переменные], затем на Make A Variable [Создать переменную] и назовите ее "responses". Теперь перетащите set item

изменила цвет на желтый). Этот блок покажет строку текста на ЖК-матрице microbit 5x5.

Теперь нам требуется способ выбора случайного ответа для отображения; итак, перетащите блок list get value at [список получает значения при] из Arrays и поместите его над 'hello' в show string. Не забудьте изменить 'list' на 'responses'.

Наш код может только выбирать первый объект в массиве 'responses', и не помешало бы добавить в проект элемент случайности. Из палитры Math



» Здесь показан весь код этого проекта. Создается не так-то много, но и это охватывает переменные, массивы и программирование, ориентированное на события.

перетащите pick random 0 to 4 [выбрать случайное число от 0 до 4] и поместите его над '0' в get value at [получить значение при]. Теперь при каждом запуске кода будет выбираться случайное число от 0 до 4. В нашем массиве пять ответов, но мы выбираем между 0 и 4: дело в том, что язык блоков, который мы используем, начинает отсчет с 0, так что наш первый объект находится в позиции 0, а пятый — в позиции 4.

Наша «программа» завершена: теперь подключите свой micro:bit, и он должен смонтироваться как USB-устройство.

Нажмите на кнопку Download [Загрузить] внизу справа, чтобы скопировать код к себе на компьютер, и перетащите скачанный файл на свой micro:bit, куда он заливается. Вы увидите на micro:bit оранжевый ЖК-индикатор. Когда индикатор перестанет гореть, код будет готов. Задайте своей micro:bit вопрос, а затем встряхните устройство, чтобы получить ответ!

Вы можете запитать свой контроллер micro:bit от официального набора батарей или воспользоваться портативной USB-зарядкой, чтобы иметь возможность обращаться к проекту, не будучи привязанным к компьютеру.

Свой контроллер GUI

Raspberry Pi

Учитесь контролировать объекты в реальном мире через индивидуальный интерфейс пользователя с помощью кода Python менее чем в 20 строк!

Работа с GPIO (General Purpose Input Output [Входы/выходы общего назначения]) Raspberry Pi поначалу может устроить. Но не бойтесь: всего за 19 строк кода Python можно соорудить графический интерфейс пользователя (GUI) для управления светодиодом и звонком.

Создание цикла для этого проекта весьма прямолинейно: просто следуйте нашей схеме и проверяйте каждое соединение. Подготовив все соединения, подключите свои аксессуары и загрузите свой Raspberry Pi в рабочий стол.

Прежде чем начать программировать, понаблюдайтесь установить одну небольшую программу. Откройте терминал (значок находится в левой верхней части рабочего стола) и введите

```
$ sudo pip3 install guizero
```

По завершении установки терминал надо будет закрыть.

Мы воспользуемся самым свежим образом Raspbian, поставляемым с *Thonny* — простым в использовании редактором Python. Вы найдете его в меню программирования.

Наш проект мы начнем с импорта трех библиотек заранее написанного кода Python. Это *guizero*, используемая для создания приложения GUI с изображениями, кнопками, ползунками и текстом; *GPIOZero*, которая связывает наш код с электронными компонентами; и из библиотеки *time* импортируем функцию *sleep*, чтобы задавать нашему коду темп.

```
from guizero import App, Picture, PushButton, Slider, Text
from gpiozero import PWMLED, Buzzer
```

```
from time import sleep
```

Чтобы задействовать в нашем проекте светодиод (LED) и звонок, надо сообщить *GPIOZero*, к каким контактам эти устройства подключены:

```
blue = PWMLED(17)
bz = Buzzer(27)
```

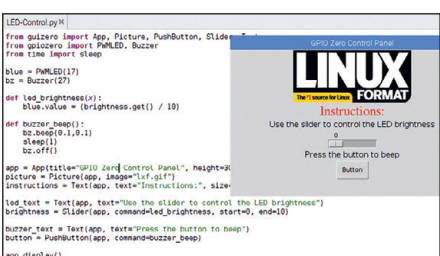
Теперь создадим две собственные функции, код которых будет управлять голубым светодиодом и сигналом.

Наша первая функция будет получать значение от ползунка, который мы создадим попозже, затем делить его на 10 и передавать значение переменной *blue*, которая регулирует яркость светодиода:

```
def led_brightness(x):
    blue.value = (brightness.get() / 10)
```

Вторая функция используется для подачи звукового сигнала пять раз в секунду (по 0,1 секунды на сигнал и на отключение).

```
def buzzer_beep():
    bz.beep(0.1,0.1)
    sleep(1)
```



Чтобы запустить код в *Thonny*, нажмите кнопку Play, и приложение запустится.

Вам нужны

- » Raspberry Pi (любой модели)
- » Голубой светодиод (LED)
- » Резистор 220 Ом (Красный Красный Коричневый Золотой)
- » Четыре провода-перемычки мама-папа
- » Макетная плата
- » Звонок
- » Код и электросхемы: <https://github.com/lesp/LXF228-Back-To-School/archive/master.zip>

```
bz.off()
```

Далее следует диалоговое окно нашего приложения 'app'. У него есть название, его определяющее, и мы задаем высоту и ширину в пикселях, чтобы вместить элементы, составляющие наше приложение:

```
app = App(title="GPIO Zero Control Panel",
          height=300, width=500)
```

Два первых элемента в приложении — это изображения, в нашем случае — логотип Linux Format. Изображения *обязаны* быть в формате GIF. Второй элемент — простой текст для отображения инструкций. Выберем шрифт и цвет:

```
picture = Picture(app, image="lxf.gif")
instructions = Text(app, text="Instructions:",
                   size=20, font="Times New Roman", color="red")
```

Теперь добавим немного советов для пользователя по управлению LED; а затем создаем ползунок *brightness* [яркость] с помощью функции с именем *led_brightness*. Значение ползунка между 0 и 10 отправляется функции, где *PWMLED* используется для задания яркости светодиода, со значениями между 0 и 1.0.

```
led_text = Text(app, text="Меняйте ползунком яркость LED")
brightness = Slider(app, command=led_brightness, start=0, end=10)
```

Затем мы аналогично создаем объяснение работы звонка плюс кнопку, пользуясь функцией *buzzer_beep*.

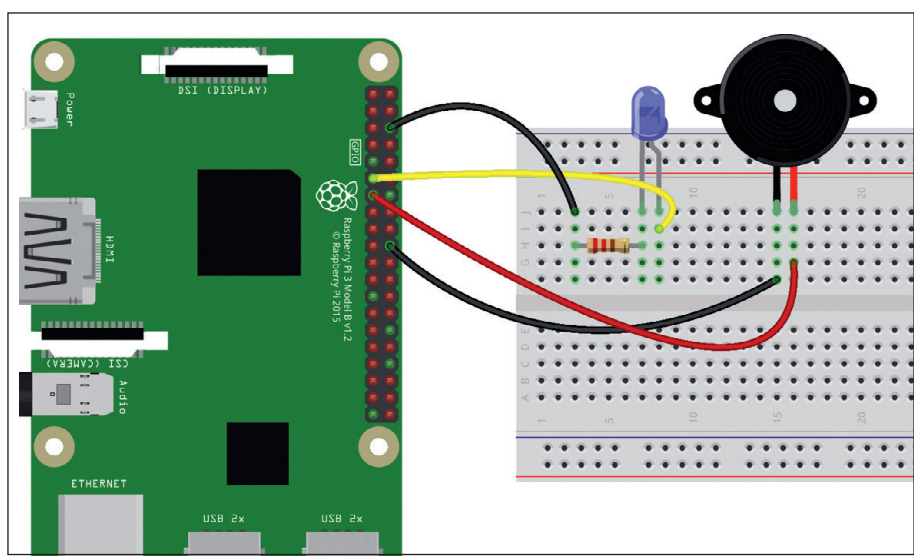
```
buzzer_text = Text(app, text="Чтоб позвонить, жмите кнопку")
button = PushButton(app, command=buzzer_beep, text="Buzzer")
```

Последняя строка кода Python обновляет приложение, чтобы показать все элементы:

```
app.display()
```

Сохраните свою работу и нажмите на значок Run [Запуск] в главном меню.

Ваше приложение появится на свет, готовое к управлению!



Такая схема с LED и звонком — идеальная отправная точка для интересующихся электроникой и творчеством. Она настолько проста, что создается за 10 минут.

Ультразвуковой сантиметр

Не завидуйте ультразвуковой отвертке Доктора Кто: ведь вы можете создать ультразвуковую мерную ленту с помощью Arduino Uno.

В начале соберите электросхему согласно диаграмме, затем вставьте свой Arduino в свободный порт USB. Установите редактор Arduino в Linux с помощью отличного руководства на www.arduino.cc/en/Guide/Linux.

Откройте приложение Arduino. У Arduino есть код, запускаемый один раз, обычно в 'setup', и код, который работает постоянно — в 'loop'.

Перед нашей установкой мы добавим три строки. Первая создает переменную — номер контакта, используемого для подключения нашего ультразвукового датчика; вторая импортирует библиотеку LCD для ЖК-дисплея; а третья сообщает Arduino, какие контакты подключены к дисплею.

```
const int pingPin = 7;
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
```

В разделе установки мы создаем подключение последовательно, чтобы избежать ошибок. Мы сообщаем Arduino, что мы используем дисплей на 2 строки шириной 16 символов. На экране мы напечатать сообщение "boot", а потом сделаем паузу:

```
void setup() {
  Serial.begin(9600);
  lcd.begin(16, 2);
  lcd.print("Booting...");
  delay(2000);
}
```

Код внутри нашего цикла проверяет расстояние до датчика и выводит его на дисплей. Добавьте две переменные: длительность сигнала датчика и расстояние. Это переменные типа float (с плавающей запятой), поскольку у них есть дробная часть.

```
void loop() {
  float duration, cm;
```

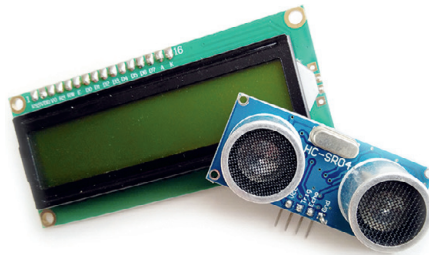
Чтобы отправить ультразвуковой сигнал-пинг, надо настроить pingPin (контакт 7) как выходной, убедиться, что он выключен, подождать, включить контакт, подождать и отключить его.

```
pinMode(pingPin, OUTPUT);
digitalWrite(pingPin, LOW);
delayMicroseconds(2);
digitalWrite(pingPin, HIGH);
delayMicroseconds(5);
digitalWrite(pingPin, LOW);
```

Поскольку для отправки и получения сигнала мы используем один и тот же контакт, наш следующий шаг — изменить контакт на ввод, а затем записать получение отраженного сигнала.

```
pinMode(pingPin, INPUT);
duration = pulseIn(pingPin, HIGH);
```

Теперь вызовем функцию, которую мы напишем после: она принимает время, за которое наш сигнал доходит до объекта, отражается и возвращается к датчику, и конвертирует его в расстояние.



» Размер нашего дисплея — 16×2, т.е. в нем 16 символов на строку и 2 строки. Наш ультразвуковой датчик — HC-SR04.

```
cm = microsecondsToCentimeters(duration);
```

Мы обновим дисплей, чтобы показать это расстояние. Перемещая курсор в верхнюю левую позицию (0,0), мы выводим "До объекта", затем переходим к следующей строке и выводим расстояние, вычисленное функцией cm. Переместив 4 пробела вдоль строки, мы печатаем "см" для подтверждения нашей единицы измерения.

```
lcd.setCursor(0, 0);
lcd.print("До объекта:");
lcd.setCursor(0, 1);
lcd.print(cm);
lcd.setCursor(4, 1);
lcd.print("см");
```

Эти данные также печатаются на последовательном мониторе для целей отладки.

```
Serial.print(cm);
Serial.print("см");
Serial.println();
```

Для финального кода внутри цикла мы делаем паузу на 300 миллисекунд и очищаем дисплей, подготовив его к обновлению расстояния в цикле.

Вам нужны

- » Arduino Uno
- » USB-вход
- » Дисплей 16×2 Hitachi HD44780 LCD
- » Потенциометр 10k
- » Резистор 220 Ом (Красный Красный Коричневый Золотой)
- » Ультразвуковой датчик HC-SR04
- » 22 провода-перемычки папа-папа
- » Макетная плата
- » Код и диаграммы: <https://github.com/lesp/LXF228-Back-To-School/archive/master.zip>

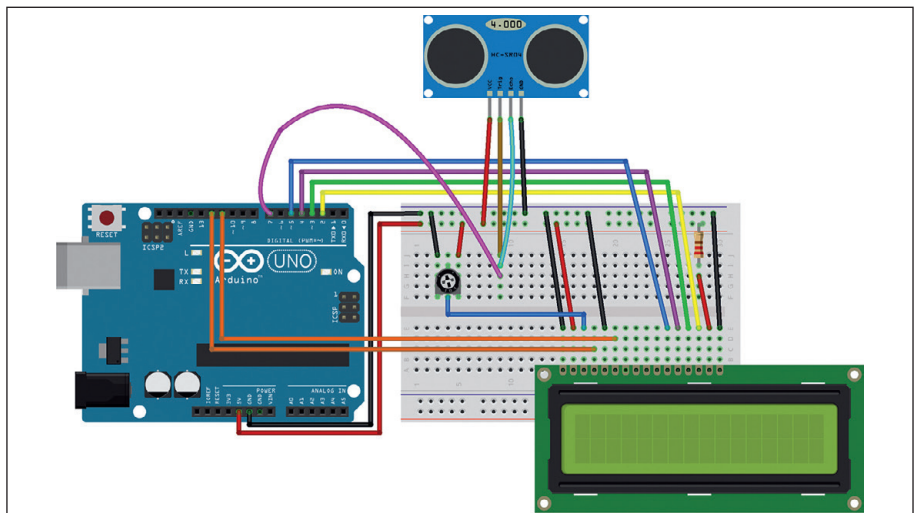
```
delay(300);
lcd.clear();
}
```

Наш последний фрагмент кода — это функция, которая конвертирует время, ушедшее на отправку, отражение и получение сигнала, в расстояние до объекта.

На сантиметр расстояния уходит 29 микросекунд (0,029 миллисекунд); а нам нужна только половина этого расстояния.

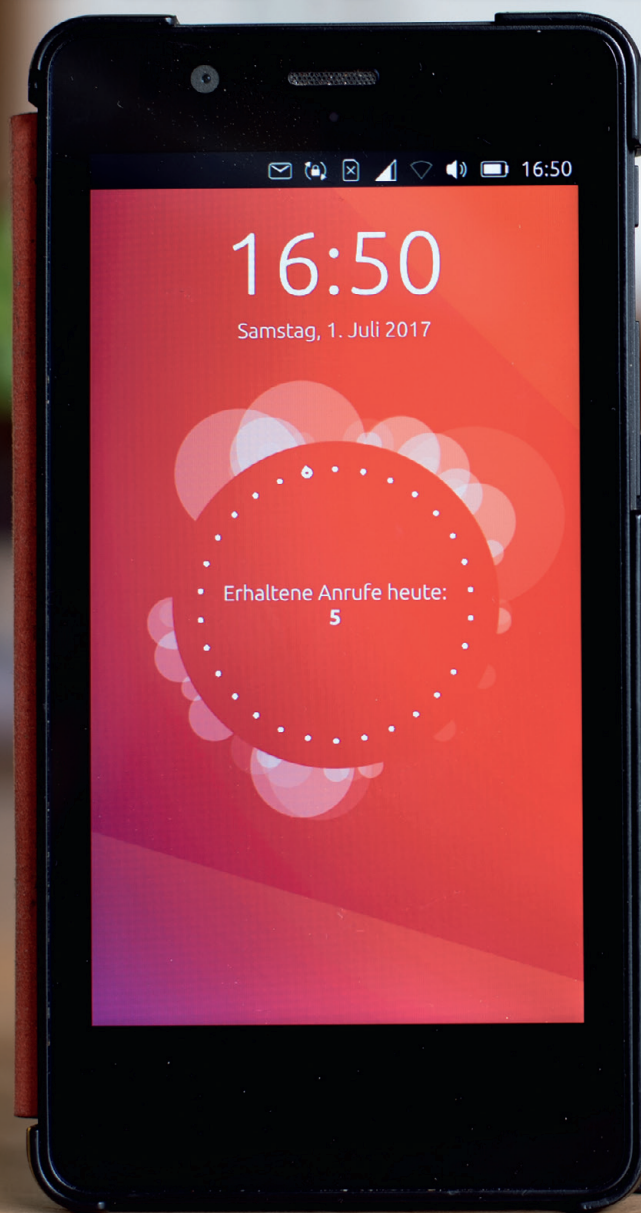
```
float microsecondsToCentimeters(float microseconds) {
  return microseconds / 29.0 / 2.0;
}
```

Ваш Arduino должен обнаружиться редактором автоматически, и вам просто надо будет нажать на стрелочку Upload [Залить] на панели инструментов, чтобы код загрузился на ваш ультразвуковой сантиметр. **LXF**



» Поначалу цикл для этого проекта выглядит запутанным. Лучший совет — аккуратно следить за каждым соединением и проверять все свои действия дважды.

КРАХ КОНВЕРГЕНЦИИ



Желаете знать, как так вышло? Симон Раффайнер поведал о взлете и крушении проекта «Ubuntu для мобильных устройств» на личном опыте.



Активистом сообщества GNU/Linux я никогда не был. На Ubuntu я перешел с Gentoo еще в 2006 г., но никогда не зависал на всех этих IRC-каналах, подписках и форумах и не посещал никаких мероприятий.

Всё изменилось в декабре 2014 г., когда я решил подыскать себе новое хобби. По основной работе я занимаюсь поддержкой суперкомпьютеров для немецкого исследовательского университета, и мне подумалось, что для разнообразия можно поработать с чем-то попроще. У меня уже был кое-какой опыт разработки для Android, но мне не нравилось, что они полностью отошли от «классической» экосистемы GNU/Linux. В UbuntuTouch же, напротив, использовались те же библиотеки, компиляторы, функции и протоколы, что и на стандартном рабочем столе или сервере Ubuntu.

Конвергенция казалась отличной идеей. Я много путешествую, и мне порядком надоело таскать с собой и смартфон, и ноутбук. А подключив смартфон или планшет Ubuntu к телевизору в отеле, плюс мышь и клавиатуру через Bluetooth, можно будет обойтись без ноутбука — вот так я думал.

Чтобы привыкнуть к системе, первым делом я решил установить Ubuntu Touch на свой смартфон Google Nexus 4 и загрузить Ubuntu SDK (комплект для разработки ПО). Разобравшись с QML и JavaScript, я создал *Game of Life* от Conway и клон популярной игры *FloodIt*; и через пару дней загрузил свои первые пакеты в магазин Ubuntu.

Тогда же мне стали ясны основные проблемы. Ubuntu Touch позиционировалась в качестве системы, идентичной Ubuntu Desktop, за исключением



установке новых вся система меняется: из-за зависимостей могут установиться дополнительные пакеты или двоичные файлы, запускаемые с привилегиями root. Приложения, запускаемые одним и тем же пользователем, нередко могут мешать друг другу и получать доступ ко всем пользовательским данным. Думаю, многим знакома ситуация с неудавшимся обновлением дистрибутива из-за того, что на каком-то одном пакете всё застопорилось или случайно удален какой-нибудь инструмент, которые вовсе не следовало трогать.

На Android операционная система и приложения полностью изолированы друг от друга. Образ базовой операционной системы доступен только для чтения (неизменяем) и един для всех устройств

О ПОТЕНЦИАЛЕ КОНВЕРГЕНЦИИ

Подключив телефон Ubuntu к ТВ в отеле, можно было бы обойтись без ноутбука — вот так я думал.

некоторых настроек в помощь работе на смартфоне. Но это абсолютно не так. (Я буду и далее использовать название Ubuntu Touch, чтобы четко различать эти две операционные системы, хотя от суффикса Touch Canonical позже отказались).

«Мы можем лучше»

Я думаю, что когда пришел заказ на создание Ubuntu Touch, системные архитекторы Canonical воспринял это как прекрасный шанс создать новую, «лучшую» операционную систему GNU/Linux с нуля. В Android и iOS тогда как раз появились некоторые новые подходы, интересные также и дистрибутивам GNU/Linux, но ими еще никто не занимался.

В большинстве дистрибутивов GNU/Linux по-прежнему используется какой-нибудь менеджер пакетов, например, RPM или Debian. Сама ОС формируется из набора базовых пакетов, но при

той же модели. При неудаче обновления ОС можно просто перезаписать тот же образ системы, находящейся во внутренней памяти, без всяких последствий. Приложения же расположены внутри песочницы [sandbox] и имеют доступ только к тем данным, которые им действительно нужны. Между пакетами нет зависимостей. Межпроцессное взаимодействие осуществляется по предопределенным методам, таким как совместное использование контента.

Чтобы воспроизвести этот подход, во всей операционной системе Ubuntu Touch было реализовано трехуровневое разделение (ядро, базовая система и приложения). Ядро и драйверы всегда зависели от используемого оборудования. Образ базовой ОС был абсолютно одинаковым на всех устройствах Ubuntu Touch. Вместо привычной команды `apt-get dist-upgrade` появилась специальная система обновления. Для приложений был

разработан особый формат пакета, под названием Click, а за запуск и изменение набора приложений теперь отвечал новый графический интерфейс Unity 8. Но этим дело не ограничилось. Чтобы обскать конкурентов, приложение Sandbox сделали жестче, чем где-либо еще. Приложения, способные украсть часть пользовательских данных? Приложения, сажающие аккумулятор без веской причины? На Ubuntu Touch такое не пройдет.

Значение этих нововведений я в полной мере ощутил в январе 2015 г., начав работать над своими приложениями. Я хотел создать сканеры для Wi-Fi, Bluetooth и мобильных сетей, чтобы видеть точки подключения, устройства и ближайшие сотовые вышки-ретрансляторы. В стандартной Ubuntu это делается без проблем, но на Ubuntu Touch была Sandbox. Здесь нельзя получить привилегии root или взаимодействовать со всеми стандартными сервисами, такими как Network Manager или BlueZ, ведь это может позволить приложениям делать всякие пакости. Вместо этого приходится работать с новыми особыми API-интерфейсами Ubuntu (программными интерфейсами приложений), которые потом, может быть, откроют диалоговое окно и запросят пользователя о праве доступа.

Заглянув в документацию, я обнаружил, что API для Bluetooth или мобильных сетей просто не было. Сетевой API был только один. Я даже не мог проверить, включен ли Wi-Fi или Bluetooth.

О ПЕРВЫХ ТЕЛЕФОНАХ UBUNTU

Не получалось сделать звонок, потому что интерфейс ни с того ни с сего прятал кнопки.

Желая помочь улучшить работу самой ОС, я предложил пару версий того, как должны выглядеть необходимые API. Я принялся отправлять отчеты об ошибках на Launchpad, чаще подключаться к каналам IRC и создавать документацию ко многим системным и аппаратным компонентам в своем личном блоге.



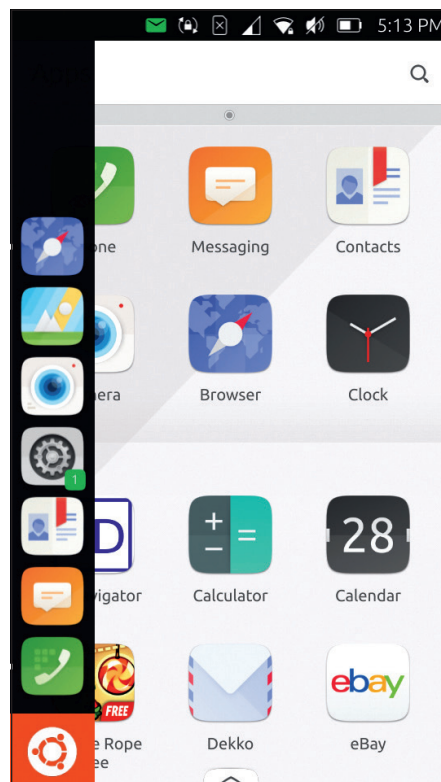
Это мое участие скоро привело меня в группу Ubuntu Phone Insiders. Я стал частью большого, яркого сообщества, приносящего всем нам огромное удовольствие.

Первый пошел

Когда испанский производитель устройств bq начал продавать смартфон Aquaris E4.5 Ubuntu Edition в рамках «flash-распродажи» в мае 2015 г., я сразу же заказал три устройства.

Как член Phone Insider, от Canonical я также получил Meizu MX4 незадолго до его выхода в продажу в конце июня того же года. Но у меня-то было четыре телефона, а большинство людей не могли добыть и одного. Со складов интернет-магазинов они исчезали мгновенно. Bq и Meizu делали поставки только в Европу и Китай, а затем в Индию и Россию. Поставок в США изначально не планировалось.

Ввод при помощи жестов и функция Scopes вызвали некоторый интерес, но первые устройства принимались неохотно. После более чем двух лет разработки они всё равно притормаживали, и их приходилось регулярно перезагружать. Изменить что-либо в системе по своему вкусу, включая мелодии звонка, было практически невозможно. Meizu MX4 перегревался. Индикатор батареи норовил показать неверные данные. Служба определения местоположения позорно подводила. Те-



Интерфейс в Ubuntu Touch выглядел мило; вот бы ее было столь же мило использовать.

Где все приложения?

Ценность мобильной ОС состоит уже не в самой ОС. Ценность в экосистеме: приложения, службы и особенно база данных, стоящая за конкретным сервисом. Скажем прямо: даже самолучшая технология ничего не стоит, если у вас нет WhatsApp.

Ubuntu была несовместима со всеми существующими средами времени выполнения. Невозможно было просто запустить или выполнить межплатформенную компиляцию приложений Android, Windows, X11 или iOS. Графическая система, службы, привязки, набор базовых библиотек — всё было иным. Создавать графические интерфейсы можно было только с помощью QML — не самый общепотребительный вариант. Не поддерживался сервер X11: вместо него был Mir, созданный тем же Canonical. Взаимодействие с аппаратными и системными службами? Получайте специальные API-интерфейсы Ubuntu.

Всё это лишило платформу привлекательности для большинства сторонних разработчиков. Никто не видел смысла вкладываться в создание новой версии своего приложения с нуля, особенно для столь узкого сегмента потенциальных пользователей. Практически ни одно приложение в магазине Ubuntu не поддерживалось его собственными разработчиками: даже клиентом Telegram Instant Messenger занимались в Canonical.

Многие разработчики внутри сообщества, такие как я, стали создавать либо «web-приложения» (предварительно сконфигурированный, неизменяемый вариант браузера с иконкой запуска), либо клоны существующих приложений. Но многие приложения опираются на платные онлайн-сервисы с крайне неблагоприятными условиями

использования. В середине 2015 г. я создал приложение *BD Navigator*, клон *Deutsche Bahn Navigator*. (*Deutsche Bahn* — национальный железнодорожный оператор Германии.) Мне пришлось декомпилировать их клиентский протокол, и я сумел воспроизвести практически любые действия, включая покупку билета на поезд. Но часть протокола была зашифрована, а ключи шифрования, полученные методом декомпиляции, в Германии запрещены. Когда я попросил разрешения, мне отказали. В итоге всё, что вышло — это маленький web-браузер со встроенными закладками на их web-страницы.

Та же история — с *WhatsApp*, *Twitter*, *Google Plus* и т.д. *WhatsApp* якобы хотел получить семизначную сумму только за лицензию на доступ к своим API, даже не за само рабочее клиентское приложение. В *Ubuntu Touch* есть встроенный *Instagram Score*, но позже и его пришлось удалить из-за изменения лицензии API.

Я до сих пор считаю, что это был самой большой просчет *Ubuntu Touch*. Проблемы в самой операционной системе еще можно было бы устранить, но никаких усилий сообщества не хватило бы, чтобы привести на платформу все недостающие приложения и сервисы. Я даже придумал идею конкурса приложений под названием *UbuContest*, который был запущен в сотрудничестве с *Canonical* в период с июля по сентябрь 2015 г. Ситуацию это не изменило, зато было очень весело.

Переломный момент

В сентябре 2015 г. Кристиан Паррино [*Cristian Parrino*] покинул пост вице-президента подразделения *Canonical* по работе с мобильными устройствами,



➤ О потенциале *Ubuntu Touch* и устройств на ней можно судить по малому числу собственных приложений.

Смартфон вам не ПК

В мире ПК мы привыкли, что можно установить и загрузить любую операционную систему, способную взаимодействовать с оборудованием, будь то *Windows*, *Ubuntu* или *OS/2*. Одно и то же ядро *Linux* может работать на множестве аппаратных средств; многие драйверы имеют открытый исходный код; и в целом незначительно важно вникать в то, какое там железо.

В мире встроенных устройств всё иначе. Здесь нет стандартного процесса загрузки; каждое устройство имеет свою специфику, и большинство производителей не используют открытые драйверы. Проприетарные драйверы, как правило, даже

не являются частью ядра *Linux*, но работают как обычные процессы среды пользователя. *Android* является сторонником данной модели, а дистрибутивы *GNU/Linux* — нет.

Решение *Canonical* состояло в том, чтобы частично адаптировать *Android* для *Ubuntu Touch*. В этих устройствах использовалось измененное ядро *Android*, а с помощью специальной библиотеки *libhybris* стандартные двоичные файлы *GNU/Linux ELF* могли взаимодействовать с библиотеками *Android*. Части драйверов *Android* были помещены в контейнер *LXD*, чтобы изолировать их от остальной части системы *Ubuntu*.

и в последующие несколько месяцев проекту уделялось гораздо меньше внимания и усилий.

Поскольку компании отказывали в доступе к своим сервисам, стало ясно, что *Ubuntu Touch* не найдет массового сбыта на рынке. Будет ниша опытных пользователей, хакеров и сторонников защиты личных данных: всех, кто не желает, чтобы *Google* и АНБ за ними шпионили, и готов раскошелиться на *Blackphone*. *Canonical* перенаправил свои внутренние ресурсы на новые устройства и *Convergence* — а действующих пользователей это ничуть не интересовало. Их интересовала, например, хорошая интеграция с их настольными ПК и *ownCloud/NextCloud*, для избавления от *Google*; *Firefox* и блокировщик рекламы вместо самопального браузера. Но самое главное — телефоны, купленные уже полгода назад, так и не стали нормальной «рабочей лошадкой».

В *Canonical* пытались отправлять всем официально поддерживаемым устройствам обновления OTA (*Over The Air*, «по воздуху») каждые четыре недели. И только четыре из 15 обновлений OTA хотя бы приблизились к этой отметке; остальные пришлось ждать вплоть до 11 недель. Число изменений (исправлений ошибок / добавления функций) достигло 256 в OTA-10 в апреле 2016 г. и упало до 23 в OTA-14 в декабре 2016 г. Исправленные были ошибки не единожды возвращались в виде регрессий. Обсуждения того, какие ошибки и функции должны войти в следующее обновление OTA, не проводились.

Не смогли вмешаться даже разработчики от сообщества: большинство решений принималось за закрытыми дверями или на закрытых форумах по отслеживанию ошибок, доступных только *Canonical* и их партнерам. В итоге, разочарованные отсутствием видимого прогресса, неучетом их пожеланий и растущим списком недостающих приложений и сервисов, пользователи всё чаще и чаще принимались конвертировать свои телефоны в *Android*.

Для разработчиков проекта ситуация была еще жестче. Я присоединился к ним только из-за отсутствия API для моих приложений. И мало того, что никакого прогресса не было; приходилось еще и постоянно воевать с сотрудниками *Canonical* насчет того, нужна ли на платформе та

или иная функция и как именно ее следует реализовать. SDK часто был недоступен по нескольким дням, а то и недель. Разработчики, занимавшиеся столь необходимыми приложениями, отворачивались от *Ubuntu Touch*.

Выход Convergence

В ноябре 2015 г. *Microsoft* выпустила *Continuum*, свой «нож в спину» *Convergence*, опередив *Canonical* на несколько месяцев. Планшет *bq Aquaris M10*, действительно сопоставимый с полноценным рабочим столом и способный запускать приложения *X11* внутри *Sandbox*, вышел только в апреле 2016 г. Кое-кто из энтузиастов его приобрел, и кое-кто до сих пор с удовольствием использует эти устройства; но в обзорах их всё равно называли медленными, а программное обеспечение — слишком незрелым и ограниченным. В одном немецком источнике был даже заголовок: «Не знаете, как угробить планшет? Спросите *Ubuntu*».

Разрыв между ожиданиями от *Ubuntu Edge* и реальностью становился всё шире и шире. В *Canonical* пообещали «убунтофон», но не слишком утрудили телефонной частью. Более ранние модели были не в состоянии официально получить конвергенцию из-за отсутствия *HDMI*-выходов — о чем многие покупатели раньше не подумали. Не было их и на последнем официальном телефоне, *Meizu Pro*, и потоковая передача с дисплея на какой-либо экран опять осуществлялась через *Wi-Fi* — решение работающее, но не самое удобное.

Думаю, что последние разработчики, включая меня, покинули проект где-то весной 2016 г. Проекты с открытым кодом должны приносить удовольствие, раз уж вам за них не платят, а этого-то и не наблюдалось. Я стал удалять себя с *Launchpad* и прочих каналов связи.

В апреле 2017 г. Марк Шаттлворт закрыл программу мобильных устройств, и *Canonical* уволила большинство разработчиков. В настоящее время интересы компании крутятся вокруг *Internet of Things* и центров обработки данных, *Ubuntu Touch* частично вошла в *Snappy* и *Ubuntu Core*. От *Ubuntu Touch* ответвился проект *UBports*, а от *Unity 8* — такие проекты, как *Yunit*. Пускай мечта и не сбылась — но я несколько не жалею о приобретенном опыте. LXF



Установка
LineageOS —
на стр. 62

LINUX НА ВАШЕЙ МОБИЛЕ

Джонни Бидвелл
убеждает вас сбежать
от дуополии Apple–Google
и сделать свой смартфон
более здоровой
площадкой...

На данный момент имеется около двух миллиардов телефонов на Linux в виде Google Android. Однако устаревающее ядро Linux, погребенное под несколькими слоями Java и проприетарным магазином приложений, еще не создает свободной мобильной утопии.

Кончина Ubuntu Phone, не говоря уже о других подобных попытках в мобильном мире, оставшихся на обочине дороги, нанесла удар по надеждам на более открытую мобильную платформу — однако не все надежды утрачены. Сообщество UBports намерено вернуть к жизни Ubuntu Touch, и уже было выложено первое обновление. Многие другие проекты живы и здравствуют на аппаратном уровне, и на разных уровнях в программном стеке. Ап-

droid существует достаточно долго, чтобы люди привыкли к нему и его экосистеме приложений, поэтому совместимость является важным аспектом для альтернативных мобильных ОС. В связи с этим *libhybris* (уровень совместимости между GNU C Li-

ней и стандартизацию средних уровней стека мобильного Linux, тоже набирает обороты и значительно облегчает разработку мобильных ОС на базе Linux, позволяя не беспокоиться о том, что фурычит внизу.

Уязвимости вроде недавно опубликованного broadpwn, способного оказать воздействие на миллиард устройств, создают броские заголовки прессы и панику, однако в большинстве случаев пользователи бессильны применить защиту. Поэтому хотите ли

вы иметь полностью открытое устройство (удачи вам в этом) или просто возможность получать обновления ОС, не атакуя просьбами своего сетевого провайдера, читайте дальше — и сделайте свой телефон (более) свободным.

Сообщество UBports намерено вернуть к жизни Ubuntu Touch.

brary и Bionic в Android) позволяет системам на *glibc* взаимодействовать с оборудованием, для которого существуют только бинарники Android.

Проект Halium, намеренный снизить фрагментацию через унификацию более низких уров-

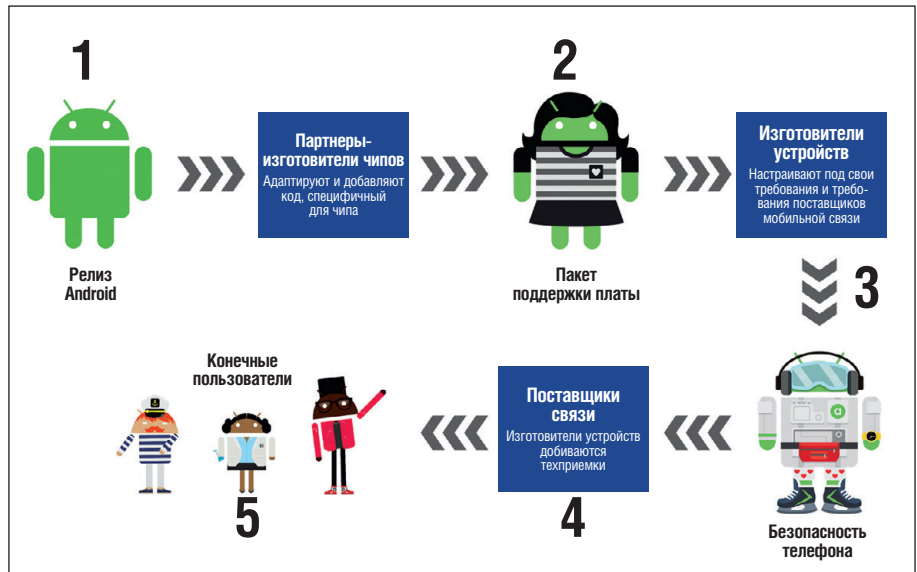
Разительные отличия прозрачного мира Linux от мутных вод экосистемы Android увидеть легко. Пользователи Linux — как минимум те, у кого установлены современные дистрибутивы — привыкли своевременно получать обновления безопасности, которые даже способны устанавливаться автоматически, в зависимости от пользовательских настроек. Для сравнения, практически половина устройств Android, бывших в пользовании на конец 2016 г., не видели ни одного обновления безопасности по крайней мере год — по словам сотрудников по безопасности в самом Android.

Это не целиком вина Google: они выпускают обновления безопасности ежемесячно, но эти обновления продвигаются только на собственные устройства Google, Nexus и Pixel. Адаптация обновлений для устройств зависит исключительно от производителей этих устройств, а продвижение обновлений к пользователям — от операторов связи. Большинство OEM не слишком заинтересовано в поддержке того, что ими уже продано, а операторы связи не слишком хотят обновлять ужасные приложения, которые они рвутся добавлять в Android «для улучшения взаимодействия с пользователем».

Есть и другие препятствия: в прошлом году выяснилось, что Qualcomm прекратил поддержку своих чипсетов Snapdragon 800/801, угробив надежды на то, что устройства с этими чипами получат официальное обновление Android 7.0 Nougat. И опять же, в данном случае трудно определить

Звучит немало обвинений, и вряд ли найдется хоть один чистенький.

степень вины: Qualcomm заявляет, что OEM больше не хотят поддерживать телефоны на этих процессорах, так зачем тратить силы? А ведь на момент объявления (август 2016 г.) ряд устройств с этими чипами, включая популярный Sony Xperia Z3, родился только в конце 2014 г. Два года жизни для



» Схема от Google показывает кучу причин, почему обновления не всегда доходят до конечного пользователя — но это знание утешает мало.

телефона даже при нынешней культуре одноразового потребления выглядят довольно коротким сроком. Со своей стороны, в Sony заявили, что чипсет не пройдет тесты совместимости Google, которые сейчас требуют поддержки OpenGL ES 3.0 и Vulkan.

Ясно одно: сейчас звучит много обвинений, и, вероятно, вряд ли найдется хоть один чистенький участник всего этого.

Google, во всяком случае, честно пытается улучшить ситуацию с обновлениями. Проект Treble, часть следующей редакции Android, стремится отделить реализуемые производителями фрагменты операционной системы от остальной ее части.

Если стандартизировать интерфейс производителя, то становится неважно, что производители чипов вдруг перестанут обновлять свой код для соответствия новой среде Android: он отлично подойдет предыдущему интерфейсу. Очевидно, что

всё это не поможет устройствам, от которых уже отказались, но зато устройства, поддерживающие следующую итерацию Android 'O' [Ред.: — «Отвратный»?], не должны обнаружить, что они внезапно лишились поддержки.

При желании, пользователи могут избежать этого синтетического или иного морального устарения, установив LineageOS, о чем вы прочтете подробнее на стр. 62. Доступны и другие специализированные прошивки-ROM на Android, включая CopperheadOS и Replicant с расширенной безопасностью.

Возможность установки сторонних ROM зависит от допустимости разблокировки загрузчика вашего телефона. Одни производители такое запрещают (хотя данная тенденция идет на убыль); другие (например, Motorola и HTC) предлагают симпатичную web-форму, через которую можно поменять номера IMEI или иные ключи на коды разблокировки, которые, в свою очередь, можно использовать с fastboot, частью Android SDK. Устройства Nexus разблокируются без кода. »

Neo900

В 2009 г. Nokia запустила N900, смартфон с операционной системой Maemo 5, на базе Linux. Ходили слухи, что версия разработчика выйдет с MeeGo Linux, драгоценным для Linux Foundation союзом Maemo и Moblin от Intel. Однако и Linux Foundation, и Intel отказались от MeeGo в пользу Tizen, и редакция разработчика N900 так и не появилась. Тем временем партнерство Microsoft с Nokia в 2011 г. и переход последней на Windows Phone способствовал отказу Nokia от MeeGo.

Однако интерес к оборудованию (и программному обеспечению, в форме Mer) не утихает, и проект Neo900 работает над неофициальным продолжением N900. Оборудование строится на свободной плате GTA04 и будет мощнее устройства-оригинала — хотя

по нынешним меркам и с одним ядром 1 ГГц и 1 ГБ ОЗУ это весьма скромно. Оно сможет работать на любой ОС, поддерживаемой GTA04, включая QtMoko и Debian Mobile, и команда надеется, что на устройство будут портированы и другие ОС. По части драйверов, здесь всё максимально свободно, кроме 3D-ускорения для чипа PowerVR и прошивки для модема для прямой передачи данных и чипа Wi-Fi.

Сейчас команда принимает предзаказы на устройство, цена которого составляет €480. Владельцы N900 могут просто приобрести плату и установить ее в свой корпус, сэкономив €130.

Дополнительную информацию см. на сайте <http://neo900.org>.



» Возможно, сейчас Nokia N900 выглядит устаревшей, но у нее есть свои поклонники.

Неудачи телефонных ОС

Многочисленные попытки создать более открытую телефонную ОС потерпели неудачу: Firefox OS, Maemo, MeeGo и Ubuntu Touch более не существуют. Но хотя бы две последних были спасены сообществом в виде UBports (см. далее) и Mer (<http://merproject.org>).

Mer (сокращение от Maemo Reconstructed) изначально был открытой версией своего прародителя MeeGo, но разработка прекратилась, когда MeeGo стал набирать силу. Не все разработчики были в восторге от направления движения MeeGo, и в 2011 г. работа над проектом негласно возобновилась. Когда в 2013 г. объявили о Tizen (и отменили MeeGo), проект Mer вышел на суд публики. Его миссия состоит в предоставлении промежуточного ПО (а не полной ОС), помещаемого между ядром Linux и графическим интерфейсом пользователя. Таким образом, производители могут настроить аппаратную часть, выбрать GUI (есть масса активных проектов на выбор) и забыть об остальном. Mer используется в Nemo Mobile, продолжении MeeGo от сообщества, и содержит хорошо известные технологии Linux, такие как *Wayland*, *systemd* и *libhybris*.

«Hybris — это решение, которое подтверждает *hybris*, позволяя применять основанные на *Bionic*

адаптации [оборудования] в системах *glibc*», говорит страница Hybris (она же — *libhybris*) на GitHub, и мы не представляем себе лучшего описания.

Bionic — это разработанная Google и используемая в Android библиотека на C под лицензией BSD; собственно, доступ ко многим драйверам можно получить только посредством функциональных вызовов *Bionic*. *Libhybris* может переделать эти вызовы в вызовы *glibc* (библиотеки GNU C), чтобы они были пригодны для систем чисто на Linux. BSD-лицензия *Bionic* разрешает в экосистемах Android разработку проприетарных инструментов, что так раздражает пуристов FOSS.

Более широкий по охвату проект Haliun намерен снизить фрагментацию телефонных ОС на базе Linux через предоставление общей базы, включая ядро Linux, уровня аппаратной абстракции Android (и драйверы доступны только в двоичной форме) и вышеупомянутой *libhybris*. Все это должно быть портируемым, и позволит легко создавать новые проекты.

Когда проект MeeGo был законсервирован, несколько работников Nokia ушли из компании и присоединились к финскому производителю смартфонов Jolla. В Jolla намеревались производить смартфоны и планшеты (от вторых они уже отказались) и независимую Sailfish OS. Sailfish —

это гибрид Linux/Mer, в который Jolla добавили проприетарный UI, хотя есть его альтернативы. Она поставляется как стандарт в телефонах Jolla, доступных в развивающихся странах, как и Jolla C, предоставляемый только разработчикам через программу сообщества по устройствам. Sailfish OS лицензирована для других поставщиков, и в феврале объявляли, что версия Sailfish для среднего класса Xperia X от Sony будет готова к продаже. Sailfish OS также была неофициально портирована на несколько других телефонов, а также на оснащенные ARM7 Raspberry Pi. Приложения Sailfish созданы на Qt и C++ с помощью ныне проприетарного модуля Silica QML. Основной коммерческий аргумент SailfishOS — *sfdroid*, позволяющий приложениям Android работать на Sailfish.

Проблемы Tizen

Цель проекта Tizen — предоставить не просто мобильную ОС, но ОС, подходящую для IoT и IVI (in-vehicle infotainment — автомобильная электроника для информирования и развлечений), носимых устройств, ТВ и т. д. Изначально им занимались Samsung, Intel, Huawei и прочие индустриальные гиганты (группа, ранее известная как LiMo Foundation), и разработка велась под эгидой Linux Foundation. Сейчас Samsung является единственной компанией, разрабатывающей эту ОС и основанные на ней продукты, но Mer Project надеется работать вместе с Samsung. В апреле гуря безопасности Амихай Нидерман [Amihai Neiderman] обнаружил в ПО несколько критических уязвимостей и заявил, что это, возможно, «худший код из всех, которые я видел». Tizen использует стек телефонии oFono, применяемый другими проектами FOSS.

Сообщество KDE также имеет свои цели в этой области, со своей собственной Plasma Mobile OS (<https://plasma-mobile.org>). Она использует среду Kirigami, чтобы сделать UI легковесным и красивым; загляните в приложение Kirigami Gallery на Google Play, если не верите нам. Сама ОС по-прежнему в стадии разработки и не готова к массовому просмотру, но ее можно установить на Nexus 5/5X. На эти устройства также можно ставить



➤ Jolla C доступен только для разработчиков, а SailfishOS 2.1.1 — для всех.

Приложения для поколения GNU

Использование спецпрошивки-ROM позволит старым телефонам работать с самыми свежими технологиями от Проекта открытого Android (Android Open Source Project, AOSP). Например, нам очень нравится, что на исходном 2013 Moto G (даже

тот, у которого хранилище 8 ГБ и нет слота SD) прекрасно запускается даже самая последняя LineageOS 14.1. Их можно найти на eVau дешевле чем за £30, что делает



их идеальными телефонами для одноразового потребления. Спецпрошивки также обходят любые искусственно раздутые производителем программы или ограничения режима модема. Для борцов за свободу ПО есть еще один аспект — возможность обойтись без проприетарного ПО, особенно от Google.

В приложениях FOSS нет недостатка: превосходное приложение *K-9 Mail* умеет делать то же, что и *Gmail*, а *Signal* позаботится обо всех ваших зашифрованных сообщениях, не отдавая их во власть Facebook. Браузер *Brave* вполне заменяет *Chrome*, как и *Chromium* и *Firefox for Mobile*. Ряд приложений используют данные OpenStreetMap (см. <http://wiki.openstreetmap.org/wiki/Android>), но на данный

момент ни одно из них не дотягивает до функциональности и гладкости Google Maps. Если вы используете открытые социальные сети, вроде *Mastodon* или *GNU Social*, то с их учетными записями может работать приложение *Twidere*, а @POTUS — отличное приложение для обращения к жителям Сети и жалобам на СМИ. Было бы упущением не упомянуть отличный репозиторий приложений *F-Droid*, где найдется больше приложений FOSS, чем может выдержать ваш акселерометр.



MaruOS, которая пытается возродить мечту о конвергенции, превращая Android в Debian при подключении внешнего дисплея.

База Android с открытым кодом (AOSP) позволила разработать нескольких индивидуальных прошивок, но многие умелые пользователи жаждут для своих телефонов чего-то более чистого. В конечном итоге их лучшей опцией может оказаться postmarketOS, но пока этот проект находится во младенчестве. Только восемь устройств «поддерживаются» (т. е. работают как минимум сеть USB и вывод графики) — и ни одно из них не в состоянии делать звонки. ОС основана на крошечном Alpine Linux и намерена дать телефонам 10-летний жизненный цикл.

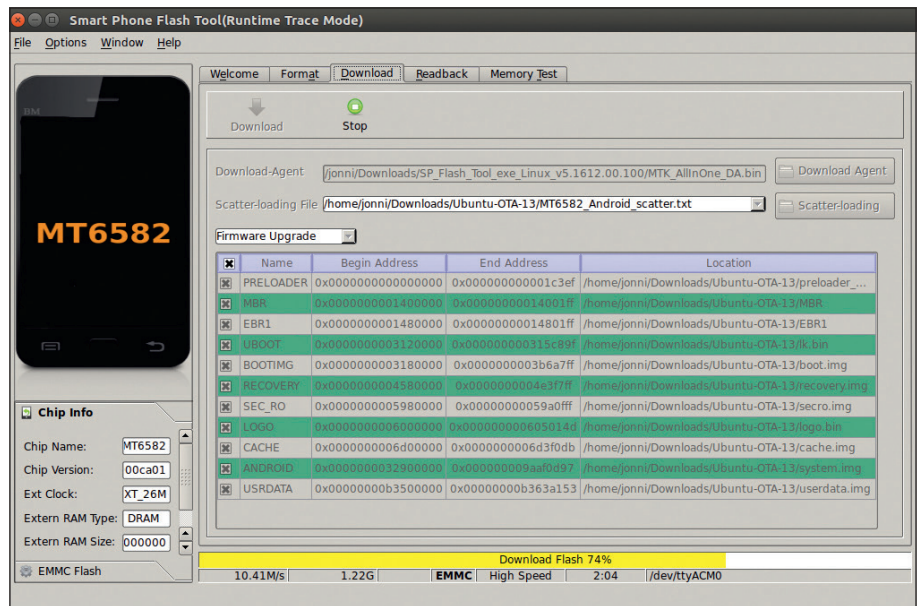
UBports

В 2015 г. многих сторонников открытого кода взволновал выход BQ Aquaris 4.5, первого телефона Ubuntu. И еще большее количество людей пришли в восторг от последовавшего за ним спустя год планшета Aquaris M10. Однако позднее их постигло разочарование, когда основатель Ubuntu Марк Шаттлворт [Mark Shuttleworth] объявил о прекращении работы над Ubuntu Touch, Unity 8 и всеми связанными с ними аппаратными проектами. Никто не возликует, услышав, что его уникальное, пусть даже и экзотическое устройство осталось без поддержки.

Однако не всё еще потеряно. В 2015 г. Мариус Грипсгард [Marius Gripsgard] основал проект UBports (<https://ubports.com>), цель которого — донести Ubuntu Touch до максимально возможного количества устройств. По мере того, как разработчики



➤ На нашем Aquaris 4.5 Ubuntu Touch работал ровно и выглядел стильно. Хотя у нас были проблемы с приложением OpenStore.



➤ Иногда, чтобы перенести новые ПЗУ на ваш смартфон, приходится браться за сомнительные инструменты; но вы просто зажмурьтесь и думайте о родине.

и защитники свободного ПО присоединялись к проекту, увлеченные его духом, сообщество росло очень быстро. Пессимисты могли решить, что раз уж в Canonical отказались от своих попыток создания мобильного устройства, то UBports прекратит существование; но на деле произошло обратное. Они взяли базу кода Ubuntu Touch и намерены ее развивать, поддерживать и не давать ей согнуться, более чем достаточно.

В июне 2017 г. UBports анонсировали свое первое обновление, OTA-1, доступное для всех устройств, изначально поддерживаемых Ubuntu Touch (телефоны и планшеты Aquaris, телефоны Meizu и Google Nexus 4 и 7), а также некоторых устройств, которые UBports поддерживали до объявления Canonical (Fairphone 2, Nexus 5, 7 и 10 и OnePlus One). Телефоны Aquaris и Meizu сейчас классифициру-

ют и кнопку громкости + и выберите fastboot в меню; скачать образ аварийного восстановления; и скопировать и вставить длинную команду. Однако в этом мире всё непросто — и дело не в том, что инструкции неправильные; но они работают только в том случае, если на вашем телефоне запущен оригинал Ubuntu Touch OS.

Если же вы установили на свое устройство Android, то раскладка разделов будет неправильной, и образ восстановления вы залить не сумеете. Чтобы продолжить, следует вернуть исходную ОС (в нашем случае — OTA-14), которую можно скачать вместе с инструкциями (на испанском) с <http://www.mibqvyo.com/descargas/categorias/aquaris-e4-5-ubuntu-edition/>.

Это процесс нетривиальный, поскольку инструмент перепрограммирования телефона SP требует наличия старой *libpng12*; но ее можно най-

ти, хорошенько погуглив. И как только мы ее установили, всё уже пошло как по маслу; и мы установили OTA-14, после чего смогли последовать простым инструкциям UBports и на-

OTA-1 — важный шаг для телефонов, разрабатываемых сообществом.

слаждаться возрожденной Stable OTA-1.

Хотя список новых функций в этом релизе может показаться весьма скромным (поддержка экспериментального AGPS, приложение приветствия и новый магазин приложений OpenStore), всё же OTA-1 — важный шаг для телефонов, разрабатываемых сообществом, и владельцев соответствующих устройств, которые больше не чувствуют себя покинутыми.

Сам интерфейс Ubuntu Touch остается неизменным — к нему, как и прежде, надо привыкать; но трудно не отдать должное его инновационному подходу и оригинальному перемещению краев. Правда, если вы перешли с Android, то будете безнадежно тыкать в низ экрана в поисках несуществующей панели кнопок. LXF

ются как устаревшие, и их ждут только обновления безопасности. Однако это безусловно лучше, чем ничего.

Linux Format повезло найти завалевшийся Aquaris 4.5 (с кодовым именем Krillin, на жаргоне Ubuntu Phone), и мы решили задокументировать наши попытки получить на него OTA-1. Судя по инструкциям на сайте, этот процесс должен быть довольно прост. У нас был выбор: использовать общецелевой Magic Device Tool (MDT, это обертка для программ, написанных для конкретного оборудования) от Marius или напрямую использовать *ubuntu-device-flash*. Во втором случае всё, что требовалось, если верить инструкциям на <https://ubports.com/page/fs-flash-phone>, это загрузить наше устройство в режиме fastboot (удерживайте кнопку включения

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

Linux  center

Наши эксперты помогут вам с любым приложением Linux!



ЕВГЕНИЙ БАЛДИН
Подтвердивший
свою квалификацию
физик.

Гиганты свободного ПО

Если я видел дальше других, то потому, что стоял на плечах гигантов.
Приписывается Исааку Ньютону

Одно из идеологических преимуществ СПО — возможность без каких-либо затратных обязательств «прислонить» свои проекты к уже готовым. Особо приятно, если подставкой для «рывка в будущее» служит качественное и всеобъемлющее решение. Оно, конечно, не обязательно, ведь если костыли из палок и веревочек, спаянные абы как без всякого плана, решают задачу, то это прекрасно! И тоже имеет свое очарование. Малый размерный класс был, есть и будет основным источником «инноваций», но нельзя и отрицать красоту и мощь «динозавров». Особенно потому, что, в отличие от вымерших «королей мезозоя», СПО, даже сложное, можно улучшать. Порой методом «волшебного пенделя».

В конце июля 2016 г. в корпоративном блоге компании Uber появилась статья о причинах перехода с PostgreSQL на MySQL (то, что чуть ранее уже был обратный переход, к делу не относится), с претензиями сотрудников Uber к работе с PostgreSQL: к эффективности записи, репликация и трудности с обновлением «на лету». Сообщество разработчиков PostgreSQL и иже с ним сначала возмутилось, заявило, что Uber просто «не умеет готовить», но затем признало проблемы и стало их решать. Ответы на заданные вопросы можно видеть в официальном пресс-релизе PostgreSQL 10. Хорошо, когда «динозавры» не распускаются.
E.m.Baldin@inp.nsk.su



В этом месяце вы научитесь...

- Сцеплять команды** 46
По мнению **Ника Пирса**, лучше одна длинная команда, чем много-много коротких, и он активно употребляет цепочные операторы.
- Сочинять музыку** 48
Шашанка Шарму влечет карьера композитора, и вместе с ним и *LMMS* вам тоже удадутся собственные чарующие мелодии.
- Познавать процессы** 52
Дмитрий Пантелеичев вникает в отведение процессам памяти, отыскивает процессы по всяческим критериям... и покушается на их убийство.
- Спасать файлы** 56
С хирургической точностью **Нейт Дрейк** выкраивает осмысленные куски из испещренного нулями и единицами цифрового полотна, пользуясь *scalpel*.
- Пробовать ПО** 58
Хотя **Бобби Мосс** любит поэкспериментировать с новыми программами, его компьютеру не угрожает крах, благодаря применению Vagrant.
- Ставить ОС на телефон** 62
LineageOS — идеальный вариант для тех, кто желает полновластно управлять своими телефонами и планшетами, уверяет **Алекс Кокс**.
- Записывать подкасты** 66
Говорят, что лучше увидеть, чем услышать, и тем не менее подкасты не сдают позиций. **Адам Оксфорд** учит обращаться с *Audacity*, оберегая ваши уши от неприятных щелчков.
- Мерить температуру** 70
Теплолюбивый **Максим Черепанов** контролирует температуру в своем Послушном Доме — на всякий пожарный случай. В смысле, при пожаре можно и сигнализацию включить.

АКАДЕМИЯ КОДИНГА



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

- Загадка LXF** 74
Михалис Цукалос занялся разнообразными сортировками, экономно поставив самому себе ограничение на допустимый объем памяти.
- Без сервера** 78
Зачем он нужен, этот сервер? — задумался **Дэн Фрост**, и перебрался на облачные приложения. Развертывание за несколько строк кода!

Terminal: Скова- НЫ ОДНОЙ ЦЕПЬЮ

Ник Пирс расскажет, как с помощью цепочных операторов связать несколько команд — причем без спецтехники.



Наш эксперт

Ник Пирс балуется с компьютерами уже более 30 лет, из них последние 10 — с Linux.

Суть связывания команд проста: берется несколько команд Linux, и команды объединяются в одну строку. Цель ясна: команды будут выполняться одна за другой, а значит, можно запустить длительный процесс, и устройство будет выполнять его само, без вашего участия. Например, вот такой цепочкой можно проверить обновления и установить их:

```
$ sudo apt-get update && sudo apt-get upgrade
```

Строка &&, разделяющий две команды, называется оператор; в Linux их много, и все они позволяют контролировать поведение каждой команды на основе предыдущей.

В этом месяце в нашем уроке про командную строку мы покажем, как работать с наиболее распространенными операторами.

Последовательные операторы

Начнем с оператора точки с запятой (;). Если разделить им команды в одной строке, они будут выполняться последовательно, одна за другой:

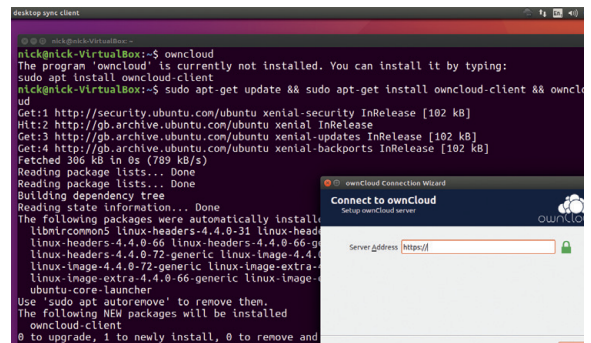
```
$ sudo apt-get update ; sudo apt-get upgrade
```

Таким способом можно поставить в очередь любое количество команд. Оператор ; выполняет каждую команду независимо от того, что происходит с предыдущей командой, а это не всегда желательно. В приведенном выше примере мы хотим, чтобы `sudo apt-get upgrade` выполнялась в том случае, если команда `sudo apt-get update` успешно завершилась.

Здесь в дело вступает оператор &&, используемый вместо оператора ;. Оператор && также известен как оператор И. Он означает, что вторая и последующие команды будут выполняться только в том случае, если предыдущие команды завершились успешно.

```
$ sudo apt-get update && sudo apt-get install приложение && приложение
```

Итак, *приложение* в нашем примере запустится только в том случае, если оно было установлено, а установлено оно будет только при успешном запуске `sudo apt-get update`. Т.е. если будет получено сообщение, что команда обновления используется другим процессом, установка приложения и все последующие команды будут отменены.



➤ При использовании оператора && последующие команды выполняются только если предыдущая выполнена успешно.

Следующий пример прозванивает удаленный компьютер, прежде чем попытаться к нему подключиться. Если не выполняется ping, то ssh не запускается:

```
$ ping -c1 192.168.0.3 && ssh user@192.168.0.3
```

Желая, чтобы команда выполнялась только при сбое первой команды, используйте оператор || (ИЛИ). Он велит второй команде запускаться только в том случае, если вывод первой имеет статус 1 или выше (отказ), а не 0 (успех):

```
$ apt-get update || echo Вы забыли вставить sudo
```

Конечно, && и || можно объединять. Это как создание выражения ЕСЛИ... ТО... в программировании:

```
$ apt-get update && sudo apt-get upgrade || echo Эй, вы забыли sudo!
```

Если нужно создать более сложные цепочки команд, понадобится оператор комбинации команд. Он позволяет группировать связанные друг с другом команды внутри символов {}, тем самым распознавая их как единую команду в контексте других.

В следующем примере сначала будет проверяться наличие файла `linux.txt`. Если файл существует, обрабатывается первая команда `echo`; если нет, будут выполняться одна за другой команды внутри операторных скобок {}:

Скорая помощь

Еще есть оператор !, т.е. «НЕ». С его помощью можно исключать определенные элементы, например, удалить из текущей папки всё, за исключением файлов `.txt`:
`$ rm -r!(*.txt)`

Выполнять или не выполнять?..

Разобраться, какой оператор за что отвечает, бывает трудно. Следующая таблица может посодействовать вам в выяснении, будет ли выбранный вами оператор выполнять то, чего вы от него ожидаете. Таблица любезно предоставлена www.linuxnix.com.

Описание	;	&&			&
Выполняется ли вторая команда, если первая команда выполнена успешно?	Д	Д	Н	Д	Д
Выполняется ли вторая команда, если первая команда потерпела неудачу?	Д	Н	Д	Д	Д
Будет ли вторая команда выполняться только в зависимости от состояния первой команды?	Н	Д	Д	Н	Н
Будут ли команды выполняться параллельно?	Н	Н	Н	Н	Д

Всё о конвейерах

Конвейеры — специальные операторы для перенаправления вывода одной программы в другую команду для дальнейшей обработки, как правило, для получения иной формы вывода. Конвейеры образуются с помощью символа `|`; на большинстве клавиатур его можно найти, удерживая `Shift` при нажатии обратной косой черты (`\`).

В следующем примере результат команды листинга каталога `ls` передается команде `less`, чтобы получить более удобный для просмотра формат:

```
produce a more$ ls -l | less
```

А вот так можно с умом воспользоваться командой для подсчета слов `wc` (word count) с флагом `-l` (указать количество строк), чтобы узнать количество объектов в указанном каталоге:

```
$ ls ~/Documents | wc -l
```

Но конвейеры способны не только служить для перенаправления вывода из одной программы на ввод другой программы.

В следующем примере загружается видео с YouTube, а затем оно пропускается через `ffmpeg`, чтобы выдернуть только аудио в формате MP3:

```
$ URL=http://www.youtube.com/watch?v=a1AnTi1x2QM
$ youtube-dl $URL -q -o - | ffmpeg -i - audio.mp3
```

Наконец, можно комбинировать несколько конвейеров — аналогично тому, как вы можете изготавливать сложные цепочки, составляя их из нескольких операторов:

```
$ youtube-dl $URL -q -o - | ffmpeg -i - audio.mp3 && play
audio.mp3
```

С помощью операторов конвейера можно скачать видео, конвертировать в аудио и, наконец, проиграть его в Terminal.

```
$ test -f ~/Documents/linux.txt && echo linux.txt существует! ||
{echo linux.txt не существует - создадим.; touch ~/Documents/
linux.txt; }
```

Оставьте пробел между `{` и командой и не забудьте поставить `;` в конце, чтобы вернуться в оболочку после выполнения последней команды.

Порядок решает всё

Иногда бывает непонятно, в каком порядке поставить команды. Операции всегда выполняются слева направо, независимо от оператора. Поскольку `&&` и `||` имеют одинаковый приоритет, первой выполняется та, что стоит раньше. Рассмотрим следующий пример:

```
$ true || echo aaa && echo bbb
```

Тут выполняется первый оператор. Поскольку он имеет значение `true`, вы можете решить, что вторая и третья команды будут проигнорированы. На самом деле, игнорируется только первая команда `echo`, а `bbb` всё равно выведется. Чтобы этого не случилось, надо заключить в скобки те команды, которые вы хотите сгруппировать:

```
$ true || (echo aaa && echo bbb)
```

Вот реальный пример из практики:

```
$ ( sudo apt-get update && sudo apt-get upgrade ) || echo
Обновление неуспешно
```

Тогда будет отображаться сообщение **Обновление неуспешно**, если любая из команд завершится неудачей, а не только `apt-get upgrade`.

Если вы начинаете путаться в операторах, можете использовать собственные команды `Bash` — `IF`, `THEN` и `ELSE`. Эти команды должны замыкаться строкой `fi`:

```
$ if ls *.txt; then echo Мы нашли текстовые файлы; else echo
Нет, текстовых файлов нет!; fi
```

Эти команды лучше всего подходят для простых условных операторов, таких, как обсуждались выше.

Запуск процессов параллельно

Все рассмотренные нами операторы запускают команды последовательно, т.е. одну за другой. Это занимает терминал, заставляя вас дожидаться завершения команды, прежде чем вы сможете перейти к следующей. Но иногда бывает нужно запустить команды в многозадачном режиме — другими словами, параллельно. Можно открыть еще одно окно терминала, если вы находитесь в рабочем столе, а можно применить оператор `&`, который велит команде,

скрипту или процессу работать в фоновом режиме, освобождая командную строку для других задач. Чтобы сделать это, добавьте оператор `&` в конец вашей команды, например:

```
$ sudo apt-get upgrade &
```

Команда выведет нечто вроде

```
[1] 3215
```

Выведенное число указывает на ID процесса, присвоенный этой команде, а сама команда теперь работает в фоновом режиме, освобождая терминал для других команд. Тот же символ `&` может при необходимости применяться для параллельного выполнения цепочки команд. Следить за процессом можно с помощью такой команды:

```
$ ps -f p 3215
```

Подставьте вместо 3215 идентификатор процесса своей команды. Нажав `Enter`, вы увидите обновление статуса. `Done` [Выполнено] указывает, что он завершен успешно, а `Stopped` [Обычно] обычно означает, что он завершился неудачей. Пока процесс выполняется, вы можете работать с другими командами.

Помимо параллельного запуска одиночных команд, вы можете также объединить несколько команд в одну строку, и все они будут выполняться одновременно в фоновом режиме:

```
$ ping -c1 google.com & ping -c1 linuxformat.co.uk &
```

Здесь вывод обеих команд будет отображаться вперемешку, поскольку они выполняются параллельно. Соблюдайте осторожность при выборе команд для параллельной работы. Нет смысла выполнять `sudo apt-get update` и `sudo apt-get upgrade` параллельно, поскольку для запуска второй необходимо, чтобы первая завершилась успешно. Таким образом, если команда или инструмент, который вы хотите запустить, не предназначена для выполнения в фоновом режиме, подумайте дважды. **LXF**

Можно создавать сложные строки команд, подобраные по неким критериям, для выполнения требуемых действий.

» Подпишитесь на печатную или электронную версии на www.linuxformat.ru/subscribe!

LMMS: СОЧИНИТЕ СВОЮ МУЗЫКУ

Шашанк «Принц Премудрости» Шарма объявил, что он достоин музыкальной темы; с LMMS вы тоже можете создать собственный трек.



Наш эксперт

Шашанк Шарма — выступающий в суде юрист из Дели и завязанный пользователь Arch Linux. Он постоянно охотится за технологическими экспонатами.



При работе с такими обширными программами, как LMMS, лучше не спешить затевать проект. Ответьте достаточно времени на знакомство с интерфейсом, и вы приметесь создавать музыку очень быстро.

Со всеми этими окнами, открытыми одновременно, интерфейс LMMS кажется жуткой мешаниной. Ограничьте свою работу одним окном за раз, и всё начнет обретать смысл.

З а счет опасности нарваться на очередную двоичную шутку, и не погружаясь в теорию музыки, давайте согласимся с тем, что есть два типа музыки: все ваши любимые музыканты, будь то *Led Zeppelin*, *Мадонна*, *The Beatles*, *Steps* и все остальные — это один тип, вокальный. Другой тип, объединяющий исключительно инструменты, по большей части используется для фильмов и телевидения. Такие произведения, как *Терминатор*, *Стартрек*, *Игра Престолов* и *Акулий торнадо* используют инструментал для создания основной темы, обычно звучащей в начальных или финальных титрах. С помощью LMMS, где имеется редактор песен, редактор битов, вход для пианино, поддержка MIDI-клавиатур и ряда инструментов, и всё это работает по умолчанию, вы и сами сможете создать собственную запоминающуюся музыкальную тему.

Ранее известный как *Linux MultiMedia Suite (LMMS)*, проект перекрестил себя в свои же собственные инициалы, только став кросс-платформенным, и теперь доступен в Linux, Windows и Mac. Хотя это и не единственная рабочая станция цифрового аудио (Digital Audio Workstation, DAW) с открытым кодом, LMMS давно зарекомендовал себя как один из лучших. Одной из причин его популярности является сходство LMMS с проприетарной альтернативой — *FL Studio*.

Большинство дистрибутивов Linux держат LMMS в своих программных репозиториях, и его легко установить с помощью `sudo apt install lmms` или `su -c "dnf install lmms"`, в зависимости от вашего дистрибутива. Сайт проекта предлагает инструкции по установке

из репозитория для openSUSE, Arch, Mageia, Mint и других. После установки вы можете запустить его из Sound & Video — или сходных по названию — меню приложений на вашем рабочем столе.

Linux Mint 18.2, основа нашей установки LMMS, поставляется с самым последним стабильным релизом LMMS — версией 1.1.3. Ваша установленная версия программы может слегка отличаться, если ваш дистрибутив содержит более старую версию или один из более новых нестабильных релизов. Независимо от внешнего вида, вы сможете следовать нашим инструкциям по созданию собственных мелодий. К сожалению, природа LMMS такова, что если бы мы вдавались в детали по каждому аспекту его структуры, то это был бы уже не учебник, а своего рода вводная статья. Наша цель — помочь вам создать музыку, и если вам покажется, что мы лишь бегло коснулись определенных элементов, тут уж ничего не поделаешь. Если вам нужно более глубокое понимание множества компонентов LMMS, обратитесь ко врезке *Ресурсы*.

Богатый интерфейс

При первом запуске вам сообщат, что директории `~/lmms` не существует и что она будет создана. Нажмите OK, и проект автоматически создаст директории `~/lmms` и заполнит ее такими поддиректориями, как **presets**, **projects** и **samples**. Эти пустые директории впоследствии будут использоваться для хранения ваших произведений, когда вы начнете работать с LMMS.

Тем, кто уже знаком с DAW, интерфейс LMMS покажется привычным и функциональным, но если вы обратились к LMMS как



Ресурсы

Графическая природа и изобилие пресетов и сэмплов может создать впечатление, что писать мелодии в *LMMS* очень просто — и так оно и есть.

А вот хорошие, запоминающиеся мелодии создать, к сожалению, уже труднее. Если у вас хороший слух, вы сумеете далеко продвинуться с *LMMS* даже без глубоких знаний в теории музыки. Но вам всё-таки потребуется немало помощи в процессе. ЧаВо (FAQ) на сайте — хорошая отправная точка.

Здесь вы сможете понять, как включать пресеты в ваш первый проект.

Весьма полезным может оказаться *LMMS: A Complete Guide To Dance Music Production [LMMS: Полное руководство по созданию танцевальной музыки]*, опубликованное в 2012 г. PacktPub, хотя сейчас оно уже устарело; но лучше всего сначала изучить справочник и руководство *Getting Started* на сайте. Проект также предлагает список

видеоуроков на YouTube. Они включают введение в Piano Roll, основы FX Mixer, использование Automation Editor, создание разных жанров музыки, таких как Dubstep, House, Trance и т. д., а также руководство по теории музыки.

Форумы обеспечивают быстрый доступ к мудрости сообщества и размещают дополнительные руководства от пользователей. Это отличный ресурс, независимо от степени вашего знакомства с *LMMS*.

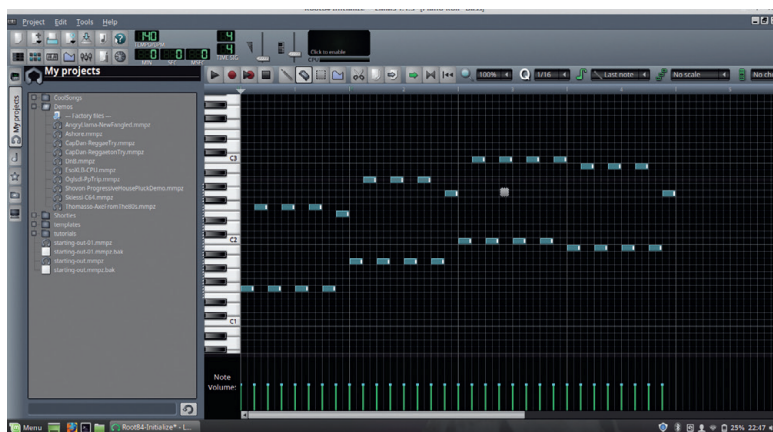
к средству выражения ваших творческих способностей, на первый взгляд интерфейс может отпугнуть. Но не волнуйтесь, мы поможем вам разобраться со множеством кнопок и со всякими окнами. Популярный интернет-мем 'Keep Calm and Carry On [Будь спокоен, продолжай]' при работе с *LMMS* сослужит вам хорошую службу.

Основная строка меню сверху может применяться для обычных операций, таких как открытие и сохранение нового файла проекта, отмена действия и повтор ввода, управление плагинами и получение подсказки онлайн. Второй ряд содержит кнопки операций с файлами, таких как создание нового проекта, открытие существующего проекта, сохранение текущего, и т. д. Наведите мышью на кнопку, и удобная подсказка объяснит вам ее функцию.

Кнопки в следующем ряду служат для переключения между разными элементами основного окна. По умолчанию *LMMS* отображает окна Song Editor, Beat+Bassline Editor, FX-Mixer и Controller Rack. Вы можете использовать кнопки в третьем ряду для отображения/сокрытия этих окон и других, таких как Piano Roll. Для каждого трека или мелодии, которую вы хотите создать с помощью *LMMS*, два ряда также имеют кнопки управления верхнего уровня, которые можно использовать для задания темпа, громкости, тона, вставки временной отметки и т. д.

И, наконец, боковая панель слева, с шестью кнопками. Нажатие на каждую кнопку включит соответствующую дополнительную боковую панель. Каждая кнопка или вкладка на левой боковой панели скрывает определенный тип ресурсов. Так, первая кнопка отображит плагины Instruments. Отсюда вы можете выбрать желаемый инструмент для создания вашего трека. Похожим образом, другие кнопки включают My projects, My samples, My presets, и т. д. Вы будете использовать эти ресурсы для создания своих мелодий.

Основной интерфейс содержит четыре окна. Song-Editor — главный секвенсор для вашей музыки, и он описывает структуру вашего произведения. Окно Beat+Bassline Editor используется для настройки различных повторяющихся звуков и битов. С помощью FX-Mixer можно вносить изменения в вашу песню и отдельные ее элементы, добавляя разные эффекты. И наконец, окно Controller Rack используется для внесения математических изменений в элементы вашей музыки. Два последних окна помогут вам внести в вашу музыку тонкую настройку, а для создания мелодий



вам понадобится работать только с Song и Beat+Bassline Editor; остальные окна предназначены для тех, кто изучал теорию музыки и способен отличить ми от до и диэзы от бемолей.

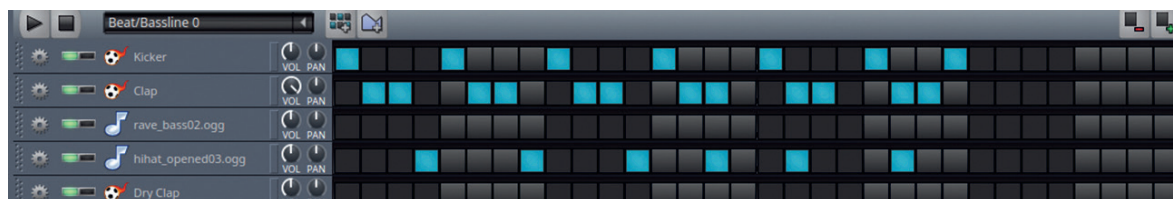
Биты и такты

Все треки в Beat+Bassline Editor по умолчанию имеют 16 битов, представленных 16-ю серыми клетками. Не углубляясь в теорию музыки, 16 битов известны как шестнадцатые доли. Два бита на шестнадцатой доле равны одной восьмой доле, а четыре бита — четверти. По мере продвижения в вашем предприятии по созданию музыки, знание долей поможет вам лучше изучить полюбившиеся музыку и песни, и потом вы сможете создавать собственные мелодии.

Биты в треке разделены на четыре шага, имеющих соответствующую подсветку более темными и более светлыми оттенками серого. Вы можете включить шаг или бит, нажав на нем левой кнопкой. Чтобы изменить громкость любого шага, наведите на него мышью и используйте колесо прокрутки мыши для увеличения или уменьшения громкости. При полной громкости шаг оказывается ярко-голубым, а средняя голубизна указывает на средние уровни. Для каждого бита в треке громкость можно задать индивидуально.

» При работе с Piano Roll пространство между нотами почти так же важно, как звук инструмента.

»



» Один из самых популярных паттернов битов известен как «Прямая бочка 4/4 [Four on the floor]» и используется в диско-, электронной и танцевальной музыке.

» Подпишитесь на печатную или электронную версии на www.linuxformat.ru/subscribe!

Скорая помощь

Добавьте ударные инструменты (на которых играют, ударяя рукой или палочкой) в Beat+ Bassline Editor, а не в Song Editor, если вы захотите, чтобы этот паттерн повторялся. Кроме того, таким образом им намного проще управлять.

Создание ритма

Сейчас мы будем работать только с Song Editor и Beat+ Bassline Editor, так что давайте закроем другие окна. Для этого нажмите на соответствующие кнопки во втором ряду сверху.

Song Editor используется для неповторяющихся событий, таких как мелодические линии. События, созданные в Song Editor, можно копировать и повторять, чтобы дополнить мелодию. Завершенная песня является компиляцией всех элементов в Song Editor, а также в Beat+ Bassline Editor. На языке *LMMS* каждый элемент называется треком, так что ваша финальная композиция будет состоять из нескольких разных треков.

Вся эта заумная теория не особенно важна, когда начинается процесс создания музыки, но всё же способна помочь вам, когда вы начнете глубже вникать в работу *LMMS* для создания более сложных и длинных треков. По умолчанию Song Editor уже имеет четыре трека: Triple Oscillator, Sample Track, Beat/Bassline 0 и Automation Track.

Чтобы начать, нажмите кнопку Samples на левой боковой панели. Вы можете выбрать различные звуки, которые хотите включить в вашу музыкальную партитуру. Нажмите на любой звук, чтобы прослушать сэмпл. Определив тот, с которым вы собираетесь работать, дважды щелкните по нему и добавьте в Beat+ Bassline Editor.

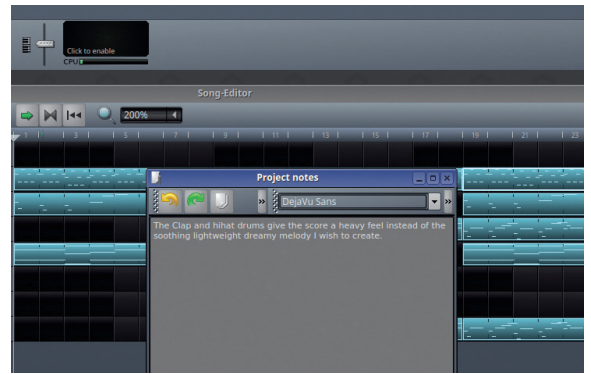
Для нашего первого бита давайте добавим пару пресетов и сэмплов. Выберите Presets на левой боковой панели и нажмите Kicker > Clap. Если вам нравится звук, дважды нажмите, чтобы добавить его в Beat+ Bassline Editor. Похожим образом мы добавили *hihat_opened03* и *rave_bass02*.

Теперь, когда инструменты на месте, время создавать мелодию!

Если вы еще не работаете с *LMMS* в полноэкранном режиме, советуем вам сделать это. Это облегчит задачу совершенствования вашего саундтрека. Определите инструменты/звуки, которые вы добавили в Beat+ Bassline Editor. Рядом с каждым треком вы найдете ряд окон.

Нажмите на серую клетку, чтобы включить звук бита, который хотите воспроизвести. Похожим образом можно включить звук, нажав на клетки для разных треков. Внеся в треки достаточно битов, нажмите на кнопку Play в верхней левой части Beat+ Bassline Editor, чтобы прослушать созданный вами бит. Если услышанное вам не понравится, внесите изменения и придерживайтесь их. Научить этому нельзя — пусть вас ведет ваше творческое начало. В качестве примера обратитесь к паттерну, показанному в одном из множества изображений здесь. Начните с этих паттернов, а затем перейдите к собственным композициям. Хотя для первого бита мы ограничились всего четырьмя инструментами, вы, конечно, можете при желании добавить больше.

Стоит также отметить, что если вы намерены создать долгую мелодию, наличных клеток по умолчанию может не хватить.



➤ В *LMMS* встроен еще и текстовый редактор. Он пригодится для создания списка всех инструментов и настроек, которые вы уже внедрили в свой текущий проект.

Количество шагов можно увеличить, нажав на кнопку Add Steps в верхней правой части Beat+ Bassline Editor. Помните, что каждая клетка обозначает один бит, т.е. один экземпляр проигрывания любого звука в треке. Если вы хотите, чтобы звук повторялся, надо нажать на ряд клеток, в зависимости от паттерна, который вы хотите создать.

Бывает, что вам хочется воспроизводить во время песни определенные треки, а определенные звуки — только в течение коротких периодов. Для такого сценария в вашем Song Editor должно быть несколько экземпляров Beat+ Bassline. Создав один интервал бита, называемый последовательностью [sequence], нажмите на кнопку Add beat/bassline в Song Editor. В список треков добавится новый трек Beat+ Bassline. Переключаться между разными редакторами Beat+ Bassline можно по нажатию на название в Song Editor. Вы также можете использовать выпадающее меню последовательностей в левой верхней части окна Beat+ Bassline Editor для переключения между последовательностями.

По умолчанию *LMMS* называет треки Beat/Bassline 0, Beat/Bassline 1, и т.д. Дважды щелкните по имени трека в Song Editor, чтобы переименовать его. Например, если ваш второй трек Beat/Bassline используется для определения музыкальной партитуры для средней последовательности вашего саундтрека, вы можете переименовать его в **Bassline-middle**. Подобным образом можно поступить с последовательностями Bassline для начала и окончания трека. Это также упростит идентификацию и индивидуальную работу над частями трека, требующими изменений.

После создания мелодии Bassline вам надо внедрить ее в структуру вашего саундтрека. Для этого вы должны следовать той же процедуре и нажимать на клетки рядом с треком Beat+ bassline в окне Song Editor. Весь ваш паттерн Beat+ Bassline будет играть для каждого бита в окне Song Editor.

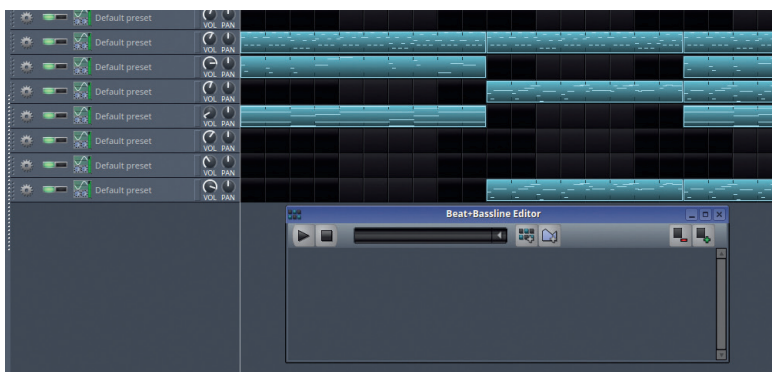
Нажмите на кнопку Play в окне Song Editor, чтобы услышать, как звучат разные треки, воспроизведенные в последовательности. Возможно, вы захотите немного переместить элементы, чтобы найти идеальный баланс между разными треками.

Настройка битов

Настройки по умолчанию темпа и уровней звука для разных инструментов и сэмплов в Beat+ Bassline Editor могут и не работать для каждого типа саундтрека. К счастью, это легко изменить. Но, как и всё остальное в *LMMS*, выяснение, что работает лучше всего, требует времени и практики, поскольку одна настройка не будет работать для всех инструментов.

Темп определяет скорость воспроизведения вашего трека в Song Editor. По умолчанию это 140, что может показаться

➤ То, что создавать бит с помощью Beat+ Bassline Editor легко, вовсе не означает, что вы должны использовать его в каждом своем произведении.



➤ Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

не в меру неистовым, особенно если вы хотите создать медленный и душевный саундтрек. Дважды щелкните по окну Tempo, чтобы открыть диалоговое окно. В нем можно задать значение между десятью и 999. Индивидуальные темпы для разных последовательностей, которые составляют всю песню, задавать нельзя. Например, если в вашем саундтреке имеются три последовательности Bassline, вы не можете установить индивидуальный темп для каждой из них.

Помимо темпа, можно определить индивидуальную громкость для каждого инструмента и сэмпла, использованного в последовательности Bassline. Вы увидите в Beat + Bassline Editor кнопку VOL для каждого трека. Нажмите на нее и передвигайте мышью для определения нужной громкости. Если значение по умолчанию слишком велико, изменяйте громкость на 50 процентов, а то и ниже, пока не найдете значение, которое будет устраивать ваш слух и соответствовать широкому контексту последовательности.

С помощью Beat + Bassline Editor вы можете создавать биты с точностью только до одной шестнадцатой. Если вам нужна большая степень контроля над нотами, например, изменение тона, длительности и громкости, воспользуйтесь Piano Roll. Двойное нажатие на любом треке в Beat + Bassline Editor откроет его в Piano Roll. Если вы уже создали трек, то Piano Roll будет заполненным.

Хотя даже новичок может с успехом создать мелодию, используя всего лишь Beat + Bassline Editor, использование Piano Roll требует глубокого понимания стандартной настройки струн EADGBE. Или же у вас должен быть отличный слух.

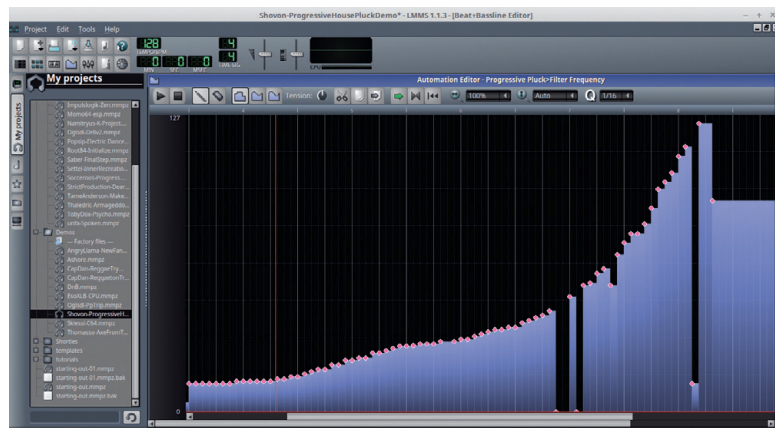
Piano roll в Beethoven

Beat + Bassline Editor поможет скомпоновать ритмичный танцевальный мотивчик, но если вы хотите создавать мелодии, стоит подумать об использовании Piano Roll.

Вначале определите инструмент, который хотите использовать, и перетащите его в окно Song Editor. Затем дважды нажмите на серую клетку этого трека, чтобы открыть инструмент в окне Piano Roll. При работе с этой функцией не забывайте оставлять промежутки между нотами, иначе ваша композиция будет звучать слишком быстро и жестко.

Кнопки на верхней панели окна Piano Roll используются для ввода нот. Нажмите на кнопку Draw mode — теперь вы можете вводить ноты, нажимая на клетчатые окна в основном окне. Для каждой введенной ноты можно контролировать громкость, с помощью нижней панели окна Piano Roll. Если длительность ноты по умолчанию слишком велика или мала, перемещайте мышью вправо до конца этой ноты, и когда курсор мыши превратится в двустороннюю стрелку, нажмите и перетаскивайте влево или вправо для увеличения или уменьшения длительности ноты.

Как обычно, нажмите на кнопку Play вверху слева, чтобы проверить ваши ноты и внести изменения и дополнения, если требуется. Опять же, обратитесь к одному из паттернов, показанных



в изображениях, и сначала повторите его, чтобы разобраться, как работать с Piano Roll. Желая удалить ноту, переместите мышью вверх и щелкните правой кнопкой. Как это ни странно, комбинация клавиш Ctrl+z не способна выполнять операцию отмены в каждом из окон LMMS, и это служит причиной частых жалоб на приложение.

После создания пианинной мелодии и паттерна бита пора посмотреть, как они сочетаются. В окне Song Editor нажмите на серые клетки для каждого трека и затем нажмите Play. В зависимости от того, как они звучат вместе, вы можете решить внести небольшие поправки или вообще начать с самого начала.

Создание примера

Мы едва коснулись поверхности того, что можно сделать с LMMS. Этот отличный пакет для создания музыки является темой подробной книги, документации и разнообразных видеоруководств на YouTube.

Чтобы дать вам представление о том, какую музыку можно создавать с его помощью, LMMS прилагает ряд демонстрационных песен. Чтобы взглянуть на них, нажмите на кнопку My Projects на левой боковой панели. Теперь нажмите CoolSongs > Zakarra > 0084-Initialize.mmpz. Нажмите на кнопку Play вверху, чтобы прослушать трек.

Хотя песни предоставлены ради показа вам всестороннего характера LMMS и, будем надеяться, вдохновят вас, всё же есть вероятность, что они обескуражат тех, кто пока еще новичок в создании музыки. Если вы новичок, перейдите в раздел Shorties во вкладке My Projects и дважды нажмите на проект. Там песни короче, и с ними проще разобраться для понимания LMMS.

На LMMS Sharing Platform (<https://lmms.io/lsp>) вы найдете музыку, загруженную другими пользователями. Создав учетную запись, вы сможете делиться своими произведениями с другими пользователями. **LXF**

➤ Функция автоматизации LMMS может использоваться для автоматизации ряда регуляторов, таких как громкость — увеличение или уменьшение по мере воспроизведения трека.

Скорая помощь

Для ваших проектов нет управления версиями, поэтому надо создать собственное. Сохраняйте различные версии ваших песен, желательно после каждого изменения. Это позволит вам вернуться к предыдущей версии проекта, если ваш процесс пойдет не туда, куда вам хотелось.

Инструменты с открытым кодом для музыкантов

Экосистема Linux может похвалиться не одним дистрибутивом, созданным для поклонников искусства. Увлекаетесь ли вы играми или графическим дизайном, непременно найдется дистрибутив Linux именно для этой области, со множеством приложений и инструментов, разработанных в помощь вашим затеям. Начинающие музыканты и любители музыки оценят богатый выбор инструментов, утилит и приложений, доступных для Linux.

Как и LMMS, Ardour — это популярная DAW, но она менее дружелюбна к новичкам. Освоив

LMMS и добившись понимания процесса создания музыки, можете переходить к Ardour.

Если вы заинтересованы в изучении теории музыки для лучшего понимания того, что составляет мелодию, ваш инструмент — Phonascus. Эта программа содержит несколько уроков и упражнений, чтобы помочь вам определять ноты, гаммы, интервалы, аккорды и многое другое. Если вы уже работали с Piano Roll в LMMS, вы должны были заметить все эти опции на верхней панели. Выпущенное под GPLv3, это приложение доступно

в репозиториях всего нескольких дистрибутивов, но проект предлагает подробные инструкции по его компиляции вручную.

Для опытных музыкантов, желающих создавать электронные нотные записи, приложение LilyPond расширяет эстетику гравированных нот до компьютерных распечаток. LilyPond поддерживает разную музыку, в том числе вокальную и классическую. Проект также предлагает скрипт оболочки для установки, который можно запустить в любом дистрибутиве Linux.

Procps-ng: GNU/Linux и процессы

Дмитрий Пантелеичев считает эту тему неисчерпаемой, но всё-таки стремится приблизиться к максимальной ширине охвата.



Наш эксперт

Дмитрий Пантелеичев считает, что любую технологию надо изучать от простого к сложному, и каждый шаг закреплять практически на примерах.

Мы продолжаем знакомство с процессами в GNU/Linux и с программным пакетом *procps-ng*, начатое в прошлом выпуске журнала.

Программа *rtop*

При запуске каждого процесса операционная система выделяет ему адресное пространство, которое называется виртуальной памятью процесса. Процесс, обращаясь к памяти, имеет дело именно с этими адресами. Только потом адреса виртуальной памяти процесса транслируются в адреса физической памяти компьютера. Значит, один и тот же адрес виртуальной памяти разных процессов в реальности соответствует разным адресам физической памяти компьютера. Этим и обеспечивается изоляция процессов друг от друга. А программа *rtop* отображает карту виртуальной памяти процесса.

Давайте теперь посмотрим, как виртуальная память выглядит. Мы уже знаем, как запускать фоновые процессы, поэтому запустим фоновый процесс:

```
dd if=/dev/zero of=/dev/null &
```

После этой команды будет выведен PID запущенного фонового процесса, а затем приглашение *Bash*. В моем случае PID процесса — 3198. У вас, конечно, цифра будет другая.

Теперь вызовем программу *rtop* и в качестве параметра укажем PID нужного процесса. Если вызвать эту программу без ключей, то отобразятся только четыре столбца без заголовков. Поэтому давайте сразу добавим ключ `-X` (буква заглавная). Тогда формат будет чуть более расширенный: отобразятся шесть столбцов с заголовками.

```
rtop -X 3198
```

На компьютере с 64-разрядной архитектурой результат будет выглядеть как на рис. 1.

Проанализируем картину, которую мы увидели.

» Первый столбец, **Address** — это страница виртуальной памяти процесса, с которой начинается сегмент. Как видим, каждая страница памяти имеет адрес, обозначенный шестнадцатую шестнадцатеричными цифрами, хотя часть старших нулей может быть, для наглядности, пропущена. Поскольку каждая шестнадцатеричная цифра соответствует четырем битам, то каждый адрес можно представить в виде 64 двоичных цифр (16×4). Такова адресация у компьютеров с 64-битной архитектурой.

Базовый адрес по умолчанию для запускаемых файлов — 0x400000. Это значение жестко определено. Вот почему у нашего процесса исполняемые команды загружаются, начиная с адреса 0000000000400000.

» Следующий столбец, **Perm** — режим доступа. Этот параметр состоит из следующих флагов.

» **r** — доступен для чтения;

» **w** — доступен для записи;

» **x** — содержит исполняемый код;

» **p/s** — флаг **p** означает, что сегмент частный (*private*), флаг **s** — что это сегмент общего доступа (*shared*), то есть транслируется в область физической памяти, доступную и для других процессов; в некоторых форматах вместо флага **p** стоит прочерк.

» **R** — для этого сегмента не зарезервировано пространство в области подкачки на диске; значит, если память будет переполнена, мы получим ошибку сегментации.

» Далее идет столбец **Offset**. Информация в виртуальную память загружается из файлов. Значения в столбце **Offset** показывают, из какого адреса внутри файла (отсчитывая от его начала) информация загружается в этот сегмент.

» Следующий столбец — **Device**. Он обозначает устройство, с которым данный сегмент связан. Старший байт — номер драйвера устройства, под которым он зарегистрирован в системе. Младший байт — номер конкретного устройства из числа всех устройств такого типа.

» **Inode** — номер индексного дескриптора (*inode*) файла, связанного с этим сегментом.

» **Size** — размер сегмента в киббайтах.

» **Rss** — resident set size, объем памяти, реально находящейся в ОЗУ компьютера. Остальная память находится или в области подкачки на диске, или в файле.

» **Pss** — proportional set size, величина, которая вычисляется так: Rss частной памяти плюс Rss общедоступной памяти, разделенная на количество процессов, которые пользуются этой общедоступной памятью.

» **Referenced** — объем «активной» памяти, к которой произошло обращение.

» **Anonymous** — количество информации, которая была изменена после загрузки из файла и, таким образом, сейчас не соответствует тому, что находится в файле.

» **Swap** — объем памяти, находящейся в области подкачки на диске.

» **Locked** — объем заблокированной памяти.

» **Mapping** — файл, с которым связан данный сегмент. Как вы можете видеть, команды в эту область памяти загружаются не только из исполняемого файла *dd*, но и из динамических библиотек: *libc-2.19.so*, *ld-2.19.so*. Динамические библиотеки включают функциональность, которая может использоваться разными программами. При загрузке основной программы в память она также загружает в память и требуемую динамическую библиотеку. И все функции этой библиотеки становятся ей доступны.

Также можно видеть, что некоторые данные читаются из файла **locale-archive**. Он содержит языковые настройки для правильного отображения дат, чисел и некоторых стандартных сообщений.

Скорая помощь

Каждый процесс обращается только к адресам своего виртуального адресного пространства. Потом они транслируются в адреса физической памяти компьютера.

Для некоторых сегментов памяти вместо имен файлов стоят пометки [heap], [stack], [vsyscall], [vdso], [vvar] и [anon]. Это все анонимные сегменты, то есть такие сегменты, которые не связаны ни с каким файлом.

» [heap] — это область для размещения «кучи». Так называется хранилище объектов, создаваемых программой путем динамического распределения памяти. В дальнейшем, после их создания, программа может обращаться к этим объектам через указатель. При передаче объекта в другую функцию, передается только указатель, что избавляет от необходимости копировать весь объект. Но когда указатель выйдет за пределы видимости функции, он уничтожится. А объект в куче останется. Значит, чтобы не происходило утечки памяти, объект надо уничтожить или непосредственно в программном коде, или с помощью «сборщиков мусора».

» [stack] — это сегмент для хранения стека программы. Стек — это хранилище, организованное по принципу LIFO («последний вошел — первый вышел»). Оно предназначено для передачи параметров в подпрограмму и для возврата в программу из подпрограмм или из обработчика прерываний.

» [vsyscall] — сегмент для обращения к функциям ядра. В этот сегмент транслируется часть адресного пространства ядра, содержащая вызовы некоторых системных функций. Это позволяет делать часто используемые вызовы функций ядра, не переключаясь из пространства пользователя в пространство ядра. Это ускоряет работу. Но недостаток в том, что, во-первых, этот сегмент расположен всегда по одному и тому же адресу (ffffffff600000), а во-вторых, размер пространства ограничен, и поэтому количество системных вызовов тоже ограничено. В настоящее время этот способ вызова функций ядра признан устаревшим. Для преодоления указанных недостатков был придуман другой механизм — создание виртуальной динамической библиотеки vDSO.

» [vdso] — это область, в которую загружается виртуальная динамическая библиотека vDSO, содержащая вызовы функций ядра. Этой библиотеки нет на диске, она динамически создается ядром и загружается в этот сегмент. Виртуальная библиотека размещается в адресном пространстве по динамически выделенному адресу. А значит, она свободна от привязки к одному определенному адресу. Адрес выделяется, исходя из размеров библиотеки. Значит, эта библиотека не имеет ограничения на размер; следовательно, может содержать большее количество функций, чем vsyscall.

» [vvar] — эта зона выделена для хранения данных, используемых библиотекой vDSO.

» [anon] — прочие анонимные сегменты.

Программа *rtmp* берет информацию из виртуальных файлов: */proc/[PID]/maps* и */proc/[PID]/smaps*. Этих файлов на самом деле на диске нет. Файловая система */proc* — это виртуальная файловая система, которая является интерфейсом ко внутренним структурам ядра, отображенным в виде файлов.

Следует добавить, что, если в вашей системе установлен не пакет *procps-ng*, а оригинальный пакет *procps*, программа *rtmp* может выглядеть по-другому.

Программы *pgrep* и *pkill*

Программа *pgrep* находит процессы по определенным критериям. Критерием по умолчанию является имя программы, которое можно вводить частично. Давайте дадим следующую команду:

```
pgrep -a ba
```

```
Будут найдены все процессы, в названии которых содержится сочетание ba. В данном случае, это writeback и Bash. Ключ -a означает, что надо показывать PID вместе с именем. По умолчанию будет показан только PID:
```

```
21 writeback
7727 bash
```

Можно также задать и другие критерии. Несколько критериев можно объединять:

```
mint@mint ~ $ pmap -X 3198
3198: dd if=/dev/zero of=/dev/null
Address Perm Offset Device Inode Size Rss Pss Referenced Anon Swap Locked Mapping
00400000 r-xp 00000000 07:00 35 52 52 52 0 0 0 0 dd
0060c000 r--p 0000c000 07:00 35 4 4 4 4 4 0 0 dd
0060d000 rw-p 0000d000 07:00 35 4 4 4 4 4 0 0 dd
01355000 rw-p 00000000 00:00 0 132 20 20 20 20 0 0 [heap]
7f5c7112f000 r--p 00000000 07:00 23476 2852 412 24 412 0 0 0 locale-archive
7f5c711f8000 r-xp 00000000 07:00 10225 1772 1120 16 1120 0 0 0 libc-2.19.so
7f5c715b3000 r--p 001b3000 07:00 10225 2044 0 0 0 0 0 0 0
7f5c717b2000 r--p 001ba000 07:00 10225 16 16 16 16 16 0 0 0 libc-2.19.so
7f5c717b6000 rw-p 001be000 07:00 10225 8 8 8 8 8 0 0 libc-2.19.so
7f5c717b8000 rw-p 00000000 00:00 0 20 12 12 12 12 0 0
7f5c717bd000 r-xp 00000000 07:00 10199 140 140 1 140 0 0 0 ld-2.19.so
7f5c719bc000 rw-p 00000000 00:00 0 12 12 12 12 12 0 0
7f5c719dd000 rw-p 00000000 00:00 0 8 8 8 8 8 0 0
7f5c719df000 rw-p 00000000 00:00 0 4 4 4 4 4 0 0
7f5c719e0000 rw-p 00023000 07:00 10199 4 4 4 4 4 0 0 ld-2.19.so
7f5c719e1000 rw-p 00000000 00:00 0 4 4 4 4 4 0 0
7ffe7f3da000 rw-p 00000000 00:00 0 136 12 12 12 12 0 0 [stack]
7ffe7f3fb000 r--p 00000000 00:00 0 8 0 0 0 0 0 0 [vvar]
7ffe7f3fd000 r-xp 00000000 00:00 0 8 4 0 4 0 0 0 [vdso]
ffffffffffe00000 r-xp 00000000 00:00 0 4 0 0 0 0 0 0 [vsyscall]
=====
7232 1836 201 1836 108 0 0 KB
```

- » -g (--group) по группе процесса;
- » -G (--group) по реальной группе;
- » -u (--uid) по эффективному пользователю;
- » -U (--uid) по реальному пользователю;
- » -P (--parent) по PID родительского процесса;
- » -s (--session) по сессии процесса;
- » -t (--terminal) по терминалу;
- » -x (--exact) только с именем точно соответствующим указанному (то есть, имя должно быть задано полностью, не частично);
- » -o (--oldest) выдать PID только самого раннего процесса из числа соответствующих критериям;
- » -n (--newest) выдать PID только последнего запущенного процесса из числа соответствующих критериям;
- » -v (--inverse) обратить критерии: выдать только те процессы, которые НЕ соответствуют критериям.

Программа *pkill* посылает процессу сигнал. Для указания процесса, которому следует отправить сигнал, эта программа может пользоваться теми же критериями, что и *pgrep*. Для указания номера или названия сигнала используется ключ *--signal*. Задавать сигналы можно как по номеру, так и по полному или сокращенному имени.

Следующие сигналы поддерживаются всеми POSIX-совместимыми системами:

- » 1 HUP, SIGHUP — потеря соединения с терминалом (Hangup);
- » 2 INT, SIGINT — процесс остановлен пользователем с терминалом;
- » 3 QUIT, SIGQUIT — отправка пользователем команды quit;
- » 6 ABRT, SIGABRT — вызов функции abort() для аварийной остановки;
- » 9 KILL, SIGKILL — безусловное завершение процесса; в отличие от сигнала TERM, этот сигнал программа не может обработать и отменить;
- » 14 ALRM, SIGALRM — истекло время, заданное функцией alarm();
- » 15 TERM, SIGTERM — завершение процесса.

Помимо этого, система GNU/Linux поддерживает и много других сигналов. Подробную информацию о них можно прочитать в справке по следующей команде:

```
info coreutils 'Common options' 'Signal specifications'
```

Если сигнал не указать, то по умолчанию посылается сигнал №15 (TERM или SIGTERM).

Давайте опять запустим процесс *dd* и найдем его по следующему критерию: последний процесс текущего пользователя. Затем отправим ему сигнал №9, «безусловное завершение». Обратите

```
dmitry@dmitry-P5E-VM-HDMI ~ $ free -h
```

	total	used	free	shared	buffers	cached
Память:	2,0G	1,4G	590M	136M	81M	958M
-/+ буферы/кэш:		377M	1,6G			
Swap:	2,0G	0B	2,0G			

```
dmitry@dmitry-P5E-VM-HDMI ~ $
```

» Рис. 2. Распределение свободной и занятой памяти.

» Рис. 1. Карта памяти процесса.

Скорая помощь

Сигнал №15 (TERM, SIGTERM) программа может обработать, и даже может не подчиниться ему. Сигнал №9 (KILL, SIGKILL) обработать невозможно. Это безусловное завершение процесса.

```
Terminal
Файл Правка Вид Поиск Терминал Справка
Every 1,0s: free -h Thu Feb 25 22:24:13 2016
                total      used      free      shared  buffers   cached
Память:         2,0G        1,7G        236M        139M        43M        1,3G
-/+ буферы/кэш:      383M        1,6G
Swap:           2,0G          3,2M          2,0G
```

➤ Рис. 3. Память при интенсивной записи в файл.

```
Terminal
Файл Правка Вид Поиск Терминал Справка
Every 1,0s: free -h Thu Feb 25 22:32:22 2016
                total      used      free      shared  buffers   cached
Память:         2,0G        1,8G        122M        132M          10M        443M
-/+ буферы/кэш:      1,4G        576M
Swap:           2,0G          13M          2,0G
```

➤ Рис. 4. Память при открытии большого файла.

внимание, что можно, вместо ввода имени текущего пользователя, ввести команду `whoami`, поместив ее между символами ``` («тупое ударение»). Тогда на это место подставится результат выполнения команды.

Команда `pgrep` нам в данном случае нужна только чтобы посмотреть на процесс, который мы собираемся убить, и убедиться, что мы параметры задали правильно и по оплошности не убьем ничего лишнего.

```
dd if=/dev/zero of=/dev/null &
pgrep -anu `whoami`
pkill -nu `whoami` --signal SIGKILL
```

Как видите, ключи `-n` (последний запущенный процесс) и `-u` (пользователь процесса) доступны как программе `pgrep`, так и команде `pkill`. Ключ `-a` доступен только `pgrep`, так как он отвечает за отображение.

Программа free

Программа `free` позволяет просмотреть количество свободной и занятой памяти в системе — как физической, так и находящейся в разделе подкачки, а также в буфере и в кэше (рис. 2). По умолчанию информация выводится в килобайтах, но это можно изменить с помощью ключей `-b` (`--bytes`), `-k` (`--kilo`), `-m` (`--mega`), `-g` (`--giga`), `-t` (`--tera`). Ключ `-h` (`--human`) выводит данные в дружелюбном к человеку виде, самостоятельно определяя, в каком формате их отображать.

```
free -h
```

Отображается следующая информация: `total` — общее количество памяти, `used` — использованная память, `free` — свободная память, `shared` — память, используемая временным файловым хранилищем `tmpfs`, `buffers` — буферизованная память, `cached` — память, находящаяся в кэше, в том числе в кэшах распределителя `slab`.

Программа `free` берет данные из виртуального файла `/proc/meminfo`.

Как мы видим, программа эта выдает результат в обычном текстовом режиме, и каждый раз, как только мы захотим получить

➤ Рис. 5. Результаты работы программы `vmstat`.

```
dmitry@dmitry-P5E-VM-HDMI ~ $ vmstat 5 10
procs -----memory-----swap-----io-----system-----cpu-----
r b swpd free buff cache si so bi bo in cs us sy id wa st
0 1 13716 1302244 17860 353808 0 2 540 1223 124 462 6 5 87 3 0
0 0 13716 1297472 17860 360008 0 0 0 2 139 580 2 0 95 3 0
0 0 13716 1282776 17892 367688 0 0 0 50 172 743 6 1 91 2 0
0 0 13716 1285676 17908 370948 0 0 0 21 145 588 3 0 94 3 0
1 0 13716 1294852 17916 361228 0 0 0 9 57 170 0 0 99 0 0
0 0 13716 1306712 17916 349384 0 0 0 0 55 158 0 0 100 0 0
0 0 13716 1306776 17916 349564 0 0 0 0 79 269 0 0 100 0 0
0 0 13716 1294084 17916 361388 0 0 42 0 188 877 5 1 90 5 0
1 0 13716 1288008 17932 367760 0 0 0 40 160 672 6 1 91 2 0
0 0 13716 1288256 17940 367544 0 0 0 48 91 335 2 0 97 1 0
dmitry@dmitry-P5E-VM-HDMI ~ $
```

информацию о памяти, мы должны вызывать ее по новой. А можно ли получать эту информацию в реальном времени?

Можно. И не только эту. И поможет нам в этом следующая программа — `watch`.

Программа watch

Программа `watch` позволяет динамически наблюдать результаты работы программ, выводящих свои данные в текстовом режиме. По умолчанию, программа перезапускается каждые 2 секунды. Но с помощью ключа `-n` (`--interval`) можно изменить интервал перезапуска. Результаты отображаются на полном экране внутри консоли.

Давайте с помощью этой программы посмотрим в реальном времени информацию о памяти. Перезапускать будем с интервалом 1 секунда.

```
watch -n 1 free -h
```

Теперь откроем вторую консоль и создадим текстовый файл очень большого размера. Добиться этого результата можно, например, если сгенерировать перечисление чисел от одного до 500000000 (пятьсот миллионов), и направить результат в этот файл. Перечисление создается с помощью программы `seq` из пакета `coreutils`:

```
seq 500000000 > bigfile
```

Пока перечисление записывается в файл (а это процесс не быстрый), смотрим на ту консоль, где работает программа `watch`. Следим, что происходит с нашей памятью (рис. 3). Особенно обращаем внимание на раздел `cached`.

У нас сформировался файл с названием `bigfile`. Посмотрим, какого этот файл размера. Используем для данной цели утилиту `du` из пакета `coreutils`.

```
du -h bigfile
```

Результат получаем такой: 4,6 гигабайт.

Теперь открываем этот файл с помощью текстового редактора `vim` (тоже, понятно, дело не быстрое):

```
vim bigfile
```

И, пока он открывается, смотрим на ту консоль, где у нас работает программа `watch` (рис. 4). На этот раз смотрим на раздел «буферы/кэш».

Теперь выйдем из программы `vim` (для этого надо набрать `:q` и нажать `Enter`). Снова смотрим на консоль с программой `watch`. Память во всех разделах должна вернуться к первоначальному значению. Итак, мы разобрались, как наблюдать за памятью при выполнении разных тяжелых и долгих задач.

С помощью программы `watch` можно наблюдать в реальном времени много всякой другой информации. Например, какой-нибудь каталог. Давайте понаблюдаем в реальном времени свой текущий каталог —

```
watch -n 1 ls -l `pwd`
```

и создадим в нем новый файл:

```
touch newfile
```

Ну как, увидели изменение в консоли с программой `watch`? Сразу после создания файла он должен появиться в списке.

Программа vmstat

Программа `vmstat` отображает статистическую информацию о памяти и активности процессора. Если ввести команду без параметров, то отобразится только одна строка, показывающая состояние с момента загрузки системы. Если задать числовой параметр, он укажет, с какой частотой обновлять информацию. Второй параметр укажет, сколько раз обновлять информацию. Если второй параметр опустить, информация будет обновляться бесконечно.

Главная задача этой программы — определить, в каком месте находится «узкое место» производительности (или, как еще говорят, «бутылочное горлышко»).

```
vmstat 5 10
```


В таблице на рис. 5 наиболее важен столбец `r` — количество потоков в очереди на запуск (`runqueue`). Его значение должно быть не больше, чем количество процессоров в системе. Если значение больше, то это сигнализирует о нерациональной или чрезмерной загрузке процессора, неправильном распределении приоритетов.

Также интересны параметры `us`, `sy`, `id`. Они означают время работы процессора в режиме пользователя (`us`) и в режиме ядра (`sy`), и время простаивания процессора (`id`). Суммарное значение должно быть 100%.

В идеале, процессор должен большей частью работать в режиме пользователя или простаивать. Если процессор работает в режиме ядра больше, чем в остальных режимах, это свидетельствует о его нерациональной загрузке или проблемах внутри ядра. Впрочем, в высоконагруженных системах время простоя может приближаться к нулю, для них это норма.

Узкое место в памяти проявляет себя, как подкачка, то есть выгрузка на диск отдельных страниц (`raging`) или всего процесса (`swaping`). Информацию об этом процессе можно получить из параметров: `si` (сколько считывается из области подкачки), `so` (сколько сохраняется в область подкачки) и `swpd` (сколько находится в области подкачки).

Программа slabtop

Распределение `slab` — это механизм резервирования участков памяти для часто используемых объектов. Он позволяет избежать затрат, связанных с повторным выделением памяти, а также уменьшает фрагментацию памяти. Каждый зарезервированный слот памяти хранится в виде кэша. Программа `slabtop` служит для просмотра полного списка кэшей `slab` в реальном времени. Подобно программе `top`, она работает в псевдографическом режиме; ее окно (рис. 6) можно прокручивать вверх и вниз с помощью клавиш со стрелками. Выход из программы осуществляется клавишей `q`. Запуск этой программы — `slabtop` — возможен только с правами суперпользователя.

Данные в эту программу поступают из виртуального файла `/proc/slabinfo`. Как и всё содержимое каталога `/proc`, этот файл на самом деле файлом не является, а является интерфейсом для получения информации о процессах в ядре. Просматривать содержимое этого файла можно только с правами суперпользователя.

Программа tload

Это псевдографическая программа, отображающая загрузку процессора. Значение графика варьирует между 0 и 1. Значение 1 означает загрузку процессора на 100%. Цифры вверху показывают среднюю загрузку процессора в течение последних 1, 5 и 15 минут.

На рис. 7 резкий подъем графика вверх начался точно в тот момент, когда в соседней консоли был запущен известный вам тяжелый процесс `dd if=/dev/zero of=/dev/null`. А когда этот процесс был убит, начался спуск вниз, который также виден на картинке.

Программа sysctl

Программа `sysctl` дает возможность «на лету» изменить настройки ядра. Какие именно настройки редактируются в программе, можно посмотреть, запустив эту программу с ключом `-a` (`--all`). Будет показан очень длинный список настроек с их значениями. Чтобы его сократить, покажем его в конвейере с утилитой `head` из пакета `coreutils`. Эта утилита по умолчанию показывает только первые 10 строк текста.

```
sysctl -a | head
abi.vsyscall32 = 1
crypto.fips_enabled = 0
debug.exception-trace = 1
debug.kprobes-optimization = 1
dev.cdrom.autoclose = 1
dev.cdrom.autoeject = 0
```

OBJS	ACTIVE	USE	OBJ	SIZE	SLABS	OBJ/SLAB	CACHE	SIZE	NAME
6144	6144	100%	0	0,01K	12	512	48K		kmalloc-8
0	0	0%	0	0,01K	0	512	0K		dma-kmalloc-8
12544	11719	93%	0	0,02K	49	256	196K		kmalloc-16
0	0	0%	0	0,02K	0	256	0K		dma-kmalloc-16
256	256	100%	0	0,02K	1	256	4K		jbd2_revoke_table_s
3910	3910	100%	0	0,02K	23	170	92K		Acpi-Namespace
0	0	0%	0	0,02K	0	170	0K		fat_cache
256	256	100%	0	0,03K	2	128	8K		kmem_cache_node
19200	18191	94%	0	0,03K	150	128	600K		kmalloc-32
0	0	0%	0	0,03K	0	128	0K		dma-kmalloc-32
3328	2816	84%	0	0,03K	26	128	104K		ext4_extent_status
0	0	0%	0	0,04K	0	102	0K		iint_cache
204	204	100%	0	0,04K	2	102	8K		eventpoll_pwq

► Рис. 6. Окно программы `slabtop`.

```
dev.cdrom.check_media = 0
dev.cdrom.debug = 0
dev.cdrom.info = CD-ROM information, ID: cdrom.c 3.20
2003/12/17
dev.cdrom.info = drive name: sr0
```

Редактировать надо только нужные настройки, и только разобравшись как следует, что они означают. Все эти настройки можно просмотреть также в виртуальной файловой системе `/proc`, в каталоге `/proc/sys`. Посмотрим, что там находится:

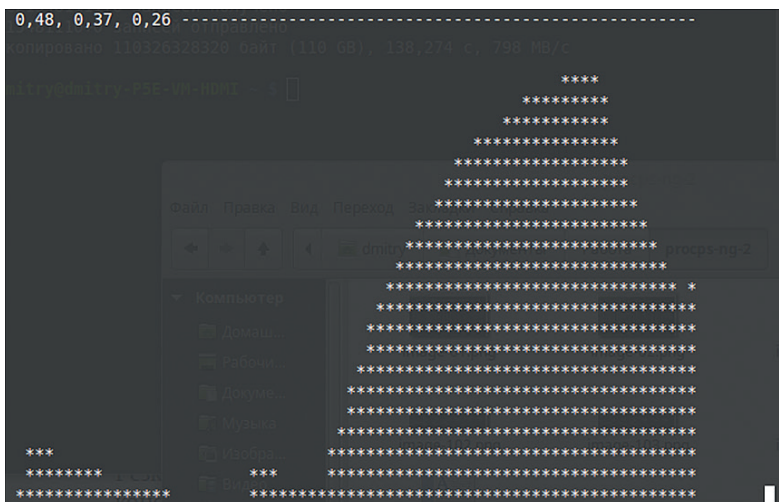
```
ls /proc/sys
Там мы увидим вложенные каталоги — те же самые, которые встретились нам в названиях настроек:
abi crypto debug dev fs kernel net sunrpc vm
```

Зайдя в подкаталог `abi`, мы увидим там файл `vsyscall32` (это виртуальный файл, конечно). Его название мы только что видели в первой строчке списка.

Программы skill и snice

Как сказано в справочной системе по этим программам, они признаны устаревшими, и оставлены только для совместимости со старыми версиями GNU/Linux. Вместо них рекомендуется пользоваться программами `pskill` из пакета `procps-ng`, `snice` из пакета `coreutils` и `renice` из пакета `util-linux`.

Итак, мы в общих чертах познакомились с пакетом `procps-ng`, набором мощных средств для работы с системой GNU/Linux. Здесь, конечно, описаны далеко не все возможности. Да их и невозможно описать полностью. Поэтому, пожалуйста, не забывайте пользоваться справками `man` и `info`. **LXF**



► Рис. 7. Окно программы `tload`.

Scalpel: Спасаем!

Нейт Дрейк вооружается инструментом командной строки *scalpel*, чтобы восстановить уничтоженные файлы.



Наш эксперт

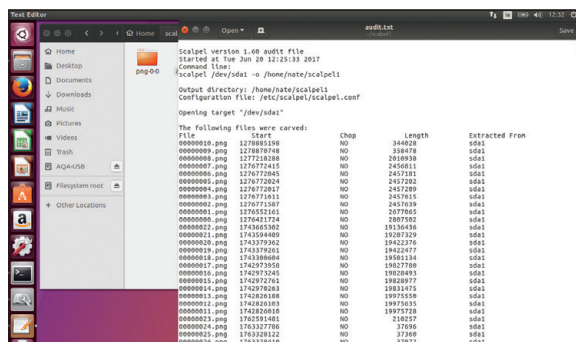
Нейт Дрейк привык пополнять свой скудный доход, предлагая услуги по восстановлению документов с помощью *scalpel*. Его первым клиентом была танцовщица бурлеска. Добром это не кончилось...

При каждом запуске *scalpel* генерирует файл аудита (*audit.txt*), со списком всех файлов, которые удалось восстановить.

Читателям, которым повезло быть назначенными в службу техподдержки своей семьи, могут знать о склонности своих близких хранить то, что мы образно говоря называем данными Шредингера. Этот термин используется для таких личных файлов, как фотографии, календари и документы, которые, по-видимому, абсолютно необходимы для семейного благополучия, но, как это ни парадоксально, недостаточно важны для резервного копирования. В итоге — если важнейший файл случайно удаляется, то, похоже, нет простого способа его восстановить.

На самом деле, как знают ИТ-грамотные люди, многие компьютеры стирают файлы не полностью, и многие из них удается восстановить с помощью специальных инструментов даже через значительное время после очистки корзины. Один из таких инструментов, одинаково любимый и гуру техподдержки, и правоохранительными органами, называется *scalpel*.

Scalpel — это утилита командной строки, которая для восстановления удаленных данных с образа, раздела или всего диска использует метод под названием «вырезание файлов [file carving]». На этом уроке мы попробуем овладеть его основами.



Scalpel очень надежен и работает практически с любым типом устройства или файловой системы. Во время тестирования мы смогли восстановить удаленные файлы PNG и PDF как с флэш-накопителя на 512 МБ, так и с VDI (виртуальный образ диска) *VirtualBox*. Однако если файлы, которые вы хотите восстановить, находятся непосредственно на жестком диске вашей системы, рекомендуем загрузить ваш компьютер с live DVD или компакт-диска. Для этой статьи мы использовали live Ubuntu 17.04. *Scalpel* доступен в репозиториях Ubuntu, и его можно установить через терминал с минимумом усилий.

Хирургически точно

Хотя скальпель — точный инструмент, он не затрагивает файловые системы, а последовательно анализирует блоки данных на диске, образе диска или в файлах подкачки. Если вам нужен подробный анализ конкретной работы *scalpel*, то документация есть на <http://bit.ly/2v32FA1>. Вкратце, однако, достаточно знать, что *scalpel* основан на прежнем инструменте восстановления данных *foremost*, который был разработан для расследований правоохранительных органов. Как и его предшественник, *scalpel* использует выделение однородных массивов данных для простого восстановления данных.

Этот метод включает использование встроенной базы данных заголовков и окончаний для конкретных типов файлов, таких как изображений PNG или файлов PDF. Заголовки и окончания состоят из строк байтов, предсказуемо отстоящих от начала, и файлы можно извлекать или вырезать прямо из образов дисков, не затрагивая имеющуюся файловую систему. *Scalpel* создан, чтобы делать это гораздо эффективнее, чем *foremost*.

При первом запуске *scalpel* проверяет свой файл настройки, который обычно находится в */etc/scalpel/scalpel.conf*. По умолчанию не выбрано ни одного файла, но вы можете следовать

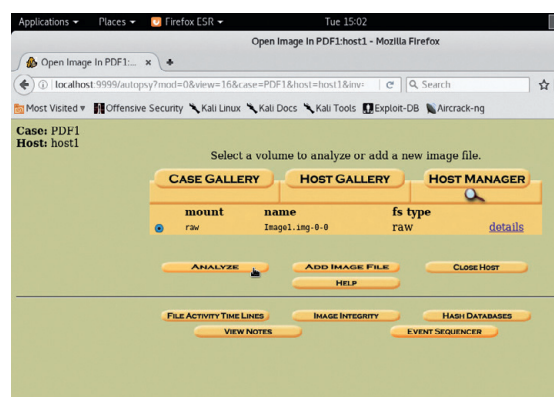
Представляем autopsy

Scalpel — один из многих инструментов, доступных как часть The Sleuth Kit [Снаряжение ищущих]. Это набор утилит, предназначенных для криминалистического анализа компьютеров. Если вы обнаружите, что *scalpel* не может восстановить данные, вам может больше повезти с другими инструментами, способными анализировать данные на более глубоком уровне, например, на уровне файловой системы.

Все инструменты запускаются и из командной строки, поэтому для начала перейдите на <https://github.com/sleuthkit/sleuthkit> и следуйте инструкциям по компиляции и запуску из документа INSTALL. Поскольку каждый инструмент разработан для определенной цели, это может оказаться нудным, поэтому проект TSK рекомендует применять инструменты в сочетании

с криминалистическим браузером *Autopsy*, обеспечивающим простой интерфейс «наведи-и-щелкни» для большинства обычных задач. Версия *Autopsy* (v2) для Linux больше не поддерживается, но ее можно загрузить с www.sleuthkit.org/autopsy/v2. Альтернатива — загрузив последнюю версию Kali Linux с www.kali.org/downloads, вы найдете готовый браузер *Autopsy* в секции Forensics раздела Applications. *Scalpel* тоже установлен.

Autopsy разработан с ориентацией на правоохранительные органы и требует сначала создать образ диска раздела, который вы хотите проанализировать, чтобы в дальнейшем не вызвать обвинения в фальсификации. Если вы хотите узнать больше, в вики Sleuth Kit есть отличная документация, которая поможет вам начать работу — посетите <http://bit.ly/2tNhLdL>.



Autopsy служит интерфейсом пользователя для передовых инструментов анализа, доступных в TSK (The Sleuth Kit). Обе эти утилиты предустановлены в Kali.

пошаговой инструкции внизу страницы, чтобы отредактировать этот файл, что позволяет... указать, какой тип вы хотите восстановить. Затем *scalpel* выполнит два последовательных прохода по исследуемому диску, обрабатывая данные порциями по 10 МБ. В каждой порции сначала ищутся заголовки файлов. После этого второй проход будет искать соответствующие им окончания. *Scalpel* также создает набор очередей работ, которые регламентируют операции вырезания файлов. Все восстановленные файлы вместе с журналом работы *scalpel* помещаются в указанный вами выходной каталог.

Двойной подход

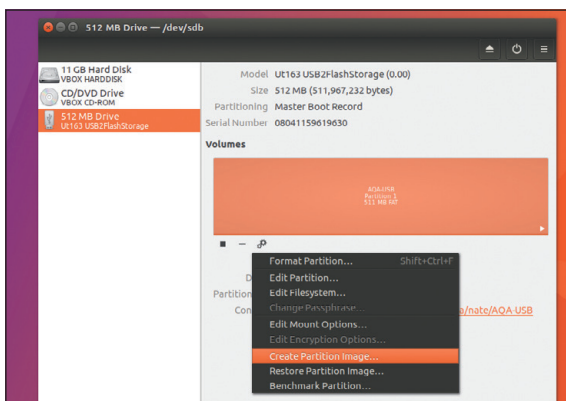
Когда агенты правоохранительных органов изымают диск для криминалистического анализа, они обычно делают образ диска — в основном, чтобы избежать обвинений в подтасовке улики после изъятия диска. Но это также означает, что если в процессе восстановления что-то засбито, можно сделать еще одну копию накопителя и повторить попытку. Когда доходит до попытки восстановить данные, также существует риск, что, если вы толком не знаете, что делаете, можно не только не преуспеть в восстановлении своих файлов, но и усложнить это занятие даже для профессионала.

У Linux, как всегда, есть пара грамотных решений. Во-первых, используйте встроенную дисковую утилиту для создания образа целевого диска, такого как USB-накопитель. Это поблочная копия всех данных в разделе. *Scalpel* анализирует образы так же легко, как и сами диски: просто запустите *scalpel* <имя-образа>, например, *scalpel /root/Desktop/image1.img*.

Второй вариант — использовать дистрибутив Kali Linux, разработанный с учетом работы в криминалистике. Мало того, что Kali поставляется с предустановленным *scalpel*, но при первой загрузке вы можете выбрать загрузку в режиме Forensic [Аналитический], что не позволит ОС монтировать жесткий диск и автоматически монтировать съемные носители, такие как USB-накопители.

Неудачные операции

Если вы сделали копию разделов или томов для анализа, то *scalpel* мало что может испортить. Способность инструмента восстанавливать файлы зависит от ряда факторов, в том числе от типа файловой системы, используемой вашим целевым диском. Системы с журналированием нередко сохраняют копии одних и тех же



» Безопасности ради, рекомендуется использовать *Disk Utility* для создания образа целевого диска для анализа.

данных в нескольких местах, делая восстановление файлов более осуществимым.

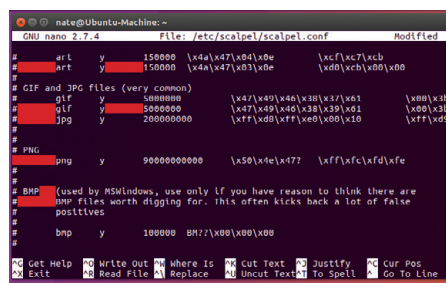
Если заголовки или окончания файлов были перезаписаны или целевой диск был зашифрован, возможно, вы не сможете восстановить удаленные файлы. Это, конечно, хорошая новость для тех читателей, которые ценят свою конфиденциальность, потому что теперь вы знаете, как ваши диски могут противостоять упрощенному криминалистическому анализу вроде этого. В наших тестах мы запускали команду *shred* на двух PDF, хранящихся на USB-накопителе. *Scalpel* не смог восстановить ни один из них. Однако они могут быть восстановлены с помощью более продвинутых инструментов, доступных в наборе Sleuth Kit (см. врезку внизу стр. 56).

На этом уроке основное внимание уделяется версии *scalpel* (1.6), доступной в репозиториях Ubuntu, но в установке самой свежей версии есть некоторые преимущества. *Scalpel 2.0* включает поддержку многопоточности для ускорения работы на многоядерных процессорах, а также лучшее распознавание типов файлов, которые сами могут содержать файлы. Вы можете легко загрузить и распаковать ZIP-файл с GitHub (<https://github.com/machn1k/Scalpel-2.0/archive/master.zip>), но для работы *scalpel* потребуется установить библиотеку регулярных выражений *tre*. Убедитесь, что у вас предустановлены инструменты *autoconf*, *automake*, *gettext*, *libtool* и *autopoint*, затем запустите скрипт *autogen.sh* в каталоге *Scalpel-2.0-master/tre-0.7.5-win32*. Ну и запустите *./configure*, *make*, затем *sudo make install* — как для *tre*, так и для *scalpel 2.0*. **LXF**

Скорая помощь

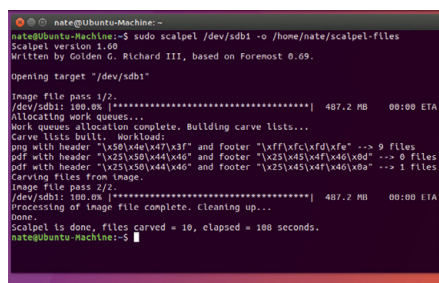
Если на компьютере не удается примонтировать диск, это может означать проблему с таблицами разделов. Попробуйте перед запуском *scalpel* восстановить данные с помощью *TestDisk* (www.cgsecurity.org/wiki/TestDisk).

Восстановите данные со scalpel



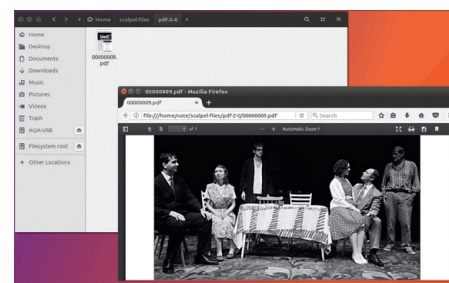
1 Настройте scalpel

Откройте терминал и запустите *sudo apt-get install scalpel* для установки *scalpel*. Затем запустите *sudo nano /etc/scalpel/scalpel.conf*. Прокрутите вниз до того типа файла, который вам требуется найти, и «раскомментируйте» строку, удалив # в ее начале. Повторите это для всех типов файлов, которые хотите восстановить, потом нажмите Ctrl+X, Y, а затем Enter, чтобы сохраниться и выйти. Теперь *scalpel* должен быть готов к работе.



2 Начните восстановление

Основным синтаксисом команды для *scalpel* является *scalpel <drive> -o <output-dir>*, например, *sudo scalpel /dev/sdb1 -o /home/nate/scalpel-files*. Для сканирования диска надо будет использовать *sudo*, но анализировать образы дисков можно и из учетной записи обычного пользователя. Это занимает некоторое время, зависящее от размера тома и количества файлов. По завершении откройте выходной каталог в файловом менеджере.



3 Анализ восстановленных файлов

Выходной каталог содержит подробный список восстановленных файлов. Есть также отдельная папка для каждого типа файла, которые вы пытались восстановить. Откройте их, чтобы узнать, удалось ли *scalpel* выкроить ваши файлы. Скопируйте все восстановленные файлы на отдельный диск, затем попробуйте их открыть. Если файлы окажутся испорченными, потребуются более сложные криминалистические методы.

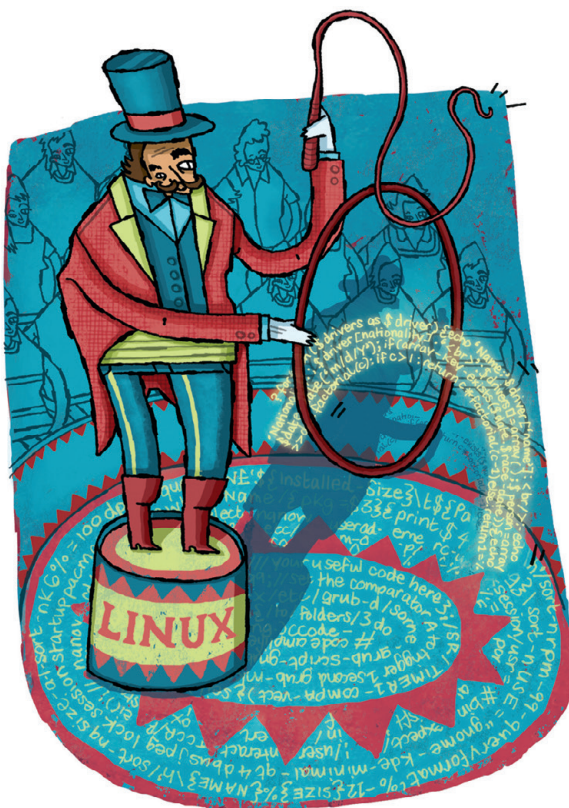
Vagrant: ВМ — ЭТО ОЧЕНЬ ПРОСТО

Бобби Мосс показывает, как попробовать свежие дистрибутивы Linux и запустить приложения из других ОС, не рискуя своим домашним компьютером.



Наш эксперт

Бобби Мосс разрабатывает облачные микросервисы для международной консалтинговой компании в сфере IT и ее клиентов. В свободное время работает над свободными проектами и возится со старым оборудованием.



Скорая помощь

Если файлы Vagrantfiles кажутся знакомыми, это потому, что в них используется тот же синтаксис, что и в языке программирования Ruby. К счастью, вам не надо быть опытным разработчиком, чтобы изменять настройки виртуальной машины, так как для этого достаточно просто добавлять, удалять и изменять пары «ключ—значение».

Услышав название 'Vagrant', вы, возможно, зададитесь вопросом: что общего с виртуальными машинами у бомжей [vagrant — *англ.* бродяга]? И вы будете в хорошей компании: по своему имени, этот проект достойно потягается с такими, как 'Kafka [Кафка]', 'Avogato [Авокадо]' и 'Mustache [Усы]'.

Уточним: Vagrant — это распространенный язык скриптов по управлению виртуальными машинами (ВМ) и их оснащению для различных инструментов виртуализации, таких как *VirtualBox* и *VMware Workstation*. Для преобразования инструкций, указанных в конфигурации (выбранная операционная система, количество ядер процессора, объем оперативной памяти или определенный пакет необходимого вам программного обеспечения) он использует драйверы. Затем эти инструкции применяются для создания виртуальных машин, соответствующих вашим требованиям. Все действия, выполняемые вручную на этих платформах, можно полностью автоматизировать, если в Vagrant есть соответствующий драйвер. Некоторые драйверы поддерживаются непосредственно Hashicorp (компанией-разработчиком Vagrant), но большинство драйверов разрабатывается более широким сообществом.

Как легко представить, такая независимость от платформы превратила Vagrant в очень популярный инструмент IT-отделов по всему миру. Возможность через один фреймворк развертывать и сворачивать по требованию несколько виртуальных машин на различных виртуализационных инструментах и облачных средах очень ценна для системных администраторов, тестировщиков и разработчиков программного обеспечения.

Однако на нашем уроке мы сосредоточимся на внедрении возможностей и гибкости Vagrant в домашнюю среду — для проверки последних дистрибутивов Linux, для запуска программного обеспечения, не предназначенного для вашей системы, и даже для того, чтобы заставить эти пошлые приложения Windows, без которых вам не обойтись, работать надежно и безопасно, без риска для вашей домашней системы с обычными повседневными задачами: игры, просмотр интернет-страниц, редактирование документов и др.

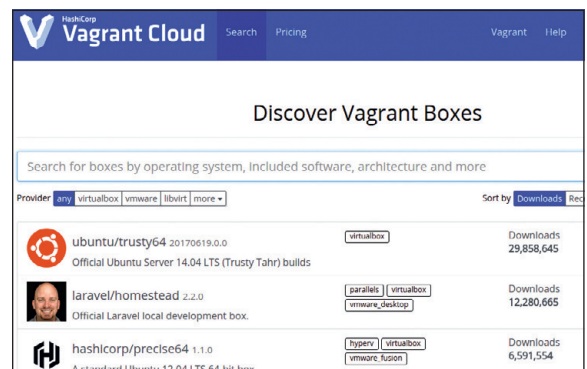
Кирпичики

Прежде чем мы начнем, следует отметить, что версии Vagrant и *VirtualBox*, взятые для нашего урока, идут в связке. Дело в том, что драйвер Vagrant надо обновить для поддержки последних версий *VirtualBox*. И может оказаться, что инструменты, которые вы загружаете через обычные репозитории своего дистрибутива, несовместимы друг с другом.

Если вам нужны все самые новые и лучшие функции, советуем загрузить соответствующие пакеты прямо с сайтов проекта, по ссылкам <http://bit.ly/1XvPzB6> и <http://bit.ly/1fPLZt5>. Также потребуется установить дополнительный пакет расширения *VirtualBox*, чтобы воспользоваться всеми функциями Vagrant, в том числе автоматическое использование общих папок и поддержка более широкого набора конфигураций оборудования.

После установки пакета расширения откройте терминал и введите такие команды:

```
$ mkdir test
```



Вы можете найти образы для любых целей — и от самой компании Hashicorp, и от более широкого сообщества.

Vagrant и Docker

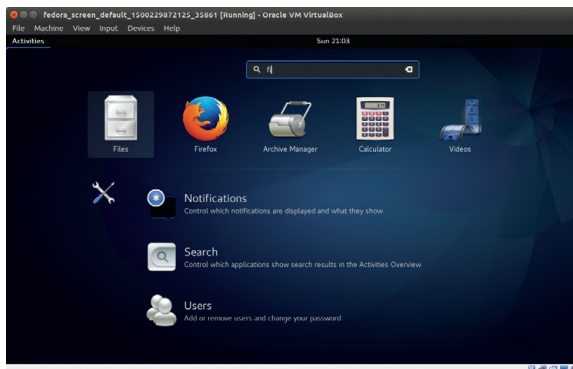
Docker — это технология контейнеризации, которая изолирует друг от друга приложения, использующие одно и то же ядро. Главное преимущество для вас, если вы не разработчик, заключается в том, что с Docker можно в Fedora запустить приложение, специально упакованное для Ubuntu, без проблем с совместимостью. Однако стоит иметь в виду, что Docker всё равно может внести изменения в основную систему хоста, поэтому следует

внимательно выбирать пакеты, запускаемые Docker в домашней системе.

Как и в Vagrant, в Docker существует "DockerFile", аналогичный Vagrantfile. Дополнительные сведения о нем доступны на <http://dockr.ly/1PalBK>.

Docker — одна из технологий виртуализации, поддерживаемых Vagrant. Но вы также можете использовать собственный файл Vagrantfile для развертывания нескольких контейнеров Docker

на одной виртуальной машине *VirtualBox*. Т.е. вместо того, чтобы выбирать между Docker и Vagrant, вы можете получить дополнительную гибкость и понизить требования к аппаратным ресурсам, запустив Docker и Vagrant в тандеме. В такой конфигурации вам понадобится только одна гостевая ОС для запуска нескольких приложений; но тогда, если виртуальная машина остановится, и Docker тоже остановится.



» Vagrant не просто автоматизирует установку виртуальных машин: с его помощью можно пробовать новые и свежие дистрибутивы Linux, не поломав свой загрузчик.

```
$ cd test
$ vagrant init
```

Вы увидите, что в созданной папке **test** появился некий 'Vagrantfile'. Это основной файл настройки, где определяются атрибуты всех виртуальных машин, созданных из этого каталога. Желая изменить виртуальную машину, изменяйте этот файл. А если вам надо создать несколько виртуальных машин, придется создать новый каталог и новый файл Vagrantfile для каждой из них.

Полная структура Vagrantfile показана на рис. 1 (внизу справа), но созданный вами файл, скорее всего, выйдет немного другим, поскольку в нем будут строки со всевозможными комментариями, которые мы для краткости опустили. Но в этом файле можно добавить перенаправление портов, если ваша конфигурация включает какие-либо сетевые приложения, например, *Apache*; добавить дополнительные общие папки для основной и гостевой ОС; и задать любые атрибуты оборудования создаваемой виртуальной машины. Позже на нашем уроке мы рассмотрим, как развернуть дополнительное программное обеспечение для гостевой ОС с помощью этого файла и инструментов для управления конфигурацией, таких как *Ansible*.

Поскольку мы работаем с *VirtualBox*, вы увидите дополнительную строку `modifyvm` для функций, которые еще будут включены в драйвер. Это работает, так как Vagrant предлагает общую оболочку для службы *VBoxManage*, предоставляющей интерфейс командной строки для всех функций, обычно доступных в графическом интерфейсе. Полную справку по всем доступным командам можно получить по ссылке <http://bit.ly/1HUjufW>, и если вы захотите использовать эти команды, любые дополнительные строки всегда будут следовать одному и тому же шаблону.

Ну вот, как устроен Vagrantfile, мы разобрались. Давайте перейдем в совершенно новый каталог и запустим следующие команды:

```
$ vagrant init jhcook/fedora25
$ vagrant up
```

Этот Vagrantfile будет немного отличаться от предыдущего. Строка `config.vm.box` теперь содержит значение `jhcook/fedora25` вместо `base`, так как мы указали требуемый образ в конце строки "init". Для целей нашего урока мы запустим экземпляр Fedora Workstation 25. Образы для различных операционных систем разнообразного назначения можно найти по ссылке <http://bit.ly/2v7xTUE>.

Первоначальный запуск образа занимает довольно много времени. Дело в том, что Vagrant загружает файл образа — «коробку [box]», содержащую виртуальную машину и ее параметры по умолчанию — с удаленного сервера, и ваш прогресс будет отличаться в зависимости от широкополосного подключения.

Однако, если вы откроете GUI — графический интерфейс — *VirtualBox*, то увидите очень странную вещь. Виртуальная машина есть, она включена и явно работает, хотя к графическому интерфейсу мы даже не прикасались!

Причина в том, что по умолчанию графический интерфейс *VirtualBox* отключен, и с текущей конфигурацией Vagrantfile вам придется вручную открыть его из самого *VirtualBox*. Хотя это очень удобно для разработчиков, создающих web-сервисы, для наших целей такое далеко не идеально. К счастью, это можно быстро и легко исправить, раскомментировав некоторые строки в файле Vagrantfile, чтобы он содержал следующий код:

```
config.vm.provider "virtualbox" do |vb|
  vb.gui = true
end
```

```
grant.configure(2) do |config|
  config.vm.box = "base"
  # config.vm.box_check_update = false
  # config.vm.network "forwarded_port", guest: 80, host: 8080
  # config.vm.network "private_network", ip: "192.168.33.10"
  # config.vm.network "public_network"
  # config.vm.synced_folder "../data", "/vagrant_data"
  # config.vm.provider "virtualbox" do |vb|
  #   vb.gui = true
  #   vb.cpus = 2
  #   vb.memory = "1024"
  #   vb.customize ["modifyvm", :id, "--cpuexecutioncap", "50"]
  # end
  # Enable provisioning with a shell script. Additional provisioners such as
  # Puppet, Chef, Ansible, Salt, and Docker are also available. Please see the
  # documentation for more information about their specific syntax and use.
  # config.vm.provision "shell", inline: <<-SHELL
  #   sudo apt-get update
  #   sudo apt-get install -y apache2
  # SHELL
```

» Рис. 1. Здесь показана полная структура файла Vagrantfile, используемого для указания атрибутов VM.

» Подпишитесь на печатную или электронную версии на www.linuxformat.ru/subscribe!

Сохраните и закройте файл, затем перезапустите виртуальную машину, выполнив следующую команду из того же каталога, в котором находится новый Vagrantfile:

```
$ vagrant reload
```

После перезапуска виртуальной машины появится знакомый рабочий стол Gnome от Fedora. Имя пользователя и пароль по умолчанию — ‘vagrant’.

С помощью команд Vagrant можно не просто запустить или перезапустить виртуальную машину. Следующие команды переводят виртуальную машину в спящий режим, пробуждают и выключают соответственно:

```
$ vagrant suspend
```

```
$ vagrant resume
```

```
$ vagrant halt
```

Перед выключением или перезапуском рекомендуется выполнить команду `suspend` или `halt`, чтобы не повредить гостевую ОС. Если вы сомневаетесь в текущем состоянии запущенной виртуальной машины, просто введите следующую команду — и узнаете, что происходит:

```
$ vagrant status
```

Наконец, желая полностью удалить виртуальную машину независимо от ее текущего состояния, используйте следующую команду:

```
$ vagrant destroy
```

Вы заметите, что во второй раз виртуальная машина создается намного быстрее, чем в первый, так как Vagrant достаточно умен,

чтобы сохранить извлеченную версию файла «коробки» на жесткий диск. Если установить в Vagrantfile соответствующий параметр, Vagrant просканирует сервер на наличие обновлений.

Также стоит отметить, что если вы скопируете любой файл в ту же папку, в которой находится Vagrantfile, то для гостевой ОС Linux этот файл будет доступен в общей папке /**vagrant**. Это можно проверить с помощью файлового менеджера на созданной виртуальной машине или подключившись к виртуальной машине из главной ОС через SSH:

```
$ vagrant ssh
```

```
$ sudo su -
```

```
$ ls -l /vagrant
```

Чтобы завершить сеанс SSH, просто дважды наберите `exit`. При любом входе в гостевую операционную систему вы можете повысить права пользователя `vagrant` до `root`, не указывая пароль при запуске второй команды.

Это поведение используется по умолчанию, так как созданные вами виртуальные машины одноразовые и используются только пока они вам нужны.

Тут есть смысл: ведь после удаления эти виртуальные машины всегда можно запустить снова благодаря созданному файлу настройки Vagrantfile.

Если вы хотите, чтобы виртуальная машина работала в течение длительного периода времени, рекомендуется изменить пароль пользователя `vagrant` и создать новую учетную запись для повседневного использования.

Ansly для Ansible

Возможности Vagrant не ограничиваются только созданием виртуальных машин и их включением и выключением. Мы можем сделать шаг вперед, автоматически устанавливая полезные приложения в базовый образ при запуске команды `vagrant up`.

Это делается разными способами. Можно приписать команд в конец файла Vagrantfile, но их, как правило, надо будет изменять для различных дистрибутивов, так как в них используются разные менеджеры пакетов или разные каталоги для размещения файлов. К счастью, есть и другие варианты.

В мире DevOps существуют системы, способные централизованно управлять установкой и настройкой различных пакетов на серверах и виртуальных машинах без учета платформы. Самые распространенные из них — *Chef*, *Puppet* и *Ansible*. Мы поговорим о них позже в этом руководстве.

Самый простой способ включить *Ansible* в Vagrantfile — добавить следующий код перед последним выражением `end`:

```
config.vm.provision "ansible_local" do |ansible|
```

```
  ansible.playbook = "playbook.yml"
```

```
end
```

Поскольку мы используем `ansible_local`, а не `ansible`, то устанавливать *Ansible* локально в домашней системе не надо. Vagrant достаточно умен и при создании виртуальной машины может установить *Ansible* в гостевой операционной системе сам.

Как вы, возможно, догадались по средней строке, требуется создать файл YAML **playbook.yml** в том же каталоге, где находится файл Vagrantfile. Также можно добавить следующие три строки в начало файла:

```
---
- hosts: all
  become_user: root
  become: yes
```

Теперь у нас есть базовая конфигурация *Ansible*, и мы можем начать создавать «задачи» для системы. Давайте удалим вашу VM Fedora и изменим следующую строку в нашем Vagrantfile:

```
config.vm.box = "boxcutter/ubuntu1604-desktop"
```

Откроется новый рабочий стол Ubuntu, но образ всё равно может оказаться немного устаревшим. Так что имеет смысл обновить



➤ Установка с помощью сценариев от *Ansible* позволяет быстро установить и настроить пакеты в вашей системе. Это особенно удобно при поломке VM!

Ansible и Fedora

К сожалению, при использовании гостевой ОС Fedora 23 или более поздней версии поддержка *Ansible* не слишком хороша. Это связано с тем, что дистрибутив больше не поставляется с Python 2, а Python 2 необходим для правильной работы модуля установки RPM в *Ansible*. И если вы попытаетесь запустить *Ansible*, то просто получите сообщения об ошибках из-за отсутствия модулей Python.

Пока в *Ansible* не перешли на Python 3 (версия *Ansible* с поддержкой Python 3 на момент написания этой статьи была в активной разработке), вы можете обойти эту проблему, отправляя менеджеру пакетов «сырые» команды:

```
- name: Upgrade all packages
```

```
raw: dnf -y update
```

```
- name: Install Chromium
```

```
raw: dnf -y install chromium
```

Недостаток описанного подхода в том, что *Ansible* не знает, были ли ваши приложения установлены успешно и используются ли в вашей системе самые свежие версии приложений. Так что эти задачи будут вызываться при каждом запуске команды `vagrant up`, а подобное поведение весьма далеко от идеального! Поэтому всегда предпочтительнее будет использовать соответствующие задачи своего диспетчера пакетов, а не отправку «сырых» команд.

все пакеты и (в качестве бонуса) установить на вашу новую виртуальную машину *Chromium*. Это можно сделать, добавив еще несколько строк в ваш файл YAML:

```
tasks:
- name: Upgrade all packages
apt: upgrade=dist
- name: Install Chromium
apt: name=chromium-browser state=latest
```

Сохранив и закрыв новый файл, просто выполните следующую команду, чтобы применить изменения:

```
$ vagrant provision
```

Эта команда подойдет и для дистрибутивов с *Yum*, если заменить `apt` на `rpm` и сократить `chromium-browser` до `chromium`.

Хотя задачи *Ansible* должны автоматически запускаться при создании нового образа, их можно запустить и вручную, скомандовав

```
$ vagrant up --provision
```

Разверните свой образ

Если при мысли о загрузке неизвестных образов с сайтов сообщества Vagrant вас пробирает дрожь, вы будете рады услышать, что очень легко самим развернуть собственные образы Vagrant для виртуальных машин *VirtualBox*. Давайте рассмотрим, как это делается.

Ранее мы уже говорили, что Vagrant можно использовать для управления виртуальными машинами Windows, на которых надо запустить старое программное обеспечение. Из-за лицензионных ограничений несерверные версии Windows не доступны как бесплатный образ Vagrant от сообщества. Поэтому вы можете выбрать именно такую систему для создания собственного образа!

Сначала надо создать в графическом интерфейсе *VirtualBox* виртуальную машину под названием `windowsvm`. Если вы используете Windows 7 или выше, может потребоваться выделить не менее 1 ГБ (в идеале — 2 ГБ) оперативной памяти и создать виртуальный жесткий диск объемом 25 ГБ. Не стесняйтесь добавлять дополнительные ядра и необходимый объем видеопамати.

Вам будет нужно выполнить обычные действия по установке. После запуска Windows необходимо установить соответствующие пакеты обновлений и сервисов, а также «Дополнения для гостевых ОС» из меню Устройства *VirtualBox*.

Существует ряд дополнительных действий, которые помогут подготовить образ Vagrant наилучшим образом. Полный список рекомендаций, таких как отключение UAC, добавление определенных записей в реестр и полная настройка WinRM (собственный интерфейс управления Windows) доступен на <http://bit.ly/2vsLepQ>.

По завершении работы виртуальной машины вы сможете найти нужные файлы в каталоге `~/Virtualbox VMs/windowsvm`. Можете удалить все файлы с расширениями, отличными от `.vdi`, `.vbox` или `.vmdk`, поскольку единственное, что вам нужно — файл жесткого диска и базовая конфигурация.



➤ Если спасает только Windows, а *WINE* не обладает достаточной совместимостью для запуска проприетарного ПО, управление VM Windows с помощью Vagrant по крайней мере сэкономит ваше время и усилия.

Затем надо создать новый файл с именем `metadata.json`. Чтобы указать целевую платформу виртуализации для Vagrant, добавьте в этот файл строку

```
{ "provider": "virtualbox" }
```

Также необходимо создать в том же каталоге пустой файл с именем `Vagrantfile`. Для упаковки мы воспользуемся собственной командой Vagrant, поэтому вам не придется ничего настраивать вручную (например, MAC-адреса). Теперь надо перейти в домашний каталог и выполнить команду

```
$ vagrant package --base windowsvm
```

Как только экспорт завершится, вы сможете добавить образ в текущую установку Vagrant и запустить его в первый раз.

```
$ vagrant box add package.box --name windowsvm
```

```
$ mkdir windowstest
```

```
$ cd windowstest
```

```
$ vagrant init windowsvm
```

```
$ vagrant up
```

Если всё пройдет хорошо, Vagrant создаст совершенно новую виртуальную машину на основе предыдущей. Чтобы увидеть рабочий стол Windows, надо включить графический интерфейс в `Vagrantfile`, как мы делали для Fedora, но проверить работу образа можно и из самого *VirtualBox*. Если вы хотите избежать таймаута SSH, установите `OpenSSH` в базовый образ. Впрочем, таймаут вполне можно игнорировать, хотя это и не очень элегантно.

В графическом интерфейсе *VirtualBox* надо выполнить некоторые действия вручную, например, настроить общие папки или изменить распределение аппаратных ресурсов. Кроме того, команда `halt` эквивалентна отключению питания виртуальной машины без ее выключения. Тем не менее, в этом образе всё равно можно запустить любые программы Windows, без которых вы не можете обойтись. **LXF**

Скорая помощь

Возможности Vagrant можно расширить с помощью плагинов. Плагины пригодятся, когда вам надо использовать новые технологии виртуализации, ускорить установку пакетов или сократить объем необходимой настройки. Подробную информацию о возможностях плагинов и о том, как их установить, см. на <http://bit.ly/2tfl4Kp>.

А где WINE?

Тем, кто не знаком с *WINE*, скажу, что *WINE* — это уровень эмуляции, который находится между программой Windows и Linux и выполняет преобразование команд между этими системами в режиме реального времени.

Вы можете установить вспомогательные библиотеки и настроить параметры совместимости для отдельных приложений с помощью различных «префиксов», которые представляют собой отдельные каталоги установки.

К сожалению, поскольку сообщество не имеет доступа к исходному коду Windows, это далеко не идеальная система. Хотя такие приложения, как *Microsoft Office*, *Internet Explorer* и *Adobe Creative Suite*, поддерживаются всё лучше, конкретные случаи могут различаться. Вероятность успешного запуска приложения можно посмотреть на <https://appdb.winehq.org>.

Еще один недостаток заключается в том, что *WINE* не изолирована от основной системы, и ваша

домашняя система может быть повреждена вредоносными программами, предназначенными для Windows.

Запуск Windows в виртуальной машине, можно избежать проблем с совместимостью и запускать приложения в полностью изолированной среде. Однако для этого потребуются больше памяти и вычислительной мощности, поскольку в дополнение к вашему приложению вы используете полноценную операционную систему.

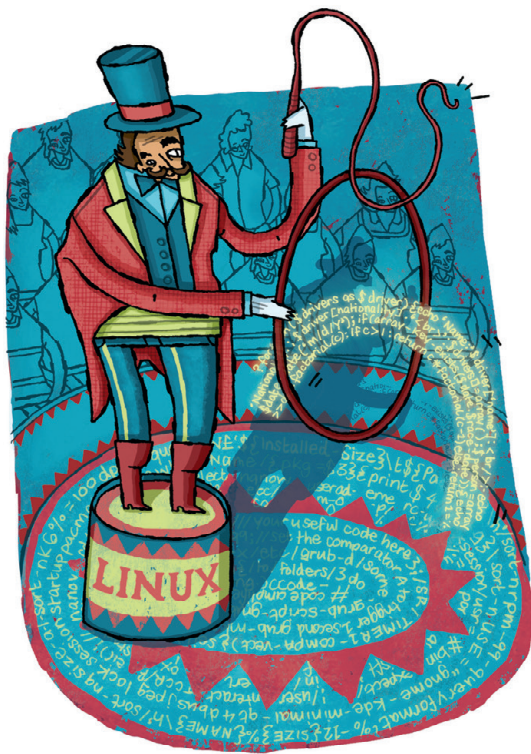
LineageOS: Ставим Android

Скрипнув зубами, **Алекс Кокс** оживил свой старый, скверно обновившийся телефон с помощью новенькой, открытой версии мобильной ОС от Google.



Наш эксперт

Алекс Кокс
 Ныне технический писатель, задержанный папаша и мастер-ломастер, Алекс расковыривает свое «железо» с тех пор, как разобрал на винтики свой ZX Spectrum.



```

Lxf@Lxf-VirtualBox: ~/platform-tools
...
(bootloader)
(bootloader) < Please cut following message >
(bootloader) <<<< Identifier Token Start >>>>
(bootloader) 5543919FC6109DB8745D3BA655349D24
(bootloader) 01E00B7588F1D08DF230598B28884237
(bootloader) EAF62F327BF2AF8BF09E693A6FE479A
(bootloader) 84BC848FEA2E94211601EDCDB11F463
(bootloader) 5803D7513C6F1C86919D272F2F54A0CB
(bootloader) B21C60AE68EBC94935B8EEC74CD7BA
(bootloader) E5885E24046C2C8987EB136A1886CD6
(bootloader) A78E5D805839FC0853684F022D207F3
(bootloader) 38F76D5942EF8CD2ED660BD128E6273
(bootloader) 2E8F9894D96AA9D0C2588DA72695EBC4
(bootloader) 26E889210C86A1430EF21A4F716773CC
(bootloader) 29B3CFE99817ADA47CE3D8940DC59E2
(bootloader) 25E42A345ED983C4286097C4D03073A2
(bootloader) 892A272967023E7537F22EEB1FA7641E
(bootloader) 37E7B1861359189ED538A0868C980AC4
(bootloader) F7392CA68FF8E1DC51EF757480EF104
(bootloader) <<<<< Identifier Token End >>>>>
OKAY [ 0.076s]
finished, total time: 0.076s
Lxf@Lxf-VirtualBox:~/platform-tools$
    
```

➤ Если ваш терминал позволяет выделять текст от и до, вырезать жетон для разблокировки устройства, полученный с помощью `fastboot oem get_identifier_token` — проще простого.

ощутимые изменения, в плане улучшения работы батареи, большей приватности общения и меньшей «прожорливости» по ресурсам (бич большинства телефонных прошивок). И эта система была разработана теми, кто любит свои телефоны и хочет сделать их лучше.

Однако в установке LineageOS на обычный Android есть свои минусы. Недавно Google внедрили в своем магазине Google Play систему SafetyNet, чтобы разработчики приложений могли убедиться, что их система «в полном порядке». Разработчики Lineage, дабы избежать неприятностей, разумно решили не обходить эту проверку, а значит, некоторые приложения могут быть недоступны через традиционные каналы, хотя, как правило, обходные пути всё же есть — поищите в Интернете файлы `.apk`, если хотите установить что-либо без использования магазина Google Play.

Базовые требования

Для установки LineageOS вам, очевидно, понадобится совместимый телефон или планшет. Полный список можно посмотреть на wiki.lineageos.org/devices/ — если у вас достаточно современный телефон, прошивку вы отключите без труда; но не ждите, что вам удастся перейти на более новую версию Android, чем та, что поддерживается вашим устройством.

Более старые устройства, увы, не подойдут, но следуя советам нашего урока, вы сможете установить старую сборку Suzyogelmod, если вам удастся ее найти.

Также на вашем ПК должен быть ADB — Android Developer Bridge. См. инструкции в разделе «Установка ADB», вверху стр. 63; это не особо сложно. Разумеется, вам также нужна правильная версия LineageOS для вашего устройства. Здесь мы устанавливаем

Денги решают всё. В случае супер-популярной ветки Android OS, Suzyogelmod, денги сменили философию родительской компании и стратегию разработки ПО с открытой политики лицензирования на коммерческую, а также и управляющего директора и руководителя проекта — причем неясно, сам он ушел или поневоле — и, в конце концов, нехватка всё тех же денег заставила корпорацию Suzyogel Inc. резко ограничить свою деятельность и полностью отказаться от поддержки Suzyogelmod. Сообществу же, активно вовлеченному в работу над проектом, пришлось перенять опеку над своим детищем и окрестить его заново. И теперь у нас есть последняя ветка Suzyogelmod и его открытый последователь, LineageOS, созданный отделившимся сообществом при огромном вкладе от бывшего главы проекта Suzyogelmod, Стива Кондрика [Steve Kondrik]. Имя пришлось сменить — в коммерческой среде же повсюду товарные знаки — но номер версии первого релиза Lineage, 14.1, напрямую наследует Suzyogelmod, и кодовая база практически не изменилась.

Прежде чем объяснять — как, давайте разберемся — почему... LineageOS — идеальный вариант для всех, кто хочет управлять своими телефонами и планшетами или экспериментировать с ними. От Android эта ОС не отличается просто потому, что разработчикам нравится возиться с мобильной ОС от Google; но внесены были

Скорая помощь

ADB не силен в исправлении ошибок, и если вы употребите немножко болтающийся USB-кабель или USB-удлинитель, могут возникнуть проблемы с передачей файлов. Пусть ваш кабель будет коротким и надежным.

Установка ADB

Android Debug Bridge, или ADB, необходим для низкоуровневой работы с операционной системой вашего телефона, а *fastboot* — для работы с загрузчиком.

Откройте окно терминала на компьютере с 64-разрядной ОС (32-разрядные более не поддерживаются) и загрузите архив с инструментами, где есть они оба, с помощью `wget https://dl.google.com/android/repository/platform-tools-latest-linux.zip`, и разархивируйте его с помощью `unzip platform-tools-latest-linux.zip`. Теперь отредактируйте файл профиля пользователя *bash*, запустив `sudo nano ~/.profile` — добавьте к нему следующие сведения,

чтобы обеспечить доступ ко всем командам *adb* и *fastboot*:

```
if [ -d "$HOME/platform-tools" ]; then
export PATH="$HOME/platform-tools:$PATH"
fi
```

Перезагрузите компьютер, и на этом с работой из Linux должно быть покончено — теперь наступает время настраивать ваш смартфон либо планшет.

На устройстве откройте Настройки > О программе, а затем найдите в номер сборки Android [Android build number]. Коснитесь его семь раз, затем вернитесь на главный экран настроек; внизу должна

появиться новая опция — параметры разработчика [Developer options].

Выберите ее, прокрутите вниз, а затем включите отладку по USB. Подключите свой смартфон к компьютеру и установите соединение; введите *adb devices* в окне терминала и нажмите ОК на смартфоне, чтобы разрешить соединение.

Если всё правильно, при повторном запуске *adb devices* ваш смартфон должен отображаться как «устройство», а не «неопознан», и вы также сможете выполнить *adb shell*, чтобы запустить отдельную оболочку Linux и управлять этим устройством. Всё просто.

прошивку, подразумевая, что она привязана к конкретному оборудованию внутри вашего устройства; никакой универсальной версии, годной для любого телефона или планшета, не существует.

Можно создать свой собственный пакет — это, конечно, лучший способ получить самые свежие функции LineageOS; а можно упростить процесс, загрузив готовое ПЗУ (ROM) с сайта <http://downloads.lineageos.org>. Если вы решили действовать самостоятельно, на вики-странице Lineage есть руководство для конкретных типов оборудования.

Наконец, вашему устройству понадобится разблокированный загрузчик. Здесь порядок действий полностью зависит от устройства, где вы будете устанавливать; для нашего урока мы взяли HTC One M7, что и будет нами отражено; не исключено, что на вашем устройстве загрузчик уже разблокирован, а если нет — методика для этого может быть совсем иной. Опять же, обратитесь на вики Lineage для получения конкретных инструкций или, в крайнем случае, попробуйте инструмент *SunShine* (<http://theroot.ninja>), если вам попался особо упёртый производитель.

Быстрый путь к свободе

Прежде чем перейти прямо к делу, давайте еще раз оговоримся, что на нашем уроке мы будем устанавливать LineageOS на HTC One M7; на вашем устройстве для этого может потребоваться меньше шагов, или всё может оказаться гораздо сложнее. Есть и риски: см. раздел «Будьте осторожны» (стр. 64), по поводу их смятения.

Во-первых, надо перевести устройство в довольно уязвимое состояние — а именно, активировать режим Fastboot. Fastboot — это протокол, используемый для записи данных прямо во флеш-память вашего телефона; там-то и будет установлена наша новая прошивка. Разумеется, всё не так просто, но давайте перезагрузим ваш смартфон в соответствующем режиме, введя в терминале *adb reboot bootloader*. Когда это будет сделано — вы сразу догадаетесь, поскольку на дисплее вашего телефона появится масса загадочной информации о конкретных версиях конкретных приложений, и где-то между всем этим — “Fastboot USB”. Затем убедитесь, что Linux всё еще видит телефон, набрав *fastboot devices*. Не исключено, что, в зависимости от настроек вашей системы, вам надо быть суперпользователем; если это так, выполните

```
cd ~/platform-tools
sudo ./fastboot devices
```

Разблокирование

Еще одна вещь, которую вы, вероятно, увидите на экране вашего устройства — это слово “locked [заблокировано]”. Загрузчик, то есть код, запускаемый при первом включении, как правило, заблокирован в подавляющем большинстве устройств, по очевидным причинам: большинство производителей предпочитают

избегать технических проблем и дискуссий о гарантии, возникающих, когда обычный пользователь начинает загружать собственные прошивки. С другой стороны, многие производители охотно позволяют разработчикам делать свое дело. Получив доступ к *fastboot*, мы сможем раздобыть жетон разблокировки нашего устройства:

```
fastboot oem get_identifier_token
```

В нашем случае надо перейти на сайт HTCDev для регистрации в качестве разработчика, а затем посетить сайт <http://htcdev.com/bootloader>, чтобы превратить свой жетон в ключ. Прежде чем перейти к процессу, сайт отображает довольно грозное предупреждение, и это действительно последний шанс отступить: потом ваше устройство будет принудительно перезагружено. Следуя инструкции, скопируйте жетон разблокировки с терминала и вставьте его в соответствующее поле, удалив из текста фрагменты “(bootloader)” или “INFO”, чтобы это был один чистый блок, и оставив без изменения идентификаторы начала и конца. Если всё прошло хорошо, ключ разблокировки устройства отправится на ваш зарегистрированный адрес электронной почты. Сохраните ключ в каталоге `~/platform-tools`, где находится ваш инструмент *fastboot*, затем вернитесь в свой терминал, перейдите в каталог `~/platform-tools` и введите следующее (изменив имя файла, если производитель вашего устройства отправил вам нечто иное), чтобы начать разблокировку:

```
unlock_okode.bin
```

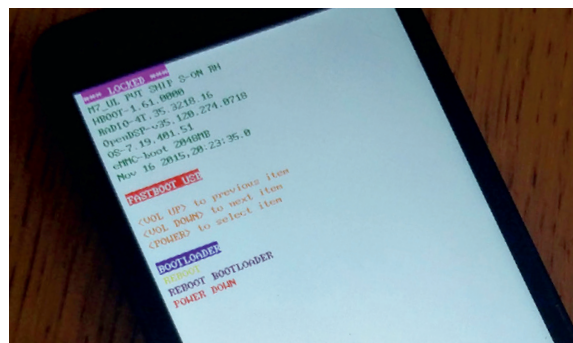
Появится еще одно предупреждение, на сей раз на экране вашего устройства. Это снова ваш последний шанс отступить: когда вы сделаете соответствующий выбор, произойдет сброс к заводским настройкам. После этого вернитесь в настройки и включите режим разработчика и отладку USB, как раньше.

Режим восстановления

Итак, мы открыли доступ к вашему загрузчику, и надо обмануть ваше устройство, убедив его, что произошел некий сбой — в режиме

Скорая помощь

Хотите просто использовать Linux на вашем телефоне без всей этой мороки? Установите Debian noroot из магазина Google Play — и сможете запускать настольные приложения на мобильных устройствах.



» Экраном быстрого доступа, как правило, приходится управлять кнопками питания и громкости, так как драйверы сенсорного экрана он загружать не будет.

восстановления телефон как никогда готов принять новую, предположительно рабочую, прошивку.

Для начала вернитесь к загрузчику своего устройства, нажав `adb reboot bootloader`, и убедитесь, что его `fastboot` теперь разблокирован. Теперь нацельте свой web-браузер на адрес <http://twrp.me> и получите версию для своего устройства. TWRP — это специальный пакет восстановления, который заменит предустановленный на вашем смартфоне или планшете — что необходимо, поскольку системы восстановления обычно по умолчанию привязаны к официальным прошивкам. Скопируйте файл в каталог `~/platform-tools`, откройте свой терминал в этом каталоге и введите следующее, изменив имя файла на соответствующее вашему устройству:

```
sudo ./fastboot flash recovery twrp-3.1.1-0-m7.img
```

Теперь надо перезагрузить устройство и ввести его режим восстановления вручную. На нашем тестовом телефоне мы делаем это, удерживая при загрузке кнопки питания и громкости; у вас может быть по-другому. Вы увидите экран, очень похожий на экран `fastboot`, который мы загрузили ранее; среди кнопок, показанных на экране, выберите опцию Восстановление [Recovery], и как только TWRP загрузится, используйте ползунок Проведите по экрану, чтобы разрешить модификации.

Запуск прошивки

При активном TWRP введите в терминале `adb devices`, для проверки, что ваш компьютер видит ваше устройство — должно быть указано, что оно находится в режиме восстановления. Получите соответствующую сборку LineageOS для своего телефона или планшета с сайта <http://download.lineageos.org> и загрузите с <http://opengapps.org> пакет приложений Google (по имени `gapps`) в соответствии с архитектурой вашего устройства, версии LineageOS, которую вы хотите установить, и желаемому количеству приложений;

если вам требуются только самые основные элементы, выберите 'nano', а если вам нужен весь спектр приложений, поставляемых с линиями Nexus и Pixel, выберите 'Stock'. Есть также промежуточные варианты: нажмите 'variant' в верхней части правого столбца, чтобы увидеть таблицу сравнения.

Установка приложений Google не является обязательной — скоро мы подробно рассмотрим альтернативу с открытым кодом; но может случиться так, что функциональность вашего Android-интерфейса без магазина *Google Play* будет довольно ограниченной. Стоит отметить, что OpenGApps размещает свои файлы на ужасных серверах, так что загрузка может потребовать некоторого времени и терпения; нам пришлось пару раз применить `wget -c` для возобновления загрузки. Поместите эти файлы в каталог `~/platform-tools` и загрузите их на свое устройство:

```
adb push lineage<filename>.zip /sdcard/  
adb push open_gapps<filename>.zip /sdcard/
```

Заметьте, что мы загружаем их в каталог `/sdcard/`; и это неизменно, даже если на вашем устройстве физически нет слота для карт SD — так уж устроена раскладка файлов Android. По сути, `/sdcard/` является обозначением «внутреннего хранилища»; если вы обнаружите, что процесс `adb push` не работает (а, как известно, на определенных устройствах он не работает), вы можете просто копировать файлы с помощью диспетчера, особенно если в вашей системе он обнаружен и смонтирован как диск.

Вернувшись на свое устройство, вы можете, по желанию, создать резервную копию своего внутреннего хранилища в текущем состоянии. Это удобно, на случай если вам понадобится вновь вернуться к базовой прошивке, поскольку не все производители ее предоставляют.

Когда вы будете готовы к установке, перейдите в меню Wipe и выберите Advanced Wipe. Проверьте соответствующие поля и сдвиньте ползунок, чтобы полностью очистить разделы Cache [Кэш], Data [Данные] и System [Система], затем вернитесь в главное меню и выберите Install [Установить]. Во внутреннем хранилище выберите ZIP-файл LineageOS и сдвиньте ползунок, чтобы начать прошивку. На данном этапе наберитесь терпения: вам может показаться, что ничего не происходит, но прервав процесс, вы можете угробить устройство.

Теперь, прежде чем перезагрузиться, повторите тот же процесс с файлом приложений Google, если вы его используете. Здесь процесс установки более прозрачный, так что вы увидите, как устанавливается каждое приложение.

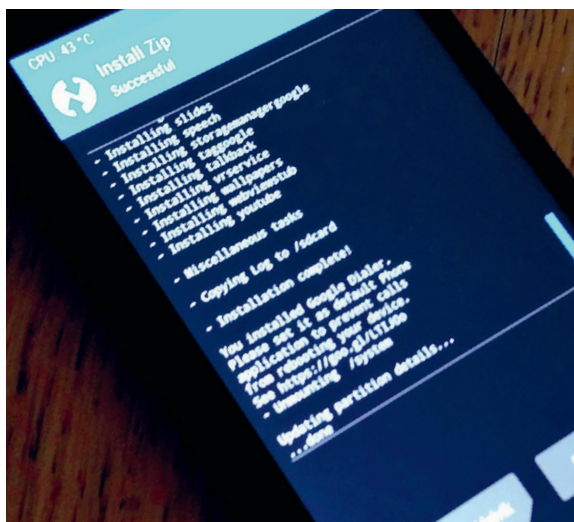
Получение прав root

Не торопитесь с перезагрузкой. Для продвинутых пользователей есть еще шанс получить root-доступ к своей системе. По умолчанию в LineageOS такой функции, разумеется, нет — из соображений безопасности: несанкционированная установка приложений несет вашей системе большой риск. Кроме того, это нужно не всем; но зато с такой функцией ваши возможности становятся гораздо шире.



Скорая помощь
LineageOS — не единственная альтернатива Android; эксперты с <https://forum.xda-developers.com> определенно смогут порекомендовать вам еще с десяток.

➤ Инструменты восстановления, подобные TWRP, принимают правильно созданные ZIP-файлы и запускают скрипты установки, когда вы приступаете к загрузке ПО на свое устройство.



Будьте осторожны

То, что мы здесь делаем, абсолютно разрушительно. Установка новой прошивки на телефон означает, что ваша старая прошивка, вместе с запасной ОС, поставляемой вместе с устройством, и всем, что вы к ней добавили, будут стерты, и хотя по ходу мы будем делать резервные копии, нет никакой гарантии, что туда войдет всё. Если вы перешиваете свой основной телефон, это значит, что вся ваша телефонная книга, все фотографии будут

потеряны. И если что-то пойдет не так, есть шансы вообще убить свое устройство, то есть превратить телефон в кирпич. Мы нисколько не пытаемся отпугнуть вас, но такие вещи хорошо бы осознавать и принимать правильные меры предосторожности.

Прежде всего, убедитесь, что вы создали полную резервную копию своего устройства. Через USB вы сможете получить доступ к внутреннему хранилищу, благодаря которому вы сможете

вручную создавать резервные копии своих личных файлов. Если вы вошли в систему с учетной записью Google, возможно, стоит также доверить свои данные облачным сервисам. Например, с помощью Google Contacts вы можете управлять своей телефонной книгой и вернуть ее на устройство после смены прошивки, а Google Photo позволяет хранить неограниченное количество фотографий в меньшем разрешении.

Открытая альтернатива

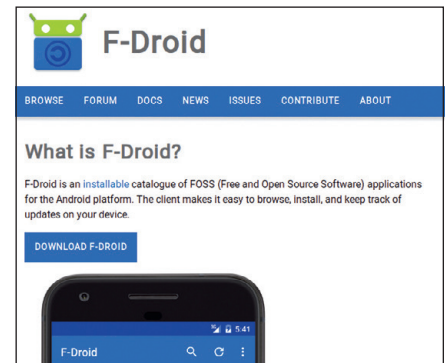
Может показаться, что устанавливать открытую ОС и тут же приложения Google — это контрпродуктивно; «открытая» часть OpenGApps относится к самому проекту, а не к приложениям, и если для вас поводом обращения к нему была именно свобода, а не обновление телефона, то сразу же уничтожить ее установкой проприетарного пакета — крайне неразумно.

К счастью, эту философию разделяют многие, и в составе проекта F-Droid есть стек приложений Open Source.

F-Droid — это, по сути, хранилище для открытого ПО, которое вы можете использовать для установки множества непатентованных приложений. Вы не найдете его в Google Play, по очевидным расхождениям в плане философии, но доступно оно каждому, даже если вы используете

стандартный телефон Android или уже установили приложения Google.

Если вы установили LineageOS и пропустили этап установки приложений Google, у вас есть кое-какой выбор открытых приложений, в том числе файловый менеджер. Активируйте ADB на своем телефоне — переключатель находится в настройках разработчика — затем, через приложение терминала, запустите `apk`, установщик F-Droid, (с <https://f-droid.org>) на ваш телефон, в каталог `/sdcard/storage`. Откройте Настройки > Безопасность и активируйте Неизвестные источники [Unknown sources], чтобы включить установку приложений в формате APK. Теперь зайдите в диспетчер файлов и нажмите на файл APK, который расположен, на самом деле, в корневом каталоге вашего устройства (чем и является `/sdcard/directory`), чтобы его установить.



➤ В F-Droid широкий выбор приложений для любых целей, выпущенных под различными свободными лицензиями.

Вы можете скачать инструмент суперпользователя LineageOS с сайта <http://download.lineageos.org/extras>. Просто выберите версию для своей ОС, загрузите ее на телефон, как раньше, и запустите с помощью TWRP; это займет больше времени, чем можно было бы ожидать от такого небольшого файла, но там много изменений на внесение изменений на устройстве.

Если позже вы решите отключить функцию суперпользователя, соответствующий инструмент вы найдете там же: нужно только загрузиться в режим восстановления и запустить его.

Первый запуск

Наконец, пора перезагрузиться. Вам предложат установить в вашей системе приложение TWRP — это полностью на ваше усмотрение; но поскольку теперь вы используете устройство с индивидуальной прошивкой, TWRP может сделать работу с восстановлением гораздо удобнее, так что это неплохая идея.

Теперь предстоит очень долгий процесс начальной загрузки, во время которого LineageOS полностью проверит внутреннее хранилище вашего устройства, с целью убедиться, что всё готово к работе. Будьте терпеливы, дождитесь, пока на экране появится анимация запуска, и вы попадете в стартовый процесс настройки пользователя. Он очень похож на стандартный Android и позволяет при желании восстановить данные с другого устройства. Вам также предлагается войти в сервисы Google: если вы не используете приложения Google или магазин Google Play, можете это смело пропустить.

Справившись с Android-частью, надо будет выполнить ряд специальных настроек LineageOS — мы бы посоветовали включить Privacy Guard, по умолчанию отключенный; с ним вы можете напрямую контролировать, какие приложения данных на вашем

LineageOS Downloads			
Extras			
Name	Version	File	Date
su (arm)		addonsu-14.1-arm-signed.zip	
		sha1 sha256	
su (arm64)	14.1	addonsu-14.1-arm64-signed.zip	2017-06-21
		sha1 sha256	
su (x86)		addonsu-14.1-x86-signed.zip	
		sha1 sha256	

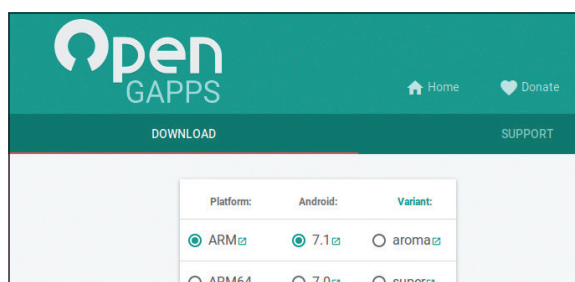
➤ Применив инструмент суперпользователя Lineage, после выполнения первоочередных задач можете удалить его — инструмент удаления вы найдете там же.

устройстве разрешены для доступа. Вы можете изменить свои настройки позже в меню Settings > Privacy [Настройки > Конфиденциальность], и если какое-то приложение вдруг слишком раздуется, вы сможете уменьшить его размер. Сделав всё, что нужно, вы попадаете в систему — с чистым обновленным устройством, где нет никакого мусора от производителя. Если вы не установили Google Apps, у вас есть кое-какие альтернативы Lineage с открытым исходным кодом, чтобы поддерживать всю требуемую функциональность.

Добавочные опции

Если вы установили пакет root, надо его включить. Войдите в приложение Настройки и нажмите на Build number [Номер сборки] семь раз, как мы это делали раньше, чтобы перейти в режим разработчика. Вернитесь на экран основных настроек и войдите в соответствующее меню. На данном этапе мы рекомендуем включить Advanced Restart — это позволяет быстро загрузиться в режим восстановления или загрузчика с помощью стандартной процедуры перезапуска — затем прокрутите вниз до опции доступа root и выберите уровень доступа, который вы готовы предоставить. В ADB он ограничивается только командами, отправляемыми через Android Developer Bridge, и это означает, что вы можете получить root-доступ (и корневую оболочку) к своему устройству, но не к приложениям.

В режиме разработчика опций гораздо больше, чем на стандартном Android. Можно добраться до самых мелких деталей и задать для своего устройства некоторые жесткие правила. Например, активация опции Не сохранять действия [Don't keep activities] идеально подойдет, если вам нужен тотальный контроль над вашими приложениями, а включить локальный терминал [Enable local terminal] добавляет новое приложение, которое дает вам постоянный доступ к вашему устройству из терминала, если ранее вы предоставили root-доступ к приложениям. LXF



➤ Не знаете, какие приложения Google установить на новый телефон? Загрузите вариант Aroma: интерактивный установщик позволит выбирать их по одному.

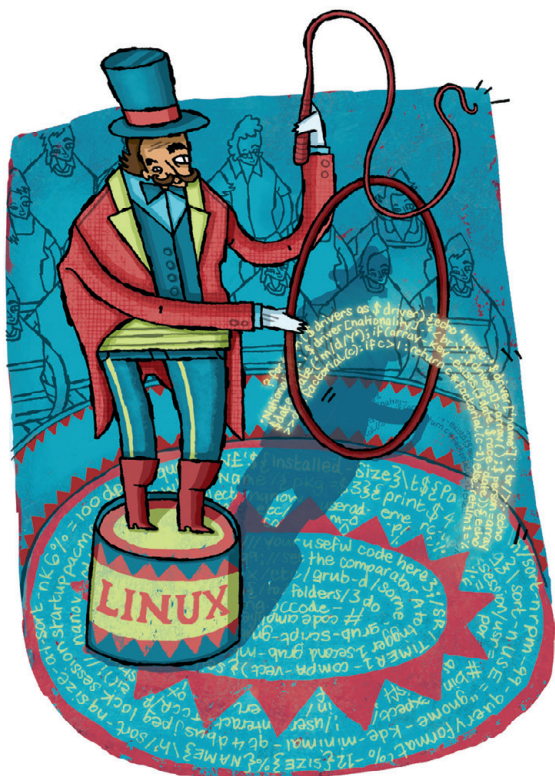
Audacity: Запишем подкасты

Хотите довести свои слова до всего мира? Адам Оксфорд объясняет, как создать идеальный подкаст с помощью простого редактирования в Audacity.



Наш эксперт

Адам Оксфорд — журналист, живущий в Южной Африке. Помимо создания мультимедиа-продуктов с помощью инструментов с открытым кодом, он занимается обучением журналистов безопасности онлайн и подаче новостей на основе данных.



Здоровый совет

Первое, чему учатся продюсеры и журналисты мультимедиа — что зритель прощает куда большее, чем можно подумать. Раньше мы учились у профессионалов, обученных в BBC, которые считали каждую нарезку и редактирование делом первостепенной важности; режиссеров видео, которые были готовы скорее скрестить мечи, чем «перейти черту»; радиопродюсеров, у которых плохо настроенный микрофон вызывал мигрень.

А потом появились Интернет и iPods, доказавшие, что люди вообще-то отдадут предпочтение дешевизне и удобству, а не качеству. Наши экраны заполнили снятые на трясущиеся камеры видео с YouTube, и пускай MP3-ненавистник Нейл Янг [Neil Young] пытался нас убедить, что это плохо, но большинство из нас не отличит аудиотрек 44,1 Гц от подделки, если положится исключительно на собственный слух.

Аудитория стерпит мозаику пикселей в видео и музыку со сжатием, но не невнятный или плохо записанный звук. Слишком сильный фоновый шум, искажения, ограничения сигнала или нечто подобное, неумышленно отвлекающее внимание от произносимых слов, приведет к тому, что подкаст просто выключат и удалят со смартфона или планшета — быстрее, чем непрошенный альбом U2.

Вы предупреждены. Вот вам серьезная причина, почему Netflix при более медленном соединении снижает качество видео, но качество аудио при этом держит на высоте.

Это не означает, что для записи своего первого подкаста вам придется раскошелиться на полноценный микшерский пульт с 16 дорожками и звуконепроницаемой студией. Можно записать вполне терпимое интервью на рок-фестивале даже на мобильный телефон; однако намного проще и надежнее сделать это с достойным качественным набором инструментов в тихом помещении, желательно в таком, где микрофон не будет ловить эхо от стен.

Существует куча базовых онлайн-руководств по наборам инструментов и подсказок по записи, однако есть два золотых правила, позволяющие улучшить всё, что вы будете делать.

Первое — всегда, всегда надевать наушники. Всегда. Когда вы записываете, когда вы редактируете, когда прослушиваете заставку. Это единственный способ услышать всё, что записывается. И потратитесь на хорошие наушники. Помимо того, что это улучшит ваши навыки редактирования, из-за плохого качества динамиков вы можете пропустить щелчок или плохо отредактированный фрагмент — они этого просто не уловят; а если ваша аудитория слушает в хороших наушниках или даже наушниках-капельках, они сразу заметят все проблемы, которые вы упустили. Если у вас ограниченный бюджет, советуем потратиться на наушники высшего качества, а микрофон купить просто достойный. Ваша аудитория будет вам за это благодарна.

Только представьте: вдруг раз в жизни вам выпадает шанс взять интервью у любимого писателя о Linux, и, услышав странный

Serial, *Freakonomics*, Radio, *TWIT*, *My Dad Wrote A Porno*: скромные подкасты входят в своего рода золотой век, предлагая большое разнообразие и качество аудио. Но несмотря на то, что подкасты — один из старейших и самых стойких форматов Интернета, их жизнеспособность в эпоху Facebook, YouTube и Twitter одно время была под вопросом. Ну откуда у современной аудитории терпение на аудиодокументалки? Сможет ли радиомор пережить убойные видео с котятками?

Фанатам можно больше не беспокоиться. В прошлом году только в США количество слушателей подкастов выросло на 23%. Как ни странно, несмотря на то, что для восприятия подкасты труднее более коротких видеоформатов, Edison Research полагает, что аж 85% передач прослушивается от начала до конца.

Обратите внимание!

Это такой обходной путь сказать, что если вы любите слушать подкасты и подумываете о создании собственного, сейчас самое время это сделать. Мировая аудитория жаждет шоу, а создать качественный подкаст не так уж сложно. Фактически, вы можете начать с надежного и привычного настольного аудиоредактора, Audacity — и на нашем уроке мы покажем вам, как задействовать его по максимуму.

Скорая помощь

Если вы записываете интервью в условиях, отличающихся от идеальных, может очень пригодиться фоновый трек. Добавьте за главным треком звук по своему желанию — музыку или пение птиц — и он замаскирует нежелательные шумы.

Основной инструментарий

Мы показали, что можно вести запись с нескольких микрофонов прямо в *Audacity*, однако нельзя обойти тот факт, что микшерский пульт не только проще в настройке, но и дает большую степень контроля над уровнями записи, и конечный результат будет более профессиональным. Хорошая новость в том, что хотя студийное оборудование стоит дорого, можно найти микшер на четыре дорожки от Gear4 всего за £40. У более дешевых микшеров может не быть такого количества XLR-входов для микрофонов, однако разумное использование переходников XLR-1/4" поможет справиться с этой проблемой.

Чем больше вы потратите на микрофоны, тем лучше, однако не думайте, что вам нужно тратить

сотни на профессиональный динамический микрофон, если вы только начинаете. Конденсаторный микрофон дешевле и, поскольку он менее чувствителен, будет захватывать меньше фонового шума, так что это — отличный способ освоить выбранное дело. Вам также понадобится подставка под каждый микрофон и, в идеале, фильтр для защиты от дыхания (это тонкая мембрана, которую вы видите перед студийными микрофонами; она помогает избежать шума от произношения смычных согласных звуков, например, звука «п»). Ее можно соорудить из вешалки и пары колготок).

Если вы подключаете микрофон прямо к компьютеру, для набора с небольшой стоимостью мы бы рекомендовали USB-микрофон. Использование

адаптеров для 3,5-мм гнезда на звуковой карте почти наверняка создаст помехи и искажение звука.

И, наконец, пара слов о том что делать с подкастом, когда вы его закончите. Вы можете просто выложить его на сервис типа Soundcloud, или загрузить на собственный сайт в виде медиа-файла; но если вы выберете специальный хост для подкастов, например, Buzzsprout (бесплатно) или Blubdry (\$12/месяц), то ваш подкаст окажется в iTunes и Stitcher, не заставив вас лезть из кожи вон и работать, чтобы в случае популярности вашего подкаста ваш личный сервер не упал под весом огромного наплыва слушателей. Кроме того, вы получите отличную аналитику о частоте загрузки вашего шоу.

фононый шум в момент, когда лысеющий красавец заговорит, вы сможете попросить его перейти в другое место. А если вы не заметите этот шум, пока не приметесь редактировать свое шоу, тут уже мало что поделаешь.

Второе золотое правило — изо всех сил стараться не позволять ведущим, гостям или корреспондентам перебивать друг друга или говорить одновременно. Даже такие прерывания, которые в разговоре возникают естественно, например, когда вы поддаете или хмыкаете в стиле Элвиса, соглашаясь с высказываемым мнением, в записи звучат ужасно. Если вы соглашаетесь с говорящим или хотите поощрить его говорить дальше, возьмите за правило кивать молча. Желая прервать говорящего, сначала подайте ему визуальный знак об этом.

Удачная Audacity

Мы можем на многих страницах рассуждать о технике хорошей записи, но это — технический журнал, и, честно говоря, лучший способ научиться — это попробовать.

Есть несколько программ для редактирования аудио в Linux, и по большей части они превосходны. Например, *Ardour 5* превращает ваш ПК в профессиональную аудиостудию и ничуть не хуже любой проприетарной программы для записи и мастеринга звука. Вы найдете *Ardour* в большинстве репозиторий по умолчанию, и его можно установить из командной строки или через центр программ в популярных дистрибутивах.

Однако новичка *Ardour* может запугать (да и относительно опытных тоже, честно говоря), и большая часть функций *Ardour* предназначена для профессионального мастеринга музыки. *Frinika* (<https://sourceforge.net/projects/frinika/>) является чуть менее пугающей версией Digital Audio Workstation (DAW), но всё же достаточно сложна и не поддерживает многие аудиоформаты.

Еще одна причина избегать этих пакетов, если вы только начинаете, в том, что звуковым сервером для большинства дистрибутивов Linux по умолчанию является *PulseAudio*, который в общем не очень-то ладит с более профессиональными пакетами. Те предпочитают более старый, но более стабильный сервер JACK (см. стр. 68); но чтобы его установить, придется выкосить *Pulse*, а это, в свою очередь, породит другие проблемы, и т. д.

Итак, если вы только начинаете, старая надежная *Audacity* будет отличной стартовой точкой для создания ваших собственных подкастов. Это не настоящая DAW, и ей не хватает парочки довольно важных функций для редактирования нескольких

дорожек и клипов, но ее легко освоить, и она способна выдавать отличные результаты.

Основное ограничение в том, что *Audacity* не слишком хорошо справляется с нелинейным редактированием. Здесь нет временной шкалы для перетаскивания клипов, и обходной путь состряпан на скорую руку (см. пошаговую инструкцию на стр. 69). Но если формат вашего подкаста — обычное обсуждение типа круглого стола, то у вас не возникнет особых проблем.

Однако прежде чем редактировать свою окончательную версию, подкаст вы должны записать. И это немного сложнее, чем кажется.

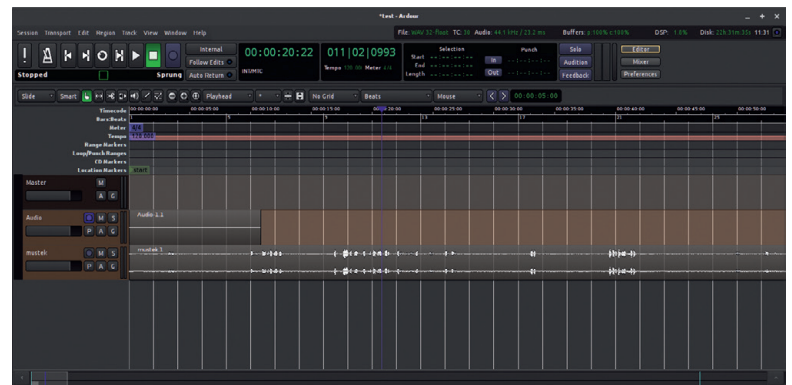
Запись аудио

Есть четыре типичных способа записи подкаста. Первый, и самый простой способ достижения профессиональных результатов — потратиться на приобретение отдельного устройства для записи (см. врезку *Основной инструментарий*, вверху), которое работает с несколькими микрофонами. Он также и самый затратный по стоимости.

Три остальных способа включают прямую запись на ваш ноутбук или ПК, и с ними разобратся чуть сложнее. Очень легко записать звук прямо в *Audacity*: вы просто открываете программу и нажимаете на большую красную кнопку записи. В выпадающем меню над временной шкалой трека вы можете выбрать источник ввода и динамик вывода; тот, что со значком микрофона, как раз и ведет запись.

Скорая помощь

Не рассчитывайте на то, что в первый же раз у вас получится отличное редактирование, особенно если вы вырезаете кусок из основного трека. Вы можете скопировать весь подкаст на вторую дорожку, попробовать два разных редактирования, потом использовать кнопку Mute, чтобы послушать одно за другим и посмотреть, какое лучше.



» *Ardour* — просто невероятный аудиопакет, но сложный в освоении для новичка.

» Подпишитесь на печатную или электронную версии на www.linuxformat.ru/subscribe!

Проблема в том, что вам нужно будет использовать отдельные микрофоны для каждого гостя — мы пробовали использовать эти ненаправленные пузырьковые микрофоны, и результаты не всегда хороши. Другая проблема в том, что *Audacity* может вести запись только от одного источника за один раз. А значит, если вы используете два микрофона, перед началом записи вам придется смикшировать их сигналы вместе.

Лучшей опцией для этого будет взять внешний микшерский пульт (см. врезку *Основной инструментарий* на стр. 67), к которому подключить микрофоны и затем направить один стереовывод на свой ПК. Настраивается это просто, и обойдется недорого, если выбрать пульт базовой комплектации. Он даст вам ручки физической настройки, чтобы изменять уровни и тон при записи; однако пульт не предназначен для подключения дополнительных инструментов.

В качестве альтернативы и последнего средства, если вам отчаянно нужно сделать запись, но вы не хотите тратить вообще ничего, можете подключить два микрофона напрямую к вашему ПК — вероятнее всего, у вас только одна звуковая карта, поэтому хотя бы один из них будет USB-микрофоном. Вдобавок USB-микрофон, скорее всего, будет лучшего качества, чем микрофон, подходящий для 3,5-мм гнезда на вашем ноутбуке. Смешивать ввод со всех микрофонов перед направлением его в *Audacity* будет довольно сложно, но не невозможно.

Самым профессиональным способом записи вашего подкаста будет использовать аудиосервер JACK, и вам стоит почитать о нем, если вы планируете превратить создание подкастов в привычное дело. JACK встраивается в реальные и виртуальные устройства практически так же, как аудиоинженер проводит кабели к разным инструментам и усилителям.

Вы даже можете установить специальный дистрибутив для аудио, как, например, Ubuntu Studio, который включает расширенные инструменты записи и ядро реального времени. Но для начинающего это будет перебором, поэтому мы ограничимся установленным по умолчанию сервером *PulseAudio*.

Pulse позволяет направлять звук только с одного физического ввода (микрофона) в виртуальный накопитель (*Audacity*). Но можно закольцевать два или более физических устройства ввода в одно (см. врезку *Хитрости Pulse внизу*).

Последний способ записи вашего подкаста — если ваши гости находятся в другом месте и вы беседуете по *Skype* (или другому приложению VoIP). Вы можете настроить канал закольцовывания, как сказано выше, и привязать его к своей программе вызова во вкладке *Recording*. Если вы используете VoIP-приложение на своем смартфоне, надо будет подключить устройство ввода с вашего компьютера к гнезду наушников на вашем телефоне.

Ваш первый подкаст

Теперь, когда вы можете записать свой подкаст, самое время его создать. Первое, что вам нужно сделать — «задать уровень». Это означает установить коэффициент усиления каждого микрофона: он не должен быть настолько тихим, чтобы не позволять слышать, что человек говорит, но и не настолько громким, чтобы голос искажался или «клипировался».

Откройте *Audacity*. Вверху справа вы увидите два горизонтальных индикатора; рядом с одним из них будет значок микрофона, а рядом с другим — динамика. Они показывают текущий уровень записи и воспроизведения и имеют цветовую кодировку зеленым, желтым и красным.

Нажав на верхний индикатор, вы можете следить за сигналом, который сервер *PulseAudio* направляет в *Audacity*. Основная идея при установке уровня — в поднятии динамиком этого уровня так, чтобы при воспроизведении речи индикатор достигал желтой области. В идеале вы просите кого-то что-нибудь сказать в микрофон и настраиваете уровень соответствующим образом. Но обычно, когда люди испытывают эмоции, они повышают голос и говорят громче; или при настройке уровня могут говорить отчетливее, чем обычно.

Поэтому вы должны присматривать за уровнями во время записи. Если они достигают красного уровня и выдают пик, это значит, что сигнал слишком громкий для записи. Если они падают слишком низко в зеленый цвет, значит, звук слишком тихий и его невозможно услышать. С помощью аппаратного микшера вы можете перемещать слайдер, чтобы оставаться в оптимальной зоне. Если вы используете *Pulse Mixer*, то вы должны уметь отлично работать мышью.

Вот и всё — можете записывать свой подкаст. По окончании не забудьте его сохранить, а затем перейдите к нашему пошаговому руководству, чтобы узнать, как его отредактировать. **LXF**



➤ Микшерские пульты не обязаны быть слишком сложными или дорогими. Самый обычный будет поддерживать до четырех микрофонов всего за £30.

Хитрости Pulse

Чтобы смикшировать в *Pulse* два микрофона, надо установить *PulseAudio Volume Control* (PAVP). В большинстве дистрибутивов вы найдете его в своем *Software Centre*, или можете установить его из командной строки по `sudo apt pavucontrol`.

Откройте PAVC и перейдите на вкладку *Input Devices* [Устройства Ввода]. Вы увидите отдельный звуковой монитор для каждого устройства. Теперь откройте терминал с помощью `Ctrl + Alt + T` и введите `pactl load-module module-null-sink sink_name=mixtrack`. Вы должны увидеть, что во вкладке

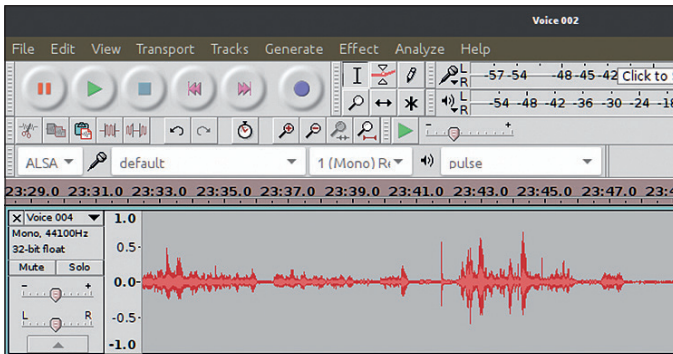
Input Devices появилась дополнительная строка под названием *Monitor of null outputs* [Монитор нулевого вывода]. Возможно, вам придется изменить меню *Show* внизу экрана на *All input devices* [Все устройства ввода].

Теперь надо что-то подать на канал монитора. Перейдите на вкладку *Recording*, которая должна представлять собой пустой экран. Затем введите `pactl load-module module-loopback sink=mixtrack` и нажмите `Enter`. Введите то же самое для всех имеющихся у вас микрофонов.

Вы должны увидеть, что во вкладке *Recording* появились новые мониторы, и у каждого вверху справа имеется выпадающее меню. Используйте эти меню, чтобы приписать каждый микрофон к каналу закольцовывания. Теперь во вкладке *Input Devices* вы должны видеть индикатор уровня звука,двигающийся в канале *Monitor of Null Output* для каждого микрофона — вы можете проверить его, слегка стукнув по каждому.

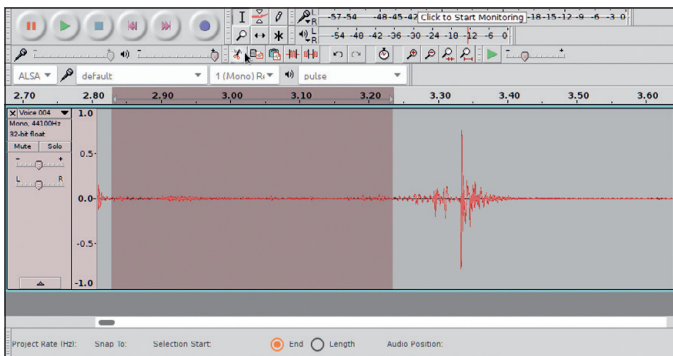
Именно этот канал будет подавать сигнал в *Audacity* по умолчанию.

Сшиваем всё вместе



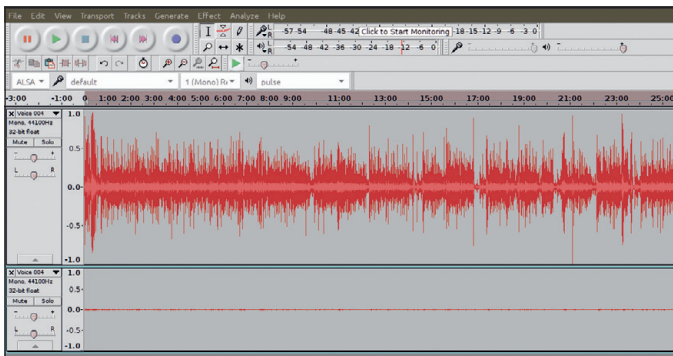
1 Импорт

Постарайтесь записывать свой подкаст как несжатый WAV-файл. Многие программы записи позволяют сразу переходить на MP3 (самые просвещенные могут также поддерживать OGG), но на этой стадии не надо вырезать из записи какие-то детали. Чтобы импортировать свой файл в Audacity, используйте File > Import или просто перетащите его в пустое окно. Если вы импортируете второй аудиоклип, он появится в качестве новой дорожки под исходной.



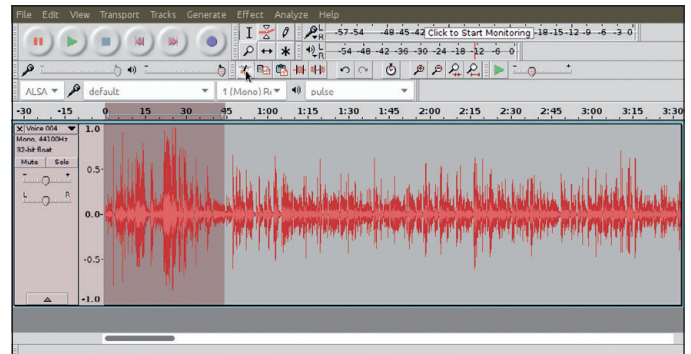
3 Базовый монтаж

Чтобы отрезать разговоры перед началом шоу, надо всего лишь выделить нужную часть файла и нажать на кнопку Отрезать (в виде ножниц). Однако хорошее редактирование несколько сложнее. Сигнал в окне трека замолкает, когда проходит отметку 0 в середине, и если вы сделаете обрезку в любом другом месте, кроме линии нуля, при воспроизведении аудио ваших слушателей раздосадует неприятный щелчок.



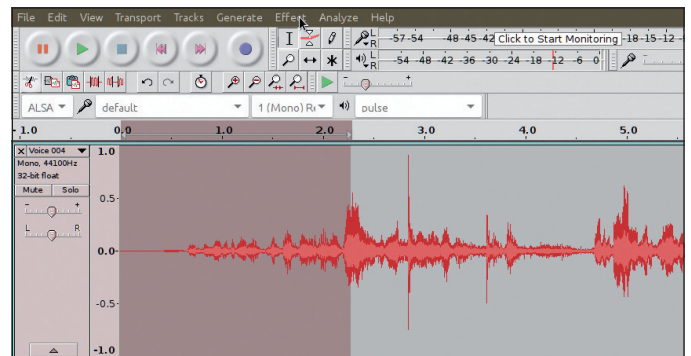
5 Вторая дорожка

Audacity — не слишком хороший нелинейный редактор, т. к. нельзя легко перетаскивать небольшие аудиоклипы по временной шкале; но и совсем ужасным его не назовешь. Если вы хотите добавить звуковой эффект или музыку за динамиком, просто импортируйте ее в качестве новой дорожки, затем выделите эффект курсором и нажмите Cut. Теперь можете переместить курсор и вставить его в любое место второй дорожки.



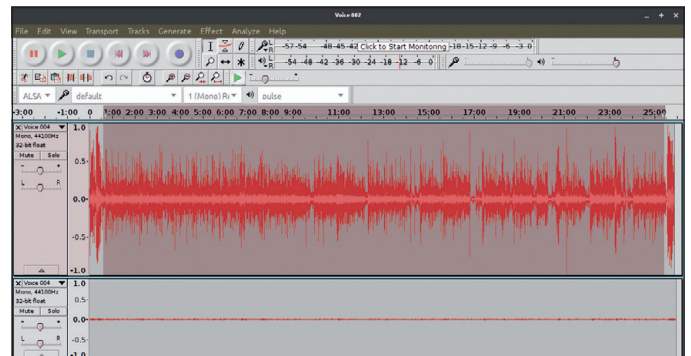
2 Сигнал

Основные кнопки управления при редактировании — это пять маленьких кнопок в середине верхней панели: Cut [Вырезать], Paste [Вставить], Copy [Копировать], Trim [Подстройка] и Silence [Тишина]. Если у вас есть несколько минут разговора, записанного перед началом шоу, надо отредактировать их с помощью Cut. Воспроизведите дорожку с самого старта, чтобы найти начало, и увеличьте масштаб сигнала с помощью Ctrl и колесика мыши, чтобы найти первые слова шоу.



4 Выкручивайтесь

Иногда обрезку на линии нуля сделать нельзя — например, за первым словом шоу не последовала тишина. Хорошим обходным путем будет выбрать полсекунды перед монтажом и перейти в Effect > Fade in [Эффекты > Затухание]. Это скроет неровности редактирования. Если вы применяете обрезку, чтобы убрать часть самого подкаста, щелчки можно сгладить затуханием и постепенным наращиванием звука.



6 Уровни

Надо будет поиграть с уровнями в фоновом треке, для проверки, что динамики не заглушаются. Выберите клип для редактирования, перейдите в Effect > Amplify [Эффекты > Усиление] и усиливайте либо ослабляйте их. Если вы увидите, что один человек говорит тише остальных, воспользуйтесь инструментом Compressor на основной дорожке, чтобы автоматически выровнять разницу между более тихими и громкими битами.

IoT: Дом, который послушен нам

Рамки физических условий комфортного существования человека довольно узки. Послушный Дом **Максима Черепанова** следит за температурой...



Наш эксперт

Максим Черепанов — заместитель директора в небольшой коммерческой фирме. По образованию железнодорожник-управленец. Linux занимается с 2008 г. Чем дольше это делает, тем дальше от Windows. За эти годы научился чистить реестр и забыл слово «антивирус».



Доброго всем дня, и послушного Послушного дома! Это не опечатка и не тавтология, это пожелания жить в своем жилище в гармонии с ним.

В одной из первых статей я высказал мысль, что воспитание (или выращивание) ПД чем-то сродни воспитанию ребенка. Сначала он ползает, потом сидит, потом ходит, бегает... Начинает говорить, и вы получаете от него его мысли и ощущения, выраженные словами. Это щемящее и сладостное чувство, когда можно сказать: «Доченька, выгляни во двор, там тепло или холодно?» — «Прохладно, папа!»

Нет, у нас будет не робот Реррег, созданный компанией мобильной связи Softbank Corp. Но тем не менее, очень удобно знать температуру за окном. Или находясь на работе, получать уведомления по электронной почте или СМС о том, что температура в квартире ниже или выше комфортной, и что включен обогреватель/кондиционер.

Почему именно температура, а не влажность, освещенность, напряженность магнитного поля? Потому что температура — это главный критерий для человека, показатель пригодности для проживания окружающей среды. Вспоминайте: когда вы слышите по телевизору, что астрономы открыли новую планету земного класса с температурой на поверхности от... И тут ваш слух обостряется, и вы слышите «От -80 до -50 градусов по Цельсию». Всё,

интерес пропадает: там жить невозможно. А вот если около 25°C, то это уже совсем другое дело!

Хватит лирики. Убедил? Тогда приступим к работе.

«Сколько вешать в граммах?»

Чем и как будем измерять? На теорию налегать сегодня не будем, а вот немного истории и экскурс в развитие технического прогресса я себе позволю.

Все мы знаем, как измеряют температуру бытовые термометры. Наиболее распространены 3 вида: термометры расширения (ртутные и спиртовые), anerоиды и цифровые. Первые показывают температуру на шкале столбиком жидкости, которая расширяется от нагревания. Вторые имеют стрелочку, связанную с пластиной, которая изгибается от нагревания или охлаждения. На третьих просто показаны цифры, а что внутри, на первый взгляд непонятно.

Термометры расширения вполне можно использовать для снятия показаний температуры и передачи ее в ПД. Существуют ртутные термометры, в которых встроен подвижной контакт, и при достижении температурой определенного значения происходит его замыкание. Это можно использовать для дальнейшей обработки. Есть термометры, обладающие многими контактами; они дороги и редки. К сожалению, это мало что дает для систем ПД. Судить можно только о том, ниже или выше определенного показания в данный момент температура в комнате.

Aнероиды (в переводе с греческого «без жидкости») с контактами я не встречал. Это не значит, что их нет, но выходит, что для ПД это не совсем пригодно.

Цифровые термометры — это как раз то, что нам надо. И использовать мы будем, конечно, не весь термометр, а только его датчик. В цифровых термометрах могут быть разнообразные датчики; нам нужен один из наиболее используемых: DS18B20.

DS18B20

Все (в том числе и я) называют DS18B20 цифровым датчиком температуры. Однако это не просто датчик: он сам — программируемый цифровой термометр. Он измеряет температуру в диапазоне от -55 до +125 градусов Цельсия, имеет программируемое температурное разрешение от 9 до 12 бит и позволяет задавать верхний и нижний температурные пороги, в случае превышения которых устанавливается флаг аварии.

Каждый термометр DS18B20 имеет уникальный 64 битный серийный номер, который используется для его адресации на шине 1-Wire. Это позволяет объединять на одной шине несколько независимо работающих термометров и осуществлять между ними и контроллером обмен данными по протоколу 1-Wire.

Также особенностью данного термометра является то, что его можно запитывать не только от источника питания, но и от сигнального провода. Это так называемый режим паразитного питания. В этом режиме для подключения DS18B20 требуется всего два провода — сигнальный и возвратный (земляной, GND).

Мы не будем пользоваться всеми функциями этого замечательного прибора: для экономии времени, журнального места и доступности всё будет сжато и упрощенно. Но даже это вполне может привести в восторг неискушенного пользователя Linux.

О протоколе 1-Wire и подключении

Эти слова всё равно сказать надо, поскольку необходимо иметь хотя бы общее представление о том оборудовании, которое мы будем использовать.

Для обмена данными термометр DS18B20 использует протокол 1-Wire (однопроводной протокол). Это низкоскоростной двунаправленный полудуплексный последовательный протокол обмена данными, использующий всего один сигнальный провод. Естественно, требуется еще и возвратный (земляной) провод, но об этом маркетологи обычно умалчивают. Протокол 1-Wire был разработан фирмой Dallas Semiconductor в конце 1990-х годов (опять золотые 1990-е, замечаете?).

Шина 1-Wire должна быть обязательно подтянута к плюсу питания через резистор номиналом 4,7 кОм (допускается варьировать номинал; чем длиннее провод, тем меньше номинал, вблизи допустимо 10 кОм). Напряжение источника питания — от 3 до 5 вольт.

Имеется несколько типов сигналов, определенных протоколом 1-Wire — импульс сброса, импульс присутствия, запись 0, запись 1, чтение 0 и чтение 1. Все эти сигналы, за исключением импульса присутствия, формируются на шине главным устройством — MASTER'ом. В нашем случае это микрокомпьютер Banana Pi.

Принцип формирования сигналов во всех случаях одинаковый. В начальном состоянии шина 1-Wire с помощью резистора подтянута к плюсу питания. Главное устройство «проваливает» на определенное время шину 1-Wire в ноль, затем «отпускает» ее и, если нужно, «слушает» ответ подчиненного (SLAVE) устройства. В нашем случае подчиненное устройство — термометр DS18B20.

Физически это реализуется так:

» **Операция записи бита** Вывод микроконтроллера устанавливается в режим выхода, и на нем устанавливается логический ноль. Выдерживается пауза, длительность которой зависит от значения передаваемого бита (0 или 1), затем вывод переводится в режим входа в состоянии Hi-z и снова выдерживается пауза.

» **Операция чтения бита** Вывод микроконтроллера устанавливается в режим выхода и на нем устанавливается логический ноль. Выдерживается определенная пауза, вывод переводится в режим входа в состоянии Hi-z, выдерживается пауза, а затем микроконтроллер считывает потенциал вывода.

Даже краткое описание процедур обмена данными займет 1-2 страницы, но об одной особенности сказать надо обязательно, она важна для понимания некоторых обстоятельств работы с этими датчиками.

Для обмена данными с устройствами надо знать их адреса на шине («Имя, сестра, имя?!»). Любое устройство 1-Wire имеет встроенный 64-разрядный код. Этот код хранится в ПЗУ (оно же ROM) и используется для адресации устройства на шине.

На словах эта процедура выглядит следующим образом.

Выполнив инициализацию, MASTER посылает команду `search rom`, и начинается цикл чтения 64-разрядного кода. Микрокомпьютер формирует на шине 1-Wire два тайм-слота чтения. В первый тайм-слот все устройства 1-Wire, подключенные к шине, выдают первый бит своего 64-разрядного кода. Во второй тайм-слот —

инвертированное значение первого бита. Если у всех устройств 1-Wire первый бит адреса единица, то микроконтроллер примет сначала единицу, а затем ноль. Если хотя бы у одного устройства 1-Wire первый бит адреса нулевой, то MASTER в обоих случаях примет ноль. Если активных устройств на шине нет, он в обоих случаях примет единицу. В ситуации, когда MASTER принимает два нуля, возникает неоднозначность: как узнать, у каких устройств переданный бит адреса ноль, а у каких — единица? Протокол решает эту проблему просто: после двух тайм-слотов чтения MASTER должен ответить устройствам 1-Wire, с какими из них он продолжит работу. Для этого он выставляет на шине соответствующий бит — ноль или единицу. Устройства, у которых переданный бит соответствует выставленному, продолжают работу; остальные замолчат (станут неактивными) до следующего сигнала сброса. Далее процедура повторяется еще 63 раза: формирование первого тайм-слота чтения, чтение состояния шины, формирование второго тайм-слота чтения, чтение состояния шины, ответ подчиненным устройствам. По завершении цикла чтения 64-разрядного кода контроллер будет знать адрес только одного устройства 1-Wire. Для получения следующего адреса надо снова запустить цикл чтения, но на сей раз в случае неоднозначности выставить другой бит.

Осилили абзац? Если нет, ничего страшного. Суть в том, что процесс получения адреса очень замедляет работу на шине.

Также замедлит работу процесс преобразования температуры; в режиме максимальной точности он займет аж 1,2 секунды.

Исходя из этого, важно помнить, что процесс получения температуры не мгновенный.

Итак, у вас есть общее представление о том, как происходит обмен данными по шине 1-Wire, и можно перейти к следующей части.

Подключение

Пожалуй, тут просто поверьте на слово; схема не единственная, но из трех проводов два подключаются однозначно: на питание и общий провод.

Я поставил резистор на 10 кОм, но у меня расстояние от датчика до контроллера 15 см; при 40 см понадобится 4,7 кОм, а от 80 см можно уже поставить 1 кОм.

Приятная новость: с железом всё, остальное только софт. Начали?

Прежде всего надо включить драйвер. У Banana Pi он уже включен в сборку на уровне ядра. Для включения будем редактировать `script.bin` — главный конфигурационный файл ядра, использующийся при загрузке. Это скомпилированный двоичный файл, внесение правок в который невозможно. Файл создается при помощи утилиты `flex2bin` из набора `sunxi-tools` из текстового файла `имя_файла.fex`.

Первое, что нужно сделать — клонировать репозиторий с утилитами. Для этого у вас должен быть установлен `Git`; если его нет — установите его. И сразу соберем утилиты:

```
# apt-get update && apt-get install git
# git clone git://github.com/linux-sunxi/sunxi-tools.git
# cd sunxi-tools
# make
```

Переходим в директорию конфигурационных файлов, получаем текстовый файл из двоичного и редактируем его:

```
# cd /boot/bananapi/
# /home/bananapi/sunxi-tools/bin2flex script.bin bananapi.fex
# nano bananapi.fex
```

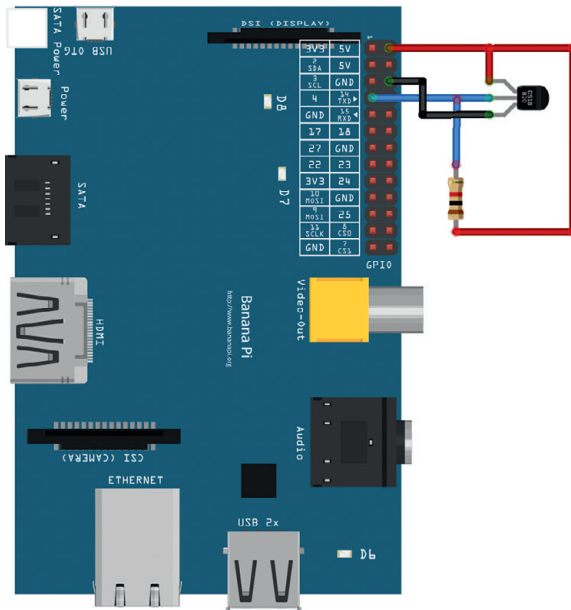
Ищем секцию для шины 1-Wire — ее, скорее всего, нет, поэтому смело добавляем в конец файла:



» Рис. 1. Датчик DS18B20.

» Подпишитесь на печатную или электронную версии на www.linuxformat.ru/subscribe!

➤ Рис. 2. Схема подключения DS18B20.



```
[w1_para]
gpio = 4
```

Вернем загрузочный файл из текстового и перезагрузим систему:

```
# /home/bananapi/sunxi-tools/fex2bin bananapi.fex script.bin
# reboot
```

После перезагрузки посмотрите, что у вас появилось в **/sys/bus/w1/devices**:

```
ls /sys/bus/w1/devices
28-01162350cfee w1_bus_master1
```

Нам нужна директория, начинающиеся с цифр 28 и чего-то там еще (это зависит от серийного номера датчика). Внутри содержится файл **w1_slave**, это и есть ссылка на блочное устройство, коим является наш датчик. Попробуйте его просмотреть — вот что получилось у меня:

```
/sys/bus/w1/devices/28-01162350cfee$ nano w1_slave
4f 01 4b 46 7f ff 0c 10 46 : crc=46 YES
4f 01 4b 46 7f ff 0c 10 46 t=20937
```

Последняя группа знаков и есть искомая температура в градусах по Цельсию, умноженная на 1000, т.е. то, что вы видите, это 20,937°C.

Дальнейшее — дело вашего вкуса и ваших представлений об удобстве. Я ленив, и заходить каждый раз по SSH и смотреть, сколько градусов, не хочу. Поэтому я набросал схему для визуализации (буду рад, если вам она понравится).

Визуализация температуры

Прежде всего, надо выдернуть из файла-блочного устройства саму температуру и привести ее в нормальный вид. И заодно записывать ее раз в 20 минут в файл-накопитель.

Скрипт, в котором делается всё одной командой, кроме переменной DATE, ее я выделил для наглядности (а то уж совсем непонятно будет) — `$ nano save_temp.sh`:

```
#!/bin/bash
DATE=`date +%d:%m:%y-%T`
cat /sys/bus/w1/devices/28-01162350cfee/w1_slave |grep t= | cut -d " " -f 10 | sed -e "s/t=/$DATE/" -e 's/^\(.\{20\}\)/\1./' >> /home/bananapi/temp.txt
exit 0
```

Не забудьте сделать этот файл исполняемым, и попробуйте запустить его в консоли. В файле **/home/bananapi/temp.txt** должна появиться строка, похожая на нижеследующую:

```
30:09:17-05:40:01 20.250
```

Это, значит, 30 сентября 2017 г. температура в комнате была 20,250 градусов.

Теперь расписание записи. Давайте сейчас использовать `crontab`:

```
$ crontab -e
Добавим в самый конец файла строку, аналогичную моей:
*/20 * * * * /home/bananapi/save_temp.sh
```

Температуру пишем каждые 20 минут — периодичность в самый раз: всё же в нормальной ситуации температура в комнате слишком резко меняться не может, а файл от слишком частых замеров будет сильно «распухшим».

Тут стоит немного отвлечься и подумать о том, что наша система вполне может быть использована для пожарной сигнализации — в самом деле, если написать скрипт, раз в минуту мониторящий температуру и при резком (на 3–5 градусов за минуту) скачке оповещающий вас звонком через *Asterisk*. Конечно, к датчику может подойти собака и подышать на него, но это маловероятно. А в остальных случаях это показатель беспорядка (уж проверить нужно обязательно).

Но вернемся к учебному проекту, его не будем загромождать упомянутым скриптом.

Итак, файл-накопитель имеется. Он состоит из двух колонок, разделенных пробелом. В первой колонке дата-время, во второй температура. Я буду выводить информацию в web-интерфейс, а значит, читать файл буду при помощи PHP.

Создаем файл, который будет это делать (`/home/bananapi$ nano /var/www/array_include.php`):

```
<?php
$string=file_get_contents('/home/bananapi/temp.txt');
$array=preg_split("[\s]",$string);
$index=0;
$array_odd=[];
$array_even=[];
foreach($array1 as $key => $value) {
    if ($index & 1) {
        if ($value<=0) $value=(($array1[$key-2]+$array1[$key+2])/2);
        array_push($array_odd,$value);
    }
    else array_push($array_even,$value);
    $index++;
}
$temp = str_replace(""," ",json_encode($array_odd));
$time = json_encode($array_even);
$temp_middle = array_sum($array_odd)/count($array_odd);
?>
```

Кратенько о том, что мы делаем:

- 1 Читаем файл-накопитель в строковую переменную.
- 2 Переносим в массив, разделяя по пробелу.
- 3 Создаем еще 2 массива: в них будут запоминаться дата-время и температура.
- 4 Перебираем массив, переносим все значения с нечетным индексом (дата-время) в первый новый массив, с четным — во второй (температура). Заодно вычеркиваем все отрицательные значения (откуда они берутся, скажу чуть ниже) и заменяем их интерполяцией предыдущего и последующего показаний.
- 5 В переменную `temp` вносим температуру, стирая попутно знаки кавычек (так надо).
- 6 В переменную `time` переносим дату-время.
- 7 Вычисляем среднее значение температуры за весь период.

Откуда отрицательные значения? Вспоминаем, что процесс преобразования температуры не мгновенный, и если мы попадаем процессом чтения на процесс преобразования, то получаем отрицательное значение. Благо, что это бывает редко, 1-2 раза в сутки. Поэтому я заменяю его средним арифметическим «заднего» и «переднего» показаний.

► Рис. 3. График температуры, построенный с помощью *highcharts.js*.



Красота на web-панели

Вы, вероятно, помните, как в одной из статей я невольно отзывался о JavaScript. К сожалению, я всё еще не могу отключить в своем браузере плагин, блокирующий скрипты. Виною тому бесстыдные «web-дизайнеры», которые лепят выпрыгивающие со всех сторон окна, призывающие купить, поехать в путешествие, обучиться, увеличить или уменьшить себе какую-нибудь часть тела...

Но обойтись без JavaScript уже невозможно. Да и неудобно без них. Вот как, например, проверить валидность введенных данных в формы? Только передать на сервер, там проверить при помощи PHP и вернуть обратно. И без принудительной перезагрузки не обойдешься...

Мы с вами уже использовали технологию Ajax для индикации состояния наших нагрузок. Поэтому применение JavaScript для вывода графика не будем считать отступлением от религиозных убеждений. И кроме того, будем без зазрения совести пользоваться результатами чужой интеллектуальной деятельности.

Итак, вуаля — <https://www.highcharts.com>, или просто чарты. Превосходно, что для некоммерческого использования эта библиотека совершенно бесплатна. Описание и способ применения есть на сайте, да и в Интернете тоже много чего можно прочитать. Поэтому просто содержимое файла, и немного комментариев (`nano /var/www/graf.php`):

```
<html>
<head>
<title>График</title>
<meta charset="utf-8">
<script type="text/javascript" src="http://code.highcharts.com/"></script>
</head>
<body>
<?php
require_once('array_include.php');
?>
<div id="container" style="width:50%; height:400px;
margin-left;"></div>
<script>
var time = <?php echo $time?>;
var temp = <?php echo $temp?>;
var temp_middle = <?php echo $temp_middle?>;
$(function () {
var myChart = Highcharts.chart('container', {
chart: {
type: 'spline'
},
title: {
text: 'Изменение температуры в комнате'
},
subtitle: {
```

```
text: 'Измерено датчиком DS18B20, среднее значение
'+temp_middle
},
xAxis: {
categories: time
},
yAxis: {
title: {
text: 'Температура (°C)'
}
},
plotOptions: {
line: {
dataLabels: {
enabled: true
},
enableMouseTracking: false
}
},
series: [{
name: 'Период замера один раз в 20 минут',
color: 'red',
data: temp
}]
});
</script>
</body>
</html>
```

Я старался писать только самое необходимое.

Итак, вначале подключаем библиотеку *highcharts.js* прямо с сайта. Если хотите, можно скачать ее и подгружать с локального диска. Затем подключаем тот файл PHP, который делает из файла-накопителя 2 массива с данными и перерабатывает их в строковые переменные. Далее мы передаем переменные из PHP в JavaScript для использования внутри библиотеки. Потом создаем объект (слой), в который мы будем записывать данные; советуем вам поэкспериментировать с параметрами. И просто передаем управление собственно библиотеке — остальное она сделает сама.

И у меня получилось то, что показано на рис. 3 — и я убежден, что так же получится и у вас.

Как видите, курсором мыши можно получить подробную информацию о конкретной точке. Также можно добавить второй, третий ряды данных, изменить вид графика, его цвета, вид точек. Это достаточно интересно, и вполне добротно на вид.

Встроить в главный файл вашего послушного дома данный график вам придется самим, а я с вами прощаюсь.

До следующего выпуска, удачи вам в программировании Послушного Дома! **LXF**

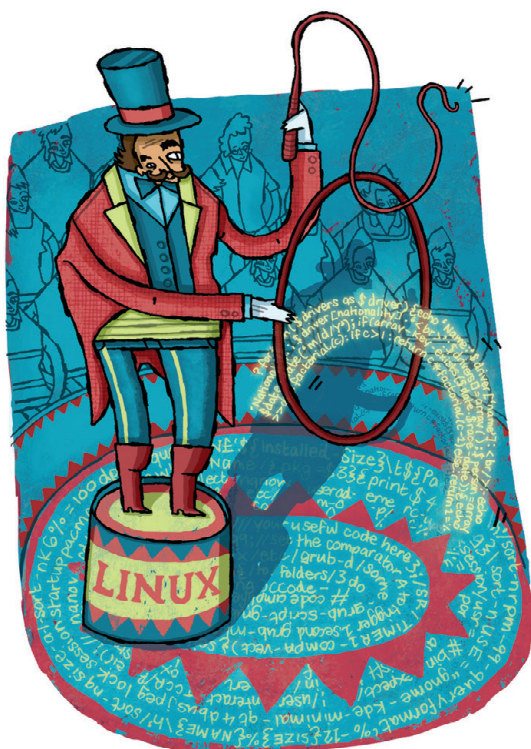
Загадка LXF: Сортировка

Михалис Цукалос научит вас различным способам сортировки данных, когда объем памяти ограничен.



Наш
эксперт

Михалис Цукалос — администратор баз данных, администратор UNIX, программист и математик. С ним можно связаться через сайт www.mtsoukalos.gr и в Twitter: @mactsouk.



```
11 func findNext(filename string) (string, error) {
12     fin, err := os.Open(filename)
13     if err != nil {
14         return "", err
15     }
16     r := bufio.NewReader(fin)
17
18     minimum, err := r.ReadString('\n')
19     if err == io.EOF {
20         return "", err
21     } else if err != nil {
22         return "", err
23     }
24
25     temp := filename + ".temp"
26     fout, err := os.OpenFile(temp, os.O_RDWR|os.O_CREATE, 0660)
27     if err != nil {
28         return "", err
29     }
30
31     for {
32         line, err := r.ReadString('\n')
33         if err == io.EOF {
34             break
35         } else if err != nil {
36             return "", err
37         }
38
39         if line < minimum {
40             temp := minimum
41             minimum = line
42             line = temp
43         }
44         fmt.Fprintf(fout, "%s", line)
45     }
46 }
```

► Рис. 1. Код Go для функции findNext(), которая находит минимальную строку в текстовом файле после сканирования всего файла.

Примите участие!

Посоревнуйтесь с Михалисом! Отправьте свой код на адрес электронной почты **LXF: linuxformat@futurenet.com** с темой: **Загадка LXF228**. Убедитесь, что ваш код компилируется без ошибок (также не забудьте хорошо задокументировать код и снабдить его подробными комментариями), и вы получите шанс обойти нашего умника.

Основная тема этого урока — поиск способов сортировки данных при ограниченном объеме оперативной памяти. Однако вы также узнаете, как читать текстовые файлы по строкам в C, Go и Python 3, а также как сортировать числа и строки — это не так уж мало для столь небольшого учебника! Еще раз, если вы хотите извлечь максимальную пользу из этого или предыдущих учебников *Linux Format*, попробуйте сначала поразмышлять о проблеме самостоятельно, придумать некоторые возможные решения и попытаться их реализовать — ведь это практическое руководство!

Для простоты на нашем уроке предполагается, что каждое входное значение представляет собой целую строку.

Используем минимум памяти

Единственный способ минимизировать потребление памяти — всё время сохранять данные в файле и читать их из файла или печатать результат на экране. Однако не считая некоторых переменных, необходимых для хранения минимального значения и содержимого текущей строки, это приведет к тому, что вам придется читать входной файл несколько раз. В следующих двух разделах будут представлены реализации этой методики на языках Go и C.

Применение Go

Программа на Go будет называться **multipleScans.go**. Как следует из названия, она сканирует входной файл несколько раз, чтобы тратить как можно меньше памяти. Чтобы всё было интереснее, предположим, что нам надо сортировать строки, а не числовые значения. В файле **multipleScans.go** основным является следующий код:

```
if line < minimum {
    temp := minimum
    minimum = line
    line = temp
}
fmt.Fprintf(fout, "%s", line)
```

Предыдущий код Go сравнивает текущее минимальное строковое значение с текущей строкой. Если текущее минимальное значение меньше текущей строки, программа записывает текущую строку в выходной файл и переходит к следующей строке или помещает текущую минимальную строку в файл и заменяет минимальное значение текущей строкой.

Таким образом, при каждом сканировании входного файла вы отыскиваете в нем минимальное значение, добавляете это

О конкурсе

Задача для программирования в этом выпуске *Linux Format* заключается в сортировке данных при ограниченном объеме оперативной памяти. Как вы понимаете, это довольно сложно себе представить, потому что мы уже привыкли к большому ОЗУ на своих компьютерах и не беспокоимся об этом.

Наряду с ограничением памяти мы добавим к программе другие ограничения, чтобы сделать задачу интереснее, а код — проще.

Однако не забывайте, что у простоты есть своя цена: это дополнительное время, которое вы тратите на размышления о задаче и ее ограничениях.

Можете ли вы предложить метод сортировки данных при ограниченном объеме памяти, который окажется лучше нашего? Сообщите нам, и, возможно, вы найдете свой код в ближайшем номере *Linux Format*!

значение в новый файл и удаляете его из входного файла. При таком способе вам нужны следующие переменные: первая переменная для хранения текущего минимального значения, вторая для хранения содержимого строки, считываемой из входного файла, а также еще несколько переменных для доступа к используемым файлам.

Однако, как вы можете догадаться, если у вас хорошо с математикой, если во входном файле 1000 строк, вам придется выполнить $1+2+3+\dots+999+1000$ операций считывания строк, а это 500500! Подсчитайте-ка, сколько операций чтения потребуется для файла в миллион строк!

Если вы не знаете, как вычислить это значение, вам поможет вот эта небольшая программа на Python 3 **sumTo.py**:

```
#!/usr/bin/env python3
import sys
if len(sys.argv) >= 2:
    upperLimit = int(sys.argv[1]) + 1
else:
    print("Please provide an integer!")
    exit(1)
sum = 0
for i in range(upperLimit):
    sum += i
print(sum)
```

Единственный параметр, который принимает программа **sumTo.py** — верхний предел. Запустив файл **sumTo.py**, мы получим следующий результат:

```
$ ./sumTo.py 100000
5000050000
$ ./sumTo.py 1000000
500000500000
```

Вернемся к исходной проблеме. Как вы видите, при запуске файла **multipleScans.go** с файлом **sortMe** в качестве параметра мы получаем следующие результаты:

```
$ cat sortMe
2 1 10 3 a r 2 c
$ go run multipleScans.go sortMe
EOF
$ cat sortMe
1 10 2 2 3 a c r
```

Учтите, что в этом выводе мы заменили переводы строк на пробелы, чтобы сэкономить место. На рис. 1 показан код функции **findNext()** языка Go, который можно найти в файле **multipleScans.go**. Эта функция возвращает минимальную строку, найденную во входном файле.

Применение C

Если вы хотите иметь полный контроль над тем, что делаете, скорее всего вы выберете язык C, и на нашем уроке мы представим код программы на C.

Программа C называется **sortStr.c** и будет сортировать строки, содержащие целые числа. В ней используется тот же алгоритм, что и в **multipleScans.go** — единственные два отличия заключаются в том, что приходится иметь дело с целыми значениями вместо строк и что результат запуска программы **sortStr.c** выводится на экран, а не сохраняется в текстовый файл.

В файле **sortStr.c** наиболее важен следующий код:

```
if (first) {
    first = 0;
    minimum = myInt;
    continue;
}
if (myInt < minimum) {
    int temp = minimum;
    minimum = myInt;
    myInt = temp;
}
fprintf(fout, "%d\n", myInt);
```

Если это самая первая строка входного файла, то в переменную **minimum** записывается прочитанная строка, и вы переходите к следующей итерации цикла **while()**. В противном случае вы проверяете, меньше ли текущее значение (**myInt**) переменной **minimum**. При запуске файла **sortStr.c** мы получим следующий результат:

```
$ cat test
12 1 1 1 1 1 2 2 3 3 3
```

Скорая помощь

Хотя C — очень мощный язык программирования, Go — хорошая альтернатива, способная улучшить качество кода и повысить его безопасность. Если у вас нет веских причин применить C, рассмотрите такие языки программирования, как Go и Rust.

```
18 int findNext(char *filename )
19 {
20     FILE* fin;
21     char buf[BUFFER];
22     int minimum, myInt;
23     first = 1;
24
25     struct stat st;
26     stat(filename, &st);
27     int size = st.st_size;
28     if ( size == 0 ){
29         stop = 1;
30         return -1;
31     }
32
33     if ((fin = fopen(filename, "r")) == NULL) {
34         stop = 1;
35         return -1;
36     }
37
38     char temp[120];
39     strcpy(temp, filename);
40     strcat(temp, ".temp");
41     FILE *fout = fopen(temp, "w");
42     if (fout == NULL) {
43         stop = 1;
44         return -1;
45     }
46
47     while (fgets(buf, sizeof(buf), fin) != NULL)
48     {
49         if (buf[strlen(buf) - 1] == '\n')
50             buf[strlen(buf) - 1] = '\0';
51
52         if (buf[0] == '\0')
53             continue;
54
55         if (buf[0] == ' ')
56             continue;
57
58         if (isEmpty(buf))
59             continue;
60
61         if (isspace(buf[0]))
62             continue;
63
64         myInt = atoi(buf);
65         if (first)
66         {
67             first = 0;
68             minimum = myInt;
69             continue;
70         }
71
72         if (myInt < minimum)
73         {
74             int temp = minimum;
75             minimum = myInt;
76             myInt = temp;
77         }
78         fprintf(fout, "%d\n", myInt);
79     }
80
81     fclose(fin);
82     fclose(fout);
83     int ret = remove(filename);
84     ret = rename(temp, filename);
85     return minimum;
86 }
```

► Рис. 2. Показан код функции **findNext()** на языке C. Этот код более объемистый, чем версия на Go (рис. 1). Вот цена, которую вы платите за написание кода на C.

» Подпишитесь на печатную или электронную версии на www.linuxformat.ru/subscribe!

```

$ ./sortInt test
1 1 1 1 2 2 3 3 3 12

```

Обратите внимание, что и в этом выводе мы заменили переводы строк на пробелы, ради экономии журнальной площади.

Теперь рассмотрим следующий особый случай:

```

$ cat test
$ ./sortInt test
0

```

Если входной файл содержит символы перевода строки и пробелы, то **sortStr.c** выведет значение 0, поэтому всегда старайтесь определить особые случаи и обязательно их документируйте. Правда, проблема с программой **sortStr.c** заключается в том, что она разрушает входной файл. Это можно исправить, реализовав функцию **C**, которая создает копию исходного входного файла.

На рис. 2 (на стр. 75) показан код **C** функции **findNext()** из программы **sortStr.c**. Чтобы уменьшить код **sortStr.c**, проверка ошибок выполняется только базовая.



Как вы можете понять из задач и кода в этом учебнике, дорога важнее, чем пункт прибытия. При решении таких задач важно попробовать как можно больше вещей на различных языках программирования!

При минимуме памяти

Хотя возможность обойтись вообще без памяти, кроме памяти для хранения переменных, может показаться очень неплохой, а иногда и соблазнительной, на самом деле это очень медленно. В большинстве случаев вы сможете задействовать некоторое пространство памяти — предположим, что пока будем выделять доступную память с помощью массива.

Чтобы сделать нашу задачу еще увлекательнее, попробуем решить другой вариант: будем обрабатывать входные файлы с положительными целыми числами без дубликатов.

```

2. mtsouk@mail: ~/doc
code$ go version
go version go1.3.3 linux/amd64
code$ cat test
3
4
10
9
1
0
30
12
5
code$ go run sortArrays.go 30 test
0 30
Passes: 2
0
1
3
4
5
9

```

► Рис. 3. На этом рисунке показан запуск утилиты **sortArrays.go**, которая умеет сортировать натуральные числа, используя заданное пространство памяти.

Для решения сначала найдем диапазон положительных целых чисел, содержащихся во входном файле, и разделим диапазон на поддиапазоны в соответствии с объемом доступной памяти. При каждом сканировании входного файла элементы массива будут соответствовать элементам одного из поддиапазонов.

После сканирования поддиапазона элементы массива, найденные во входном файле, выводятся на экран, и процесс перезапускается. Так будет продолжаться до тех пор, пока все поддиапазоны не будут обработаны и весь диапазон чисел не будет проверен. Кроме того, вам не придется удалять из входного файла элементы, потому что при каждом просмотре файла вы будете знать, что ищите.

Итак, в следующих двух разделах будут представлены две утилиты командной строки, написанные на языках Go и Python 3. Эти программы сортируют положительные целые числа с использованием заданного объема памяти, но проблема в том, что они не работают со строками.

Словари Python 3

Скрипт будет называться **sortArtrays.py** и будет принимать два аргумента командной строки: объем доступной памяти и имя текстового файла. Входной файл остается нетронутым.

Здесь вас ждет сюрприз: в реализации Python 3 мы вместо массива воспользуемся словарем, так как словари в Python 3 более гибкие. Кроме того, пользуясь словарем, вы не занимаете лишнего объема памяти.

В файле **sortArtrays.py** наиболее важен следующий код:

```

if (value >= space) and (value < space+size):
    key = value
    data[key] = 1

```

Если значение находится в желаемом диапазоне, то в переменную словаря **data** вставляется новая пара, что и делает требуемую работу. При запуске файла **sortArtrays.py** мы получим следующий результат:

```

$ ./sortArtrays.py 2 test
2 8
Прогонов: 4
2 3 4 5 8

```

В следующем разделе этот же алгоритм будет реализован на Go.

Массивы Go

Теперь реализуем сценарий с фиксированным объемом оперативной памяти на языке Go. Утилита на Go будет называться **sortArrays.go** и будет принимать два параметра: объем доступной памяти и имя текстового файла для обработки. Этот файл должен содержать только положительные целые числа, также называемые натуральными числами.

Первая идея — использовать массив Go. Поскольку массивы Go имеют фиксированный размер, они идеально подходят для имитации фиксированного объема ОЗУ. Недостаток этой идеи в том, что, передавая функции массив, вы фактически передаете копию массива, а это означает, что любые изменения в массиве по завершении функции будут утеряны. Но раз **sortArrays.go** выводит данные на экран, это не проблема. Настоящая проблема с массивами Go заключается в том, что их размер нелегко определить во время выполнения. Поэтому в **sortArrays.go** мы будем использовать срезы [slice].

Поскольку каждый элемент среза будет использоваться для заданного числа, снова записывать это число не надо.

Кроме того, так как начальное значение всех элементов среза будет равно 0, то каждый раз, когда вы найдете число, относящееся

► Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

```

1  #!/usr/bin/perl -w
2
3  use strict;
4  use warnings;
5
6  my ($in, $out) = @ARGV;
7
8  if ( (not defined $in) || (not defined $out) ) {
9      die "Need two argument!\n";
10 }
11
12 open(INFILE,'<', $in) or die "$!";
13 open(OUTFILE,'>', $out) or die "* $!";
14 print OUTFILE sort <INFILE>;
15 close OUTFILE;
16 close INFILE;
17

```

► Рис. 4. Показан код Perl-скрипта `sortRAM.pl`, который сортирует входной файл без каких-либо ограничений оперативной памяти.

к текущему поддиапазону, значение соответствующего элемента среза станет равным 1, чтобы обозначить попадание.

В файле `sortArrays.go` для нас важнее всего следующий фрагмент кода:

```

if valueToExamine >= space && valueToExamine < space+size {
    index := valueToExamine - space
    data[index] = 1
}

```

Этот код проверяет, находится ли натуральное число внутри желаемого поддиапазона. Если да, значение в срезе данных изменяется на 1.

Например, если у вас во входном файле присутствует число 305 и вы используете 100 ячеек памяти, а минимальное число в текстовом файле равно 0, то число 305 будет обрабатываться в четвертом гоне и будет обозначаться шестым элементом среза, с индексом 5. Затем вы изменяете значение шестого элемента на 1.

На рис. 3 показаны различные варианты выполнения программы `sortArrays.go`.

Perl для скорости

Perl особенно известен тем, что может решать сложные задачи при посредстве небольшого объема кода. Поэтому в этом разделе мы устраним ограничения на использование памяти и воспользуемся Perl для сортировки текстового файла, чтобы лучше понять преимущества наличия большого количества памяти. Имя скрипта Perl — `sortRAM.pl`, его код можно увидеть на рис. 4.

Повторим еще раз: удивительно, как много всего можно понаделать при помощи даже очень небольшого кода на Perl! При выполнении файла `sortRAM.pl` мы не получим никакого видимого результата на экране, но входной файл будет отсортирован по алфавиту и по числам.

Эффективна ли программа `sortRAM.pl`? Если у вас много памяти, то довольно эффективна. Но если вы попытаетесь отсортировать огромный текстовый файл на действительно занятом рабочем сервере, это может поставить под угрозу общую производительность сервера.

В следующем разделе мы подробнее расскажем об эффективности представленных утилит при сортировке относительно большого текстового файла с числовыми данными.

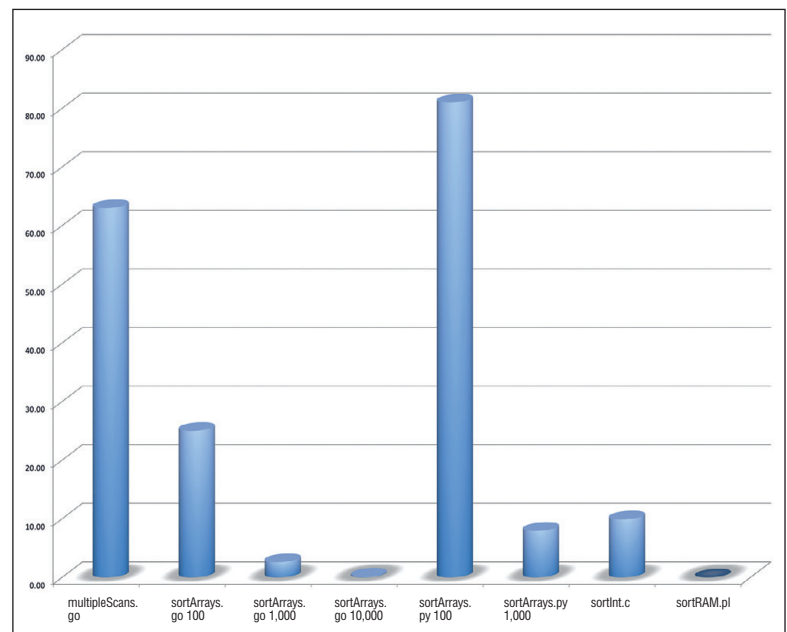
Оценка производительности

Для проверки производительности всех программ, разработанных на этом уроке, мы будем использовать утилиту `time`. Хотя утилита `time` не на 100% точна, она даст очень хорошее представление о том, какая программа выполняется быстро, а какая — очень медленно, и насколько эффективны используемые алгоритмы. Все программы будут обрабатывать текстовый файл со 10 000 случайными натуральными числами.

На рис. 5 (внизу) показан график значений команды `time`, которая использовалась для оценки скорости выполнения программ нашего урока.

Как вы видите, результаты получились более или менее ожидаемые: чем больше оперативной памяти, тем быстрее работает программа. Программа `sortInt.c` оказалась довольно быстрой, но самым быстрым стал скрипт Perl.

Второе место занимает программа на Go `sortArrays.go`, которая использовалась с 10 000 ячеек памяти. **LXF**



► Рис. 5. Производительность программ, разработанных на нашем уроке, на основе поля `real` утилиты `time`.

Добавление ограничений к задаче

Как вы видели из реализации программы `sortArrays.py`, наличие большого количества ограничений или дополнительной информации о данных обычно полезно и помогает упростить написание программы.

Кроме того, код программы `sortArrays.go` намного быстрее и проще, чем код программы `multipleScans.go`, потому что вы знаете, что данные лежат в заданном диапазоне, дубликаты

отсутствуют и у вас больше памяти, чем в программе `multipleScans.go`.

Общий совет здесь состоит в том, что следует всегда стараться больше узнать о данных и использовать эту информацию, чтобы улучшить программу, так как более простая структура программы — это всегда хорошо.

А проводить слишком много времени, пытаясь найти шаблоны и ограничения в данных

и проанализировать данные, не всегда хорошо. Иногда нужно просто написать первую версию своей программы и попытаться ее улучшить, когда вы узнаете больше о том, какие данные вы будете обрабатывать.

Как однажды справедливо выразился классик программирования Дональд Кнут [Donald Knuth]: «Преждевременная оптимизация — это корень всех зол!»



Без сервера: Развернем в AWS

Развертывайте облачные приложения несколькими строками кода и обходясь без серверов. Вам поможет **Дэн Фрост**.



Наш эксперт

Дэн Фрост — технический директор Zev.com, писатель и исследователь идей и технологий. Дэн часто сидит в Twitter @danfrost, где он порицает (а затем восхваляет) новые компьютерные штуки.



Бессерверные вычисления позволяют создавать масштабируемый, готовый к рабочей среде код без необходимости развертывать серверы и управлять ими. Многим из нас от облачных вычислений нужно именно это: больше не думать о серверах, не оплачивать инфраструктуру и не управлять сложными правилами масштабирования.

Бессерверное программирование можно использовать для разработки серверной части для приложений, сайтов, виртуальной реальности, конвейеров данных и др. На самом деле, почти всё, что можно разработать на сервере, выполнимо и с помощью бессерверной архитектуры.

На нашем уроке мы рассмотрим, как применить бессерверное программирование для создания базового конвейера данных, который проверяет искомые или подозрительные строки в журналах и выводит полученные результаты в базу данных NoSQL — DynamoDB.

Понятие конвейера (набор функций, связанных друг с другом с помощью событий) для бессерверного программирования имеет основополагающее значение. Этот шаблон можно использовать для преобразования изображений, создания видео, майнинга данных и много чего еще.

В нашем примере мы предоставим API для отправки файлов и вторую функцию, которая сканирует файлы на ключевые слова. Эти функции пригодны для анализа файлов журналов, поиска твитов, содержащих определенные слова, или поиска организаций, упомянутых в новостях. Код для поиска слов очень прост, но давайте сосредоточимся на бессерверной архитектуре.

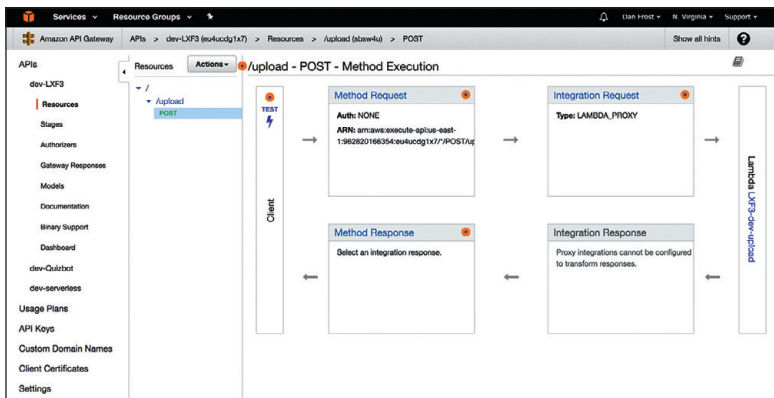
Я расскажу вам обо всем, что необходимо сделать для установки. Вы, кроме того, можете раздобыть подробную информацию о фреймворке на сайте <https://serverless.com/framework/docs/>. Весь код нашего урока доступен на <https://github.com/danfrost/serverless-programming-tutorial>.

Фреймворк

Развитие вычислительных технологий идет от большого к малому; от огромных компьютеров до крошечных смартфонов. А развитие программирования движется от монолитных серверов, на которых выполняются огромные программы, к очень маленьким специализированным микросервисам, которые делают свою работу очень хорошо.

Но когда вы создаете микросервис, который, например, изменяет размер изображения и помещает на него ваш логотип для использования в социальных сетях, или сканирует пакеты на наличие подозрительных данных, этот микросервис всё равно должен где-то работать. Обычно вы запускаете сервер, устанавливаете на него программное обеспечение, управляете резервными копиями и контролируете работоспособность сервера, чтобы он был доступен всё время.

Некоторым системам без серверов никак не обойтись — сайты на Wordpress или сайты электронных магазинов предполагают работу на сервере. Но бывает и такой код, для которого целый сервер не требуется. Я развернул десятки сценариев, которые запускаются один раз в день, чтобы объединить несколько систем вместе, отслеживать трафик или собирать статистику. Для них не нужен сервер, и они часто развертываются на случайных серверах, но, похоже, это не совсем подходящее место.



➤ С AWS могут возникнуть сложности при настройке, но бессерверные вычисления вам помогут. Здесь мы настроили шлюз API всего несколькими строками кода.

Serverless и бинарники

В нашем примере используются уже установленные библиотеки Python (*boto3* и *logging*), но если вы обрабатываете изображения, занимаетесь машинным обучением или решаете другие задачи, которые связаны с дополнительными пакетами или двоичными файлами, вам придется установить их самостоятельно. А если у вас нет доступа к серверу?

В этом случае сохраните библиотеки в корневом каталоге проекта. Итак, например, если вы хотите использовать библиотеку *numpy*, добавьте ее в основную папку проекта Serverless. Я обнаружил, что с помощью создания символических ссылок

на внешние библиотеки гораздо удобнее управлять большим количеством зависимостей. Чтобы включить библиотеки, ознакомьтесь с дополнительными сведениями о параметре 'package' в файле **serverless.yml**. Я часто проверяю все загружаемые файлы с помощью команды `unzip -l .serverless/*.zip`, просто чтобы убедиться, что я всё делаю правильно.

С бинарниками всё сложнее, но принцип тот же: их надо скомпилировать для Amazon Linux и включить в каталог проекта Serverless. Если вы не хотите запускать Amazon Linux, можете либо запустить

экземпляр EC2, либо использовать Docker или локальную виртуальную машину. Если не скомпилировать файлы для Amazon Linux, вы получите бессмысленные сообщения об ошибках, устранить которые Google не поможет.



Здесь мы и приходим к бессерверным вычислениям. Вместо развертывания сервера вы создаете и развертываете «функции», которые решают узкие и конкретные задачи. Как и команды Unix, они делают что-то одно, и делают это очень хорошо. Эти функции могут быть простенькими и включать всего несколько строк кода на Python — или сложными, и использовать, например, двоичные файлы для анализа изображений. Выбор за вами. Но наилучшей практикой и тенденцией в бессерверных вычислениях является создание отдельных функций для очень узкого и конкретного применения.

Функции запускаются «событиями», способными возникать в нескольких местах. Если вы хотите написать API для своего приложения, как мы собираемся сделать в этом примере, можете привязать функции к API конечных точек. Или, если вы хотите обработать файл, можете предоставить запуск событий хранилищу файлов Amazon, S3. Вы можете даже запускать события из чата Facebook. Это позволяет создавать приложения для чата с минимумом усилий.

Приступая к работе

Для этого урока я собираюсь использовать фреймворк Serverless (serverless.com), который позволяет развертывать приложения в AWS, облаке Google, MS Azure и IBM. Все представленные здесь примеры предназначены для AWS, но вам следует проверить, какая платформа соответствует вашим потребностям. Фреймворк Serverless в настоящее время поддерживает несколько сред выполнения, включая Node, Java Maven, Gradle, Scala и C#. Сообщество очень активно, поэтому следите за новостями о поддержке вашей среды в их канале Slack.

Большое преимущество использования фреймворка Serverless — это время на настройку базовых вещей, которое вы сэкономите. Получить конечную точку API в AWS очень просто, а в Serverless это вообще тривиально.

Для начала создайте учетную запись AWS, если у вас ее еще нет. (Стоимость того, что мы собираемся создать, составляет буквально гроши, но обязательно всё удалите после окончания работы.) Затем установите фреймворк Serverless и создайте новый проект с использованием Python 3 в качестве среды выполнения. Затем посмотрите, что получилось:

```
$ npm install -g serverless
$ serverless create --template aws-python --path
MyServerlessProject
$ cd MyServerlessProject && ls
handler.py
serverless.yml
```

Каталог проекта практически пуст, в нем всего два файла: **handler.py** и **serverless.yml**. **Serverless.yml** — это файл для настройки проекта, а **handler.py** содержит функции, которые будут запускаться. Создайте профиль IAM сами и настройте свои учетные данные:

```
serverless config credentials -p aws -k XXX -s XXXXX --profile
tutorial-profile
```

Теперь измените **serverless.yml** для использования Python 3 и только что настроенного профиля:

```
provider:
  name: aws
  runtime: python3.6
  profile: tutorial-profile
```

Знайте!

За запуск кода на Amazon Web Services может взиматься плата. Мы не несем ответственности за любые расходы, вызванные выполнением команд в данном руководстве. О возможных расходах и сборах см. во врезке «Очень низкие цены» на стр. 79.

»

```
[→ Coding serverless create --template aws-python --path MoveFastAndBreakCode
Serverless: Generating boilerplate...
Serverless: Generating boilerplate in "/Users/dan/Coding/MoveFastAndBreakCode"
```



```
Serverless: Successfully generated boilerplate for template: "aws-python"
```

» Запуск и развертывание нового проекта в настоящем облаке может занять не более минуты.

» Подпишитесь на печатную или электронную версии на www.linuxformat.ru/subscribe!



```
region: us-east-1
Откройте файл handler.py и измените параметр 'message':
"message": "Это крошечная функция...",
Давайте развернем и вызовем функцию:
serverless deploy -v
serverless invoke -f hello
```

Фреймворк Serverless создал конфигурацию CloudFormation, которая используется для создания всех ресурсов, необходимых для работы вашей крошечной функции в облаке AWS. Затем он сворачивает и выгружает ваш код. Это потрясающе, но таким способом мы используем только малую часть возможностей фреймворка. Почему бы нам не попробовать создать нечто более впечатляющее?

Рабочая среда

Основная последовательность действий для проектов Serverless такова: изменяем файлы локально, запускаем локально (если это возможно), развертываем в AWS и запускаем в AWS. Давайте настроим хорошую рабочую среду для такого рабочего процесса. Чтобы получить возможность удобной записи в журналах, добавьте следующие строки в начало файла обработчика:

```
import logging
logger = logging.getLogger()
logger.setLevel(logging.DEBUG)
logger.info("Обработчик стартовал")
```

```
Затем добавьте следующую строку в начале функции hello:
logger.info("Начнем, пожалуй.")
logger.info(event)
```

Потом откройте два окна. Запустите команду `serverless logs -t -f hello` в первом окне и `serverless invoke -f hello` — во втором. При вызове функции вы увидите сообщения в журналах.

Я обнаружил, что открытые журналы и возможность быстро развертывать изменения реально ускоряют работу. По мере усложнения проекта вы можете воспользоваться *pylint* или другими предварительными обработчиками, такими как SASS, пакеты React и др.

Налаживаем конвейер

Для нашего проекта понадобятся две лямбда-функции: одна — для API отправки содержимого в контейнер S3, и вторая — для сканирования интересующих нас файлов и отправки этих файлов в ДупамтоDB. Для начала добавьте следующий код в файл `serverless.yml` в разделы 'provider' и 'functions'.

```
provider:
...
iamRoleStatements:
- Effect: "Allow"
Action:
- "s3:*"
```

```
Resource: "arn:aws:s3:::dropbucketlxf/*"
functions:
...
upload:
handler: handler.upload
events:
- http:
path: upload
method: post
```

В файле `handler.py` добавьте в начало файла `'import boto3'` и добавьте следующий метод в файл:

```
def upload(event, context):
logger.info(event)
jsondata = json.loads(event['body'])
data = jsondata['data']
data_hash = hashlib.sha1(data.encode('utf-8')).hexdigest()
s3conn = boto3.resource('s3')
s3conn.Bucket('dropbucketlxf').put_object(
Key=data_hash + '.json',
Body=json.dumps(data))
body = { "message": "Uploaded file" }
response = {
"statusCode": 200,
"body": "done"
}
return response
```

После этого выполните развертывание, как и раньше. Откройте файлы журналов для метода 'upload' в одном терминале, а в другом — вызывайте свою конечную точку. Чтобы начать запись в журнал, несколько раз вызовите команду `curl`:

```
$ curl -XPOST https://YOUR-ENDPOINT-DOMAIN/dev/upload -d '{"data": "Невинное событие"}'
$ curl -XPOST https://YOUR-ENDPOINT-DOMAIN/dev/upload -d '{"data": "ПЛОХОЕ событие"}'
```

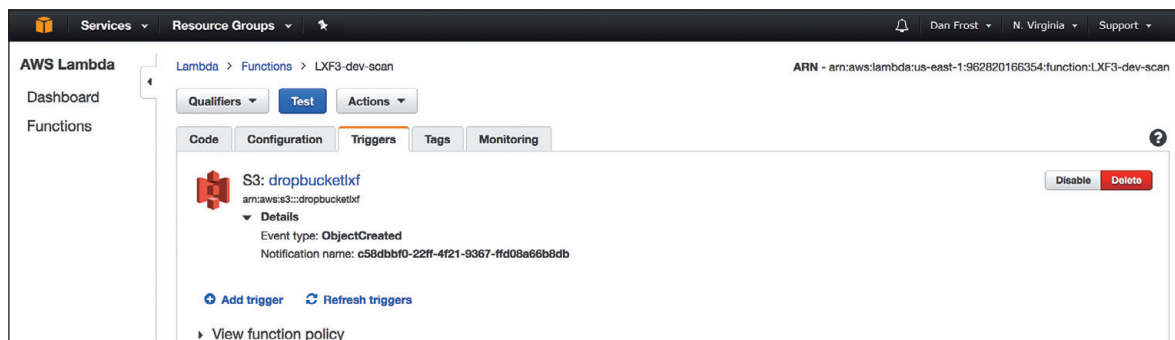
Сделав это, перейдите на консоль S3, чтобы просмотреть свои данные: <https://console.aws.amazon.com/s3/home>. Вы найдете свои данные в файле. Наша цель — обнаружить «ПЛОХОЕ» событие, не запуская сервер.

Создаем события S3

Наш метод «сканирования» будет запускаться создаваемым объектом S3 с использованием магии бессерверных вычислений, управляемых событиями. Чтобы подключить новый метод к событиям S3, добавьте следующий код в раздел 'functions' файла `serverless.yml`:

```
scan:
handler: handler.scan
events:
- s3:
```

➤ В Lambda немало полезного, но нет того, что вам надо. Хотя это решаемая проблема.



➤ Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

```
bucket: dropbucketlxf
event: s3:ObjectCreated:*
```

Этот код лучше всего читать с конца: при создании объектов в контейнере dropbucketlxf вызвать **handler.scan**. Теперь создайте метод для сканирования нужных строк и добавления этих строк в базу данных:

```
def scan(event, context):
    logger.info(event)
    object_key = event['Records'][0]['s3']['object']['key']
    tmpfilenamein = '/tmp/tmpfilein_' + object_key
    logger.info("Получен файл: " + object_key)
    logger.info("tmpfilenamein = " + tmpfilenamein)
    s3conn = boto3.resource('s3')
    s3conn.Bucket('dropbucketlxf').download_file(object_key,
    tmpfilenamein)
    bad_lines = []
    for line in open(tmpfilenamein, 'r'):
        if 'badthings' in line:
            bad_lines.append(line)
    logger.info("Got bad_lines = {}".format(bad_lines))
    dynamodb = boto3.resource('dynamodb')
    table = dynamodb.Table('badones')
    for line in bad_lines:
        table.put_item(Item={'line': line})
    response = {
        "statusCode": 200,
        "body": "Completed scan"
    }
    return response
```

Теперь разверните и протестируйте конвейер, запустив еще несколько событий в API. В частности, несколько раз отправьте ПЛОХОЕ событие, как мы уже делали с помощью утилиты *curl* выше. Эти данные будут сохранены в S3, а затем проанализированы по простому методу, приведенному выше. Чтобы найти результат, загляните в DynamoDB, базу данных с высоким уровнем масштабирования: <https://console.aws.amazon.com/dynamodb/home>.

Всё хорошо, если вам нужно только просмотреть базу данных; ну, а если вы хотите обработать интересующие вас строки?

Обслуживает API

Наконец, давайте создадим API для получения этих строк, с помощью которого приложение или панель мониторинга смогут оповестить вашу команду разработчиков. Снова определим конечную точку:

```
auth:
  handler: handler.get_bad_lines
events:
- http:
  path: get_bad_lines
  method: get
```

Как и прежде, прочитаем это в обратном порядке: при вызове функции `get_bad_lines` вызовите обработчик. Затем напишем обработчик:

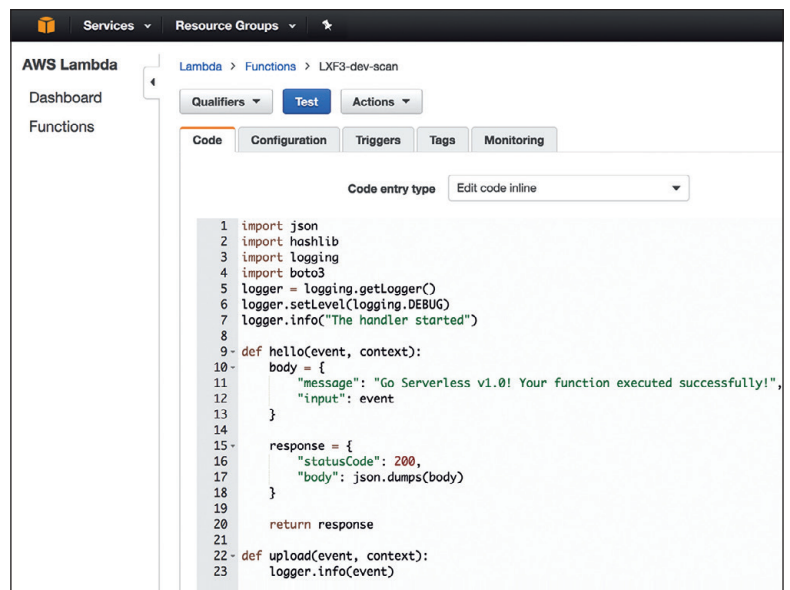
```
def get_bad_lines:
    dynamodb = boto3.resource('dynamodb')
    table = dynamodb.Table('accounts')
    items = table.scan()
    return {
        statusCode: 200,
        Body: json.dumps(items)
    }
```

Этот код просто загружает таблицу DynamoDB и возвращает все результаты. Разверните приложение с помощью метода `serverless deploy -v` и найдите новый метод в разделе 'endpoints' ближе к концу вывода. Обработайте получившийся вывод утилитой *curl*, как в прошлый раз.

Без конечных точек

Я хочу подчеркнуть, что мы не разворачивали никакого оборудования и до настоящего момента потратили сущие копейки. Но нам удалось создать две конечные точки, которые после небольшой правки можно использовать в крупных рабочих системах. Вы пользовались знакомыми инструментами, которые требуют крайне малых изменений для использования в облаке.

Я призываю вас попробовать фреймворк Serverless для любого web-проекта прямо сейчас. Скорость и дешевизна разработки делают его отличным продуктом. **LXF**



```
1 import json
2 import hashlib
3 import logging
4 import boto3
5 logger = logging.getLogger()
6 logger.setLevel(logging.DEBUG)
7 logger.info("The handler started")
8
9 def hello(event, context):
10     body = {
11         "message": "Go Serverless v1.0! Your function executed successfully!",
12         "input": event
13     }
14
15     response = {
16         "statusCode": 200,
17         "body": json.dumps(body)
18     }
19
20     return response
21
22 def upload(event, context):
23     logger.info(event)
```

➤ Создание микросервиса Serverless, готового для развертывания в AWS Lambda.

Очень низкие цены

Вы можете перейти на Lambda и потратить всего несколько долларов на полосу пропускания и хранилища.

Бесплатный уровень Lambda включает 1 млн бесплатных запусков и 400 000 ГБ-секунд вычислительного времени в месяц. Вычисление тарифицируется в ГБ-секундах — это означает, что чем больше памяти использует процесс, тем выше стоимость. Но я бы не приступал к оптимизации слишком рано, так как 1536 млн запусков стоят

всего около \$0,000002501 на 100 мс выполнения, и вы получите 266667 запусков бесплатно.

Другие затраты, которые вы понесете, включают полосу пропускания (трафик к конечным точкам API и S3), хранилище S3 (расходы могут быть большими для большого объема данных) и вычислительные службы или обслуживание данных, такие как RDS.

Вероятно, что сайты с более постоянными требованиями к нагрузке, такие как блоги и электронные

магазины, могут не подойти для Serverless и привести к увеличению расходов. Если вы планируете использовать Serverless для более крупных систем, вам необходимо понять, когда выполняются ваши методы и сколько памяти они используют.

Тем не менее, любую серверную часть API для приложения, системы обработки бэк-офиса или системы интеграции намного проще реализовать через Serverless, чем с помощью обычных серверов.

Подпишись на Linux Format



Журналу
Linux Format
12 лет!

Издается в России
с 2005 года

Печатная
версия



Электронная
версия (PDF)

Годовая подписка
4500 ₽*

Полугодовая подписка — 2430 ₽*

БОНУС

Подписка
на PDF-
версию
журнала

ДОСТАВКА

200 пунктов
самовывоза
в 50 городах
России

БОНУС

Образ DVD-
приложения
к каждому
номеру

БОНУС

Получение
каждого номера
в день выхода
журнала

* Цена не включает стоимость доставки.

ПОДРОБНОСТИ НА САЙТЕ
WWW.LINUXCENTER.RU/LINUXFORMAT



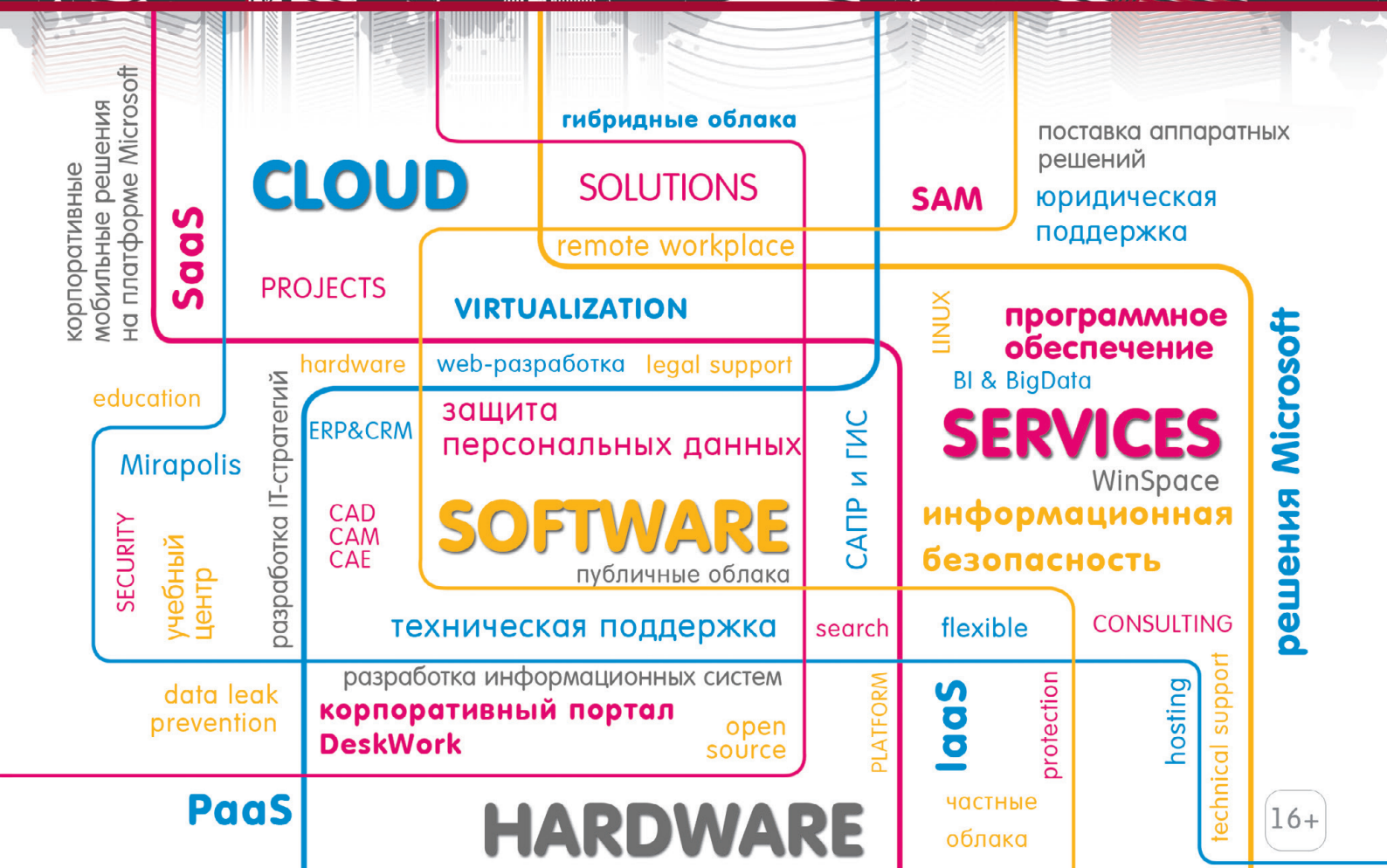
softline®

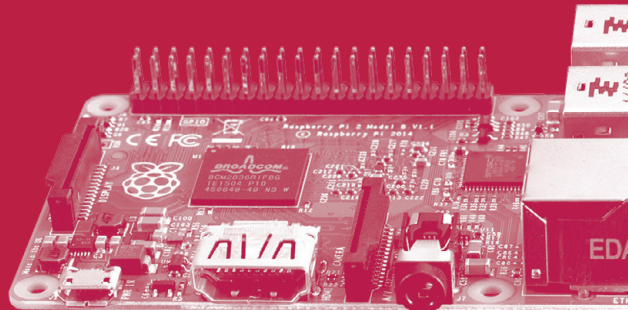
Cloud Software Hardware Services

20+

Years in IT

IT-архитектура вашего бизнеса





Ваша порция смачных новостей, обзоров и учебников от Raspberry Pi

Энди Уорбертон
— амстердамец,
web-дизайнер
и отец.



Привет...

Привет, я Энди, мы с семьей живем в Амстердаме. Вообще я web-дизайнер, но также люблю повозиться с Raspberry Pi!

С Linux я связан почти 20 лет: сначала работал с традиционным стеком LAMP, в качестве web-дизайнера и разработчика, а совсем недавно всерьез занялся физическими проектами с Pi. Год назад я купил Pi 3 для сборки ретро-аркады, которую мы делали вместе с сыном, а потом понял, что мне вообще нравится возиться с оборудованием. С тех пор я создавал множество всё более сложных проектов, в том числе влихнул Pi Zero W в контроллер SNES и соорудил сетевой динамик для нашей гостиной. Сейчас делаю Robot Rover из поломанного автомобиля-игрушки R/C!

Сообщество Pi поистине очень открытое и дружелюбное. От компаний-производителей, таких как Pimoroni и Adafruit, до просто любителей в Twitter — все готовы помочь новичкам. Лично я больше всего почерпнул из примеров GitHub и учебников в YouTube. Там так много информации, что даже тот, кто отродясь и не прикасался к Linux, вскоре сможет делать потрясающие проекты!

Познакомившись с командой организаторов Berlin Raspberry Jam на берлинской Maker Faire, я загорелся идеей сделать первый в Амстердаме Raspberry Jam (<http://amsterijam.me>). Очень помогла поддержка сообщества — есть группа Slack, специально для тех, кто организует Raspberry Jam. Механизм очень отлаженный, и это возможно только благодаря потрясающему сообществу!

Смута вокруг Arduino, наконец, поутихла

Все товарные знаки теперь принадлежат холдингу BSMI.

Платы Arduino снижали большую популярность в сообществе умельцев, благодаря своей доступности, низкому энергопотреблению и открытому коду. Однако раскол и битва за товарный знак осложнила работу сообщества и разделила его. До недавнего времени существовало две компании: Arduino LLC (<http://arduino.cc>, созданная пятью соучредителями проекта Arduino) и Arduino SRL (<http://arduino.org>, ранее Smart Projects, созданная соучредителем Arduino Джанлуко Мартино [Gianluco Martino]) и недавно возглавленная Федерико Мусто [Federico Musto]). На определенном этапе компания LLC была вынуждена представить новый (европейский) товарный знак Genuino, чтобы огрардить поставщиков от судебных разбирательств. В прошлом году две фирмы официально зарыли топор войны,

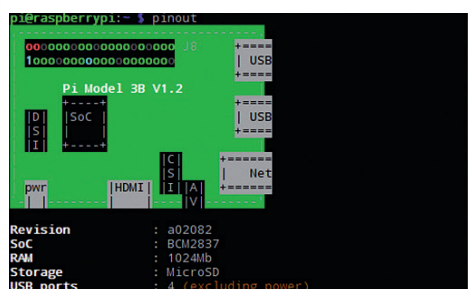
и все товарные знаки перешли к новой холдинговой корпорации, Arduino AG. Теперь эти товарные знаки принадлежат группе BSMI, созданной четырьмя соучредителями, кроме Мартино. По словам Максимо Банци [Massimo Banzì] из BSMI: «Это начало новой эры для Arduino». Мы тоже с нетерпением ждем побольше новых гаджетов и поменьше судебных тяжб. Подробности в официальном блоге: <https://blog.arduino.cc/2017/07/28/a-new-era-for-arduino-begins-today/>



Обновленный GPIO Zero

От всех макетных бед!

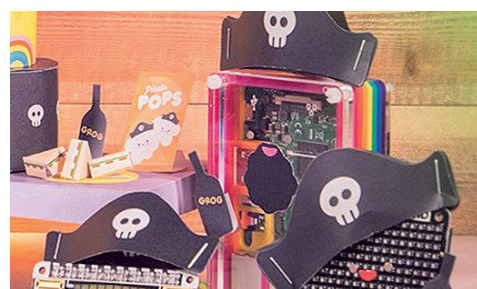
Удобнейший модуль GPIO Zero Python обновлен до версии 1.4. То, что начиналось как простая обертка для не очень дружелюбной к новичкам библиотеки Rpi.GPIO, выросло в сильную самостоятельную особь. Начинающие программисты теперь могут использовать управляемые событиями или декларативные методы для включения и выключения напряжения на GPIO, и мир снова в безопасности! www.raspberrypi.org/blog/gpio-zero-update/

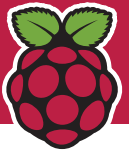


С днем рождения, Pimoroni!

Йоркширский Pi удался на славу.

В августе базирующийся в Шеффилде Pimoroni отпраздновал свой пятый день рождения. Из малька в водах сообщества умельцев он вырос до чего-то гораздо более внушительного: в его проектировании, производстве и продаже теперь задействовано около 40 человек. Пол Бич [Paul Beech] из Pimoroni является автором логотипа Raspberry Pi, а также культового корпуса PiBow. www.raspberrypi.org/blog/pimoroni-is-5-now/





Thunderborg

Лес Паундер отправлен из будущего, чтобы обеспечить приход к власти Skynet, и он нашел источник силы искусственного интеллекта...

Вкратце

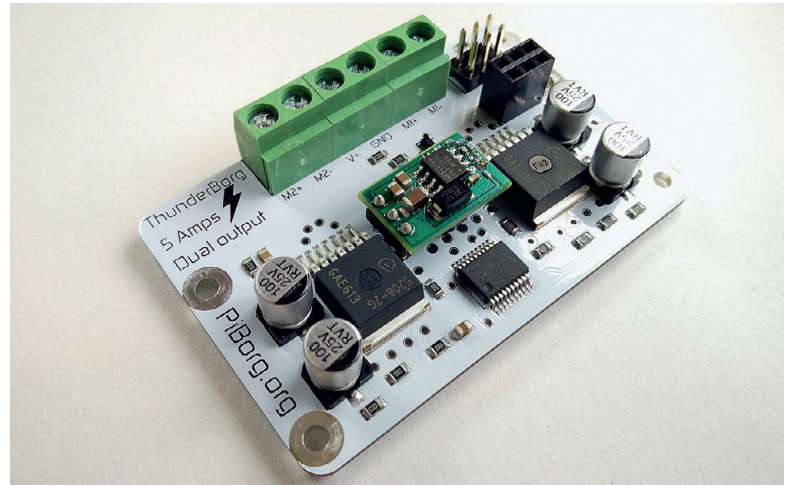
» Плата контроля и управления мотором, разработанная для больших и малых проектов робототехники. Thunderborg подключается всего к шести контактам GPIO, но предлагает обширную и простую в использовании библиотеку Python, которую все пользователи, независимо от их умений, могут применить для простой реализации проектов робототехники.

Роботам еще предстоит захватить власть над миром, но Raspberry Pi они уже захватили! Несть числа проектам робототехники в сообществе, от простых ботов до мощных и умных дронов, способных сами собой управлять. И на передовой этих мощных роботов — PiBorg, который производит платы для всех моделей Pi, и сейчас компания представила свою самую мощную плату. Шаг вперед, Thunderborg!

Thunderborg немного больше Pi Zero, но имеет монтажные отверстия, чтобы на него можно было смонтировать Pi Zero, да и другие модели Pi. На плате есть коннектор, соединяющий первые шесть контактов вашего Pi с платой, предоставляя интерфейс управления I2C и питания через контакты 5 В и GND [Земля]. Встроенные регуляторы питания Thunderborg (5 В) позволяют безопасно запитать ваш Pi через GPIO. Напряжение питания — между 7 В и 35 В, через клеммы с винтовым креплением, так что потребуется внешняя батарея.

Входы Motor 1 и Motor 2 помещены с двух сторон входов питания, их можно подключить к небольшим микроредукторным электродвигателям или к моторам с большим вращающим моментом, на 12 В. Можно даже управлять шаговыми двигателями: это моторы с высокой степенью точности, допускающие очень тонкий контроль (правда, медленные).

Плата использует два мостовых контроллера моторов 5206-2G H, которые могут запитать моторы, требующие тока до 5 А, максимально 6 А, и до 40 В! Это огромная мощность для такой маленькой платы.



» Thunderborg — впечатляюще мощная плата, при этом простая в установке и работе. Ее доступность позволит любому пользователю быстро ее освоить.

С помощью Thunderborg можно управлять двумя моторами, но благодаря умной системе расширений можно подключить и другие платы Thunderborg — для управления до 200 моторов с одного Raspberry Pi.

Установка программ проходит со свистом, в основном благодаря однострочному скрипту установки с сайта PiBorg, но, конечно, при установке программ с помощью таких скриптов всегда будьте осторожны. Единственный недочет, который мы заметили у программы — она устанавливается для Python 2, а не для более современного [Ред.: — Всего-то 9 лет!] Python 3. Оно, конечно, не губительно, но неприятно для тех, кто учится программировать на более новой версии.

Мы протестировали плату на двух моторах: микроредукторном и более крупном 5-В моторе из набора для сборки робота. Сначала мы использовали GUI для управления скоростью каждого мотора с помощью ползунка. Это работает с помощью PWM (pulse width modulation — широтно-импульсная модуляция, ШИМ), которая изменяет рабочий цикл для создания постепенного увеличения или уменьшения скорости.

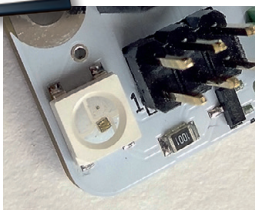
Создать собственный код для моторов поможет простая в использовании библиотека Python. Чтобы настроить код Python для определения платы, надо всего лишь импортировать библиотеку, создать объект и затем инициализировать плату, и всё это в трех строках кода. Затем, чтобы задать скорость моторов, вам нужно присвоить процент скорости каждому мотору.

Скорость каждого мотора можно проверять и сообщать пользователю — очень удобно для обратной связи. Библиотека также контролирует встроенный светодиод (LED) neopixel, который по умолчанию показывает статус заряда: если он красный — мощность низкая, а если зеленый — максимальная. Вы можете проверять состояние батареи с помощью библиотеки.

Thunderborg определенно является мощной платформой для робототехники. Плата проста в использовании и настраивается без особо сложных действий, которые могут запутать пользователя. Было бы замечательно, если бы по умолчанию устанавливалась библиотека для Python 3, но этот мелкий недочет не должен отвратить вас от замечательной платы. LXF

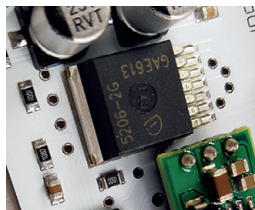


Свойства навскидку



Индикатор LED

В составе Thunderborg — отличные светодиоды neopixel, а благодаря библиотеке Python с ними легко управляться.



Контроллеры мотора

Они позволяют ставить на плату маленькие и большие моторы. Для большинства пользователей мощности хватит.

LINUX FORMAT Вердикт

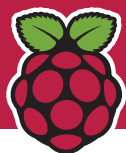
Thunderborg

Разработчик: PiBorg
Сайт: www.piborg.org/thunderborg
Цена: От £37

Функциональность	9/10
Производительность	9/10
Удобство в работе	9/10
Справедливость цены	9/10

» Производители Pi должны дать шанс этой простой встраиваемой модели, идеальной для управления энергопотреблением и робототехники.

Рейтинг 9/10



Pi Zero: Делаем сервер печати

Лес Паундер показывает, как оживить старый принтер, подключив его с помощью Wi-Fi и Raspberry Pi Zero W к своей домашней сети.



Наш эксперт

Лес Паундер — любитель мастерить, и он в восторге от того, на что способен Pi. Лес путешествует по Великобритании с Raspberry Pi Foundation, пропагандируя Pisademy, и размещает свои изобретения на <http://bigles.es>.

Обычно Raspberry Pi Zero W используется для создания встраиваемых проектов, роботов и т.д., в основном благодаря своему малому размеру и беспроводному соединению. Но как эти его свойства помогут нам сократить электронные отходы?

Что ж, помните тот принтер, хороший такой, который переехал на антресоли? Тот, у которого нет печати через Wi-Fi? На нашем уроке мы применим Raspberry Pi Zero W в качестве сервера печати CUPS, через который сможем печатать с нашего ноутбука с Linux (Ubuntu). Мы подключим Pi Zero W к нашему принтеру, и он будет обслуживать запросы принтеру на печать.

Настройка Pi Zero W

Наша первая задача — установить Raspbian Lite на карту microSD. Raspbian Lite — урезанная версия Raspbian, которая не предлагает среды рабочего стола, и ее можно установить на SD-карты емкостью всего 2 ГБ. Лучший инструмент для переноса Raspbian Lite на SD-карту — *Etcher* (<https://etcher.io>), потому что это очень простой GUI для такой непростой задачи.

Создав карту, вставьте ее в Raspberry Pi Zero W, затем подключите HDMI и клавиатуру. И, наконец, подключите питание: у вашего Pi уйдет пара мгновений на загрузку, и он выдаст вам экран приглашения. Имя пользователя и пароль — стандартные “pi” и “raspberrypi”. Войдя в систему, не поленитесь сменить пароль: работая с Pi в сети, всегда надо помнить о безопасности.

В терминале введите `$ passwd`, затем, когда вам предложат, введите свой текущий пароль, а за ним — новый пароль. Вам предложат подтвердить правильность пароля.

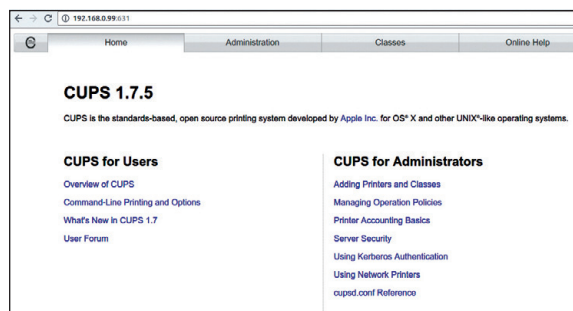
Пока что наш Pi не в сети и не в Интернете. Чтобы это исправить, надо настроить Wi-Fi. В терминале мы отредактируем нашу настройку `wpa_supplicant` — она позволит Pi Zero W разговаривать с роутером. Введите следующее:

```
sudo nano /etc/wpa_supplicant/wpa_supplicant.conf
Теперь воспользуемся текстовым редактором nano. С помощью клавиш со стрелками перейдите в низ файла.
Добавьте следующие строки и измените “YOUR WIFI SSID” и “YOUR WIFI PASSWORD” на соответствующие в вашей настройке.
network={
  ssid="YOUR WIFI SSID"
  psk="YOUR WIFI PASSWORD"
}
```

Теперь нажмите Ctrl+X, введите ответ “Y” и затем нажмите Enter, чтобы сохранить и выйти. Кстати, самое время перезагрузиться, чтобы убедиться: наши изменения были внесены правильно. Поэтому введите

```
$ sudo reboot
```

Затем, как только Pi перезагрузится, снова войдите в систему. Подключившись к Интернету, давайте обновим ПО на нашем Pi. Raspbian Lite, как и предполагает название, основан на Debian.



➤ Это не для горячих напитков: *CUPS* [cup — *англ.* чашка] позволит вам печатать с вашего Pi Zero W на любом старом принтере.

Инструмент управления пакетами называется *apt* (Advanced Packaging Tool), и сначала мы обновим список устанавливаемых программ. В терминале введите:

```
$ sudo apt update
```

После этого обновим ПО, установленное на нашем Pi. Параметр `-y` дает Pi указание автоматически устанавливать программы:

```
$ sudo apt upgrade -y
```

Наш финальный шаг с *apt* — установить систему печати *CUPS* (Common Unix Print System). *CUPS* разработана как технология с открытым кодом в Apple, хотя изначально это был независимый проект. Чтобы установить *CUPS*, введите в терминал следующее:

```
$ sudo apt install cups
```

Когда *CUPS* установится, надо убедиться, что наш пользователь по умолчанию “pi” находится в нужной группе, чтобы печатать через *CUPS*. Для изменения пользователя мы воспользуемся `usermod`, и добавим пользователя (-a) в группу (-G) `lpadmin`.

```
$ sudo usermod -a -G lpadmin pi
```

Следующий шаг — задать статический IP-адрес для нашего Pi. Это поможет нам определять местоположение Pi в нашей сети. В терминале введите

```
$ sudo nano /etc/dhcpd.conf
```

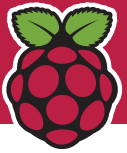
Прокрутите в низ файла, где мы теперь запишем раздел, который задает статический IP-адрес для беспроводного интерфейса (wlan0) нашего Raspberry Pi. Вам надо будет изменить значения, чтобы они соответствовали вашей сети. В нашем случае мы изменили `ip_address` на `192.168.0.99`.

```
interface wlan0
static ip_address=192.168.0.99/24
static routers=192.168.0.1
static domain_name_servers=192.168.0.1
```

Нажмите Ctrl+X, затем ответьте Y и нажмите на Enter, чтобы сохранить и выйти. Теперь давайте используем SSH, чтобы отсоединить клавиатуру от Pi Zero W. Нам надо включить SSH-сервер с помощью инструмента настройки Raspberry Pi Configuration. В терминале введите

Вам нужны

- » Raspberry Pi Zero W
- » Raspbian Lite
- » Подходящая 2-ГБ+ SD-карта
- » USB-клавиатура
- » Кабель mini-HDMI—HDMI
- » Старый принтер
- » Интернет
- » Pi PSU
- » Адаптер microUSB—USB



Добавление принтера к вашему Linux-ПК

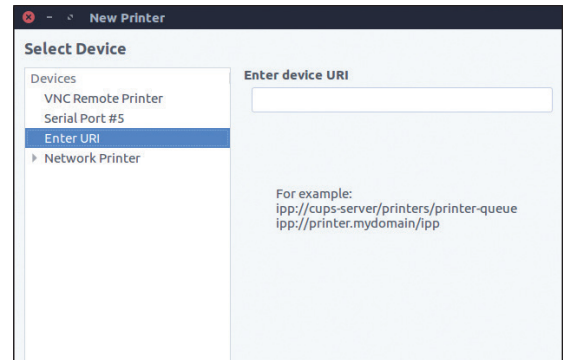
У нас есть сетевой принтер, так давайте скажем Ubuntu, где его искать.

Мы используем Ubuntu, но эти инструкции сработают и с другими дистрибутивами Linux, хотя некоторые шаги могут немного отличаться. То же самое относится к использованию CUPS с устройствами Apple и Windows. CUPS будет работать с этими операционными системами, но потребуется немного больше настройки. Если вам нужна дополнительная помощь, установка CUPS на вашем Raspberry Pi имеет отличную систему подсказки; но вы также можете заглянуть на сайт CUPS (www.cups.org) ради самой последней информации и поддержки.

Откройте приложение *Printers*, нажав на шестеренку System Settings в верхней

правой части экрана. Затем нажмите на значок *Printers* в новом окне. Если принтер не определился автоматически, нажмите *Add*, и в следующем диалоговом окне вы должны увидеть принтер в списке Network Printer [Сетевой принтер].

Нажмите *Forward* и дайте принтеру имя, описание и местоположение, соответствующие тем, которые мы использовали при установке принтера с CUPS. И, наконец, нажмите *Apply* — и ваш принтер готов к использованию с любыми приложениями. Вам только нужно выбрать принтер, щелкнуть правой кнопкой по значку принтера и затем установить принтер по умолчанию. Теперь при каждом обращении к печати файлы будут отправляться на новый сетевой принтер.



➤ **Добавить принтер на наш ноутбук с Ubuntu было просто: на самом деле, сервер печати определился автоматически, и установка прошла прямо по-Божески.**

```
$ sudo raspi-config
```

Затем с помощью клавиш со стрелками перейдите в *Interfacing Options* и нажмите на *Enter*. В новом меню перейдите к *SSH* и снова нажмите на *Enter*. Включите *SSH*, выбрав *Yes*, и нажмите на *Enter*. И можете выйти из меню, перейдя снова к началу и выбрав *Finish*. Отныне мы можем удаленно управлять нашим Pi через *SSH*, поэтому давайте перезагрузим Pi и проверим нашу новую настройку:

```
$ sudo reboot
```

Теперь можно отсоединить клавиатуру от Pi Zero W и подключить принтер к свободному USB-порту. Затем включите принтер и убедитесь, что Pi Zero W видит это подключение.

Подключите компьютер

Пора перейти к другому компьютеру, с которого мы будем печатать. В нашем случае это ноутбук с Ubuntu. С помощью терминала на вашем компьютере Linux или приложения типа *PuTTY* для Windows войдите через *SSH* на ваш Pi, используя статический IP-адрес, заданный вами ранее:

```
$ ssh pi@192.168.0.99
```

Войдя в систему, вы можете администрировать ее и запускать обычные обновления, чтобы поддерживать систему в безопасности, если это необходимо. А клиент *SSH* можно закрыть, потому что он нам больше не нужен.

На своем компьютере с Linux откройте браузер и перейдите на IP-адрес своего Pi Zero W, за которым должно следовать 631. У нас это 192.168.0.99:631.

Вы увидите домашнюю страницу с вкладками в верхней части экрана и тремя разделами в центре. Нашей первой задачей с CUPS будет настройка нашего подключенного принтера. Нажмите на вкладку *Administration* и на следующем экране — на *Add Printer* [Добавить принтер]. При первом использовании вы получите

предупреждение 'Требуется обновление [Upgrade Required]'; не пугайтесь, просто нажмите на предложенную ссылку. Это может привести к ошибке 'Your connection is not private [ваше соединение не является частным]'. Если это произойдет, нажмите *ADVANCED* и затем выберите *Proceed to IP ADDRESS* [Перейти на IP-адрес]. Далее вам предложат ввести имя пользователя и пароль — они те же, что мы использовали ранее.

На экране *Add Printer CUPS* должен суметь определить ваш принтер, особенно если вы используете старый принтер, что и является целью данного проекта. Если принтер найден, нажмите на радио кнопку рядом с ним и затем нажмите *Continue* [Продолжить].

На следующем экране дайте принтеру имя, описание и местоположение. Затем отметьте кнопку *Sharing* [Общий]. Нажмите *Continue*, чтобы перейти к следующему экрану. Здесь мы зададим модель принтера, которая должна определиться автоматически; а если нет, то выберите свою. Затем нажмите *Add Printer*.

Наш следующий экран установит для принтера опции по умолчанию. Это ваши личные предпочтения, и, возможно, вам придется немного повозиться с настройкой, чтобы добиться лучшего качества печати. После окончания нажмите *Set Default Options* [Установить опции по умолчанию]. Теперь ваш принтер готов к использованию. Нажмите на гиперссылку, чтобы открыть настройки принтера, и в *Maintenance* [Управление] выберите *Print Test Page* [Печать тестовой страницы]. Через несколько мгновений на вашем новом беспроводном принтере будет свежееотпечатанная страница!

Теперь у вас есть беспроводной сервер печати, работающий на Raspberry Pi Zero W. Вы можете печатать через него, используя любой компьютер — с Linux, Windows или Apple. А еще у вас освободилось место на антресолях — можете отнести туда всякий хлам, который скопился в прихожей и лежит там с самого вашего переезда... LXF

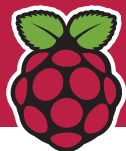
Скорая помощь

Этот проект можно также реализовать на других моделях Pi. Полезно при использовании более старых моделей Pi — например, исходной модели. Измените настройки статического IP-адреса с wlan0 на eth0, потому что в ней нет Wi-Fi.



➤ **Спасенный с антреселей и возвращенный к жизни, этот старый принтер Epson делает то, что ему всегда удавалось лучше всего (12 лет назад).**





Ufw: Брандмауэр «ИЗ ПРОСТЫХ»

Нейт Дрейк превращает ваш Pi в ультра-безопасную точку доступа в комплекте с брандмауэром, чтобы защитить ваш компьютер от темных сил Интернета.



Наш эксперт

Нейт Дрейк привык пополнять свой скудный доход, предлагая услуги по восстановлению документов с помощью *scalpel*. Его первым клиентом была танцовщица бурлеска. С тех пор он и заскользил по наклонной плоскости...

Вы, наверное, помните эту новость. В декабре 2016 г. BBC сообщила, что печально известный червь *Mirai* заразил немало маршрутизаторов, используемых клиентами британского интернет-провайдера *TalkTalk*, ворую их пароли Wi-Fi. Червь использовал пароли администратора по умолчанию, жестко привязанные к нескольким машинам, и хотя целью был захват устройств для их употребления в атаках DOS (отказ в обслуживании), последствия ошеломляют. Даже те немногие пользователи, которые утруждаются изменением у себя паролей по умолчанию, редко заботятся о том же для межсетевых экранов.

Просту говоря, это означает, что настройки вашего маршрутизатора и теоретически устройства в вашей домашней беспроводной сети могут быть взломаны злоумышленниками. К счастью, на помощь опять приходит *Raspberry Pi*. Мы покажем вам, как.

Наш проект можно разделить на две части. Во-первых, настройте свой Pi как беспроводную точку доступа. Другими словами, Pi сам создаст беспроводную сеть для подключения. Затем Pi можно подключить к вашему маршрутизатору, и любой подключенный к его беспроводной сети может получить доступ к Интернету.

Вторая часть этого проекта включает установку на Pi простого в использовании брандмауэра, блокирующего любые открытые порты, до которых могут добраться злоумышленники. Вы также можете настроить брандмауэр для доступа к законным программам и заблокировать любые сайты, которые, по вашему мнению, небезопасны. Имея подходящее оборудование, вы можете получить собственную защищенную точку доступа с брандмауэром и начать работать с ней менее чем за 20 минут.

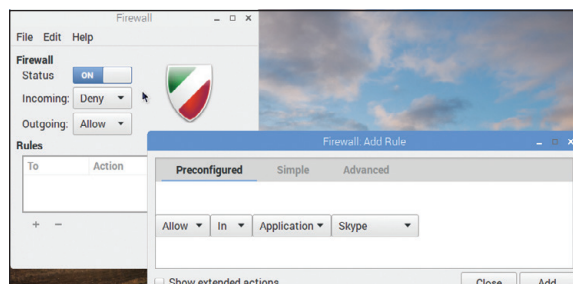
Необходимый минимум

Используемое беспроводное устройство должно быть совместимо с *hostapd* (немного об этом). Наш проект технически совместим со всеми моделями Pi, хотя Pi должен быть с поддержкой беспроводной сети и иметь порт Ethernet для подключения к маршрутизатору. *Raspberry Pi 3* хорошо себя в этом зарекомендовал.

Настройка, если это вообще возможно, должна выполняться с помощью клавиатуры и монитора, а не по SSH, поскольку, как только Pi создаст собственную точку доступа, вы не сможете подключиться к Pi по беспроводной сети. Однако вы можете подключиться к нему через беспроводную сеть вашего маршрутизатора, если знаете IP-адрес.

Если вы предпочитаете компактный Pi Zero, убедитесь, что у вас есть USB-кабель OTG Host и USB-адаптер MicroUSB OTG (доступный у всех поставщиков аксессуаров для Pi — от Pi Hut, например, всего за £2, <https://thePIhut.com/products/usb-to-microusb-otg-converter-shim>). Для подключения к USB-порту вам также понадобится конвертор RJ45-USB. Тогда вы сможете подключить Pi Zero к маршрутизатору. Подробнее — на схеме (стр. 89).

Вам, кроме того, потребуется дополнительное устройство для проверки правильности работы беспроводной точки доступа,



Ufw — напарник *Ufw*. Его преднастроенные правила позволяют использовать наиболее популярные приложения вроде *Skype*, не требуя определять номера портов, которые следует открыть.

а также сетевого кабеля для соединения Pi с маршрутизатором. Большинство современных маршрутизаторов имеют по крайней мере один USB-порт, поэтому вы должны иметь возможность управлять Pi оттуда.

Безопасность прежде всего

По соображениям безопасности и для экономии ресурсов лучше всего для домашнего брандмауэра иметь отдельный Pi и не использовать его еще и для других целей. Если это вообще возможно, подключите Pi к монитору и работайте с этим проектом непосредственно из терминала, а не через SSH. Это уменьшает вероятность обрыва связи на жизненно важной стадии процесса.

Лучше всего начать с чистой установки последней версии *Raspbian* на Pi. По завершении установки откройте терминал и запустите следующие команды — для проверки, что ваша система обновлена:

```
sudo apt-get update
sudo apt-get upgrade
```

Если у вас нет доступа к монитору, можно подключиться к Pi по беспроводной сети Wi-Fi через маршрутизатор и SSH. Сначала подключите Pi к маршрутизатору Ethernet-кабелем. Если у вас есть мобильное устройство, можно использовать удобное приложение *Fing* для простого определения IP-адреса всех устройств в вашей локальной сети. *Fing* доступен для бесплатной загрузки как из магазина приложений iOS, так и из Google Play.

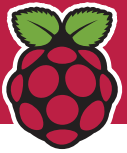
Если вы удовлетворены работой точки доступа, обязательно отключите на своем маршрутизаторе функцию беспроводного доступа, чтобы подключиться можно было не к нему, а только к брандмауэру Pi.

Точка доступа настраивается и управляется удобной программой *hostapd*. Шаг 3 нашего руководства на стр. 91 описывает создание файла конфигурации, где вы укажете параметры новой беспроводной сети Pi, такие как имя, пароль и тип шифрования. К этому *hostapd.conf* всегда можно вернуться, чтобы изменить настройки, если захотите.



Скорая помощь

Чтобы узнать, поддерживает ли ваша беспроводная карта режим точки доступа, откройте терминал и запустите команду `iw list`. Если поддерживает, то под 'valid interface combinations' вы должны увидеть буквы AP.



Хотя любая беспроводная карта может подключить вас к сети Wi-Fi, не все карты Wi-Fi могут создавать собственные беспроводные сети. К счастью, модуль Wi-Fi, встроенный в Raspberry Pi 3, поддерживает данную функцию, и это хорошая причина рассмотреть вопрос его применения как вашего брандмауэра.

Мы также установим дьявольски умную программу *dnsmasq* для работы с сетевой инфраструктурой. Она также берет настройки из небольшого файла конфигурации, который вы создадите на шаге 4. В предлагаемых настройках *dnsmasq* использует Google для своего DNS-сервера (8.8.8.8), но, если хотите, вы можете это изменить на другого поставщика DNS, такого, как OpenDNS (208.67.222.222). Чтобы обеспечить максимальную производительность, можно указать более одного DNS-сервера, просто добавив новые строки для них в файл *dnsmasq.conf*, например,

```
server=208.67.222.222
server=208.67.222.220
```

Несложный брандмауэр

Ufw (Uncomplicated Firewall, несложный брандмауэр) поставляется с рядом дистрибутивов Linux и оправдывает свое название — он очень прост в использовании. Это отнюдь не единственная программа брандмауэра: на самом деле в природе есть целые дистрибутивы Linux, такие как IPFire, специально предназначенные для работы шлюзом между вашим компьютером и интернетом, с красочными окнами. Зачем тогда использовать довольно-таки прозаическое приложение командной строки? Помимо того, что *Ufw* выдержал испытание временем, его легко установить и настроить.

После включения *Ufw* запустится автоматически. По умолчанию блокируются все входящие соединения и разрешаются все исходящие. Это может досадить, если вы и другие пользователи вашей сети захотите использовать приложения вроде *BitTorrent* или *Skype*; но, к счастью, настройки исключений для этих правил просты. У *Ufw* также есть графический напарник для тех, кому еще не совсем комфортно в командной строке. Дополнительную информацию см. во врезке «Настройка брандмауэра» на стр. 90.

Ufw также имеет возможность блокировать подключения к заданным IP-адресам или диапазонам, так что вы можете запретить

пользователям сети посещать определенные сайты — см. «Блокировка домена» (врезка внизу).

Хотя *Ufw* можно использовать для блокировки отдельных сайтов, любые загружаемые web-страницы будут показывать рекламные баннеры и всплывающие окна. На самом деле простейший способ справиться с ними для вас и других пользователей в вашей сети — установить расширения браузера, типа Ublock Origin и Ghostery: они предотвращают и рекламу, и отслеживание куки [cookie]. Мобильные пользователи также могут воспользоваться официальным *Adblock Browser*.

Если это неосуществимо, можно удалить DHCP (протокол динамической настройки узла) для Pi по умолчанию и настроить вместо него *dnsmasq* для автоматического перенаправления известных рекламных серверов на внутренний IP-адрес. По умолчанию это создаст уродливые бельма в середине ваших web-страниц, но эту проблему можно решить, установив также удобное приложение *pixerv*. Единственной целью его жизни является поддержка блокировки рекламы посредством замены ее на прозрачное изображение GIF размером 1×1. Это заставляет блокировку рекламы выглядеть опрятнее.

Если вы заинтересованы в добавлении функции блокировки рекламы в Pi, посетите web-сайт Adafruit: <https://learn.adafruit.com/raspberry-pi-as-an-ad-blocking-access-point>.



Блокировка домена

Ufw — это мощный межсетевой экран, он может использоваться для блокировки доступа к определенным сайтам. Правда, это также делается с помощью *dnsmasq*, но в форме «перехвата DNS», что является довольно дурной процедурой, обычно выполняемой мошенниками.

Преимущество использования *Ufw* заключается в том, что он блокирует домены по IP-адресу. Это означает, что вам нужно записать только одно правило на адрес, а не помнить различные домены и поддомены, используемые сайтами. Представьте, что вы пытаетесь заблокировать все варианты [Facebook.com](https://www.facebook.com), например — fb.com, m.facebook.com...

Разумеется, для блокирования сайтов по IP-адресу вам эта информация понадобится. Откройте терминал на своем Pi или подключитесь через SSH и используйте

команду `host` для просмотра IP-адреса нежелательной страницы — например:

```
host strawberryfunk.com
```

Вы получите IP-адрес запрашиваемого сайта. Затем используйте `ufw deny` для блокировки этого домена, например:

```
sudo ufw deny out from any to 205.178.189.29
```

Можно проверить, что домен успешно заблокирован, с помощью `ping`:

```
ping -c 1 strawberryfunk.com
```

Если вы правильно настроили *Ufw*, `ping` должен выдать вам сообщение, что домен недоступен.

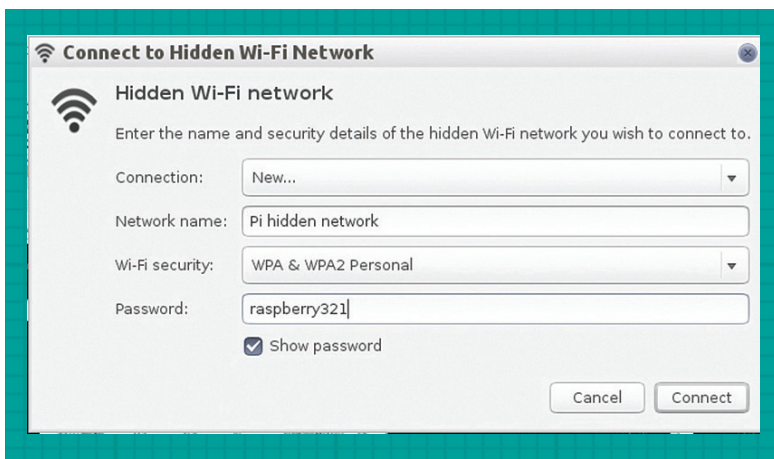
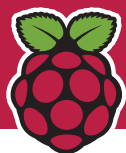
Ufw также способен блокировать исходящие подключения к определенным вами портам. Например, желая уменьшить вероятность использования Pi для отправки спам-сообщений, вы можете отключить порт 25 следующим образом:

```
sudo ufw deny out 25
```

```
fish /home/pi
pi@raspberrypi ~-> host strawberryfunk.com
strawberryfunk.com has address 205.178.189.129
strawberryfunk.com mail is handled by 10 p.web
pi@raspberrypi ~-> sudo ufw deny out from any t
Rule added
pi@raspberrypi ~-> ping -c 1 strawberryfunk.com
PING strawberryfunk.com (205.178.189.129) 56(8
--- strawberryfunk.com ping statistics ---
1 packets transmitted, 0 received, 100% packet
pi@raspberrypi ~-> █
```

» ping проверит, что домен успешно заблокирован. Некоторые сайты имеют более одного IP-адреса, поэтому поочередно блокируйте их.

» Подпишитесь на печатную или электронную версии на www.linuxformat.ru/subscribe!



Подключение к скрытой сети немного усложняет жизнь другим пользователям сети, но гораздо больше — хакерам, потому что им нужны и сетевое имя, и пароль.

Наведение мостов

Наш проект фокусируется на соединении беспроводного интерфейса Pi и интерфейса Ethernet. Это позволяет вашей точке доступа оставаться подключенной к Интернету. Если вам надо пере- проверить имена своих интерфейсов, вы всегда можете запустить команду `ifconfig`. По умолчанию имена этих интерфейсов должны быть `wlan0` и `eth0` соответственно.

Если вы не можете подключить Pi к маршрутизатору, теоретически можно добавить еще один беспроводной интерфейс, например, официальный модуль Wi-Fi для Raspberry Pi, и использовать его для доступа к беспроводной сети вашего маршрутизатора. Однако это менее безопасно и снизит вам скорость соединения. Если ваш маршрутизатор где-то в недоступном месте (или вообще вне зоны досягаемости), подумайте об использовании устройства типа “hoterplug”, которое использует для сетевых подключений электропроводку вашего дома или офиса.

Не отчаивайтесь, если точка доступа или брандмауэр не работают при первом выполнении шагов. Сетевые настройки Pi очень гибкие, и вы можете вернуться снова по шагам, не переустанавливая Raspbian.

Если вы используете драйвер Wi-Fi, не встроенный в Pi 3, или официальный адаптер Wi-Fi для Raspberry Pi, возможно, придется изменить значение `driver=` в файле `hostapd.conf`. За информацией обратитесь к сайту <http://linuxwireless.org/en/users/Documentation/hostapd>.

Вы можете отключить брандмауэр `Ufw` навсегда и удалить все правила такой командой:

```
sudo ufw reset
```

При необходимости вы также можете восстановить ранее сохраненные настройки с помощью `dnsmasq`, изменив исходную команду:

```
sudo mv /etc/dnsmasq.conf.orig /etc/dnsmasq.conf
```

Если вы решите заблокировать определенные порты или службы — см. врезку «Блокировка домена» внизу стр. 89 — то имейте в виду, что ИТ-грамотные пользователи в вашей сети могут обойти это с помощью VPN. Разумеется, через `Ufw` можно заблокировать порты, обычно используемые VPN, например, 1154, или блокировать весь исходящий трафик, а затем включать его для определенных приложений и служб. Некоторые приложения будут использовать порты случайным образом, затрудняя блокировку определенных протоколов, таких как BitTorrent. Желая предотвратить потоковое воспроизведение большинства музыкальных и видеосайтов, вы можете предусмотреть блокировку протокола UDP.

Если вы решите использовать для блокировки рекламы расширение браузера или `pixelserv`, то некоторые сайты могут отбражраться некорректно или вообще не загружаться. В этом случае расширения блокировщика рекламы обычно можно временно отключить, щелкнув по его значку в меню вашего браузера.

По соображениям безопасности было бы разумно использовать выделенный Pi исключительно как точку доступа и не хранить на нем какие-либо личные данные. Для дополнительной безопасности обязательно используйте длинный надежный пароль — приведенный в руководстве является только примером.

Для полной безопасности подумайте о создании скрытой сети Wi-Fi для Pi. Отредактируйте файл `hostapd.conf`, открыв терминал и набрав

```
/etc/hostapd/hostapd.conf
```

```
Замените ignore_broadcast_ssid=0 на ignore_broadcast_ssid=1.
```

Теперь на всех устройствах надо ввести имя сети, а также пароль для подключения. При редактировании файла подумайте о выборе нового имени, для обеспечения дополнительного уровня безопасности.

Если у вас есть устройства, использующие кабельное соединение, например, домашний сервер, можно оставить их подключенными непосредственно к маршрутизатору для доступа в Интернет, но имейте в виду, что они не будут видны в беспроводной сети Pi. Однако в этом есть и другая сторона: Pi не будет мешать их действиям.

Наконец, имейте в виду, что ваш маршрутизатор также может использовать брандмауэр. Если вам удобно настраивать маршрутизатор, вы можете открывать и закрывать порты в соответствии с настройками `Ufw`. Кроме того, можно полностью отключить брандмауэр маршрутизатора и позволить Pi управлять всем. **LXF**



Узнать обо всех возможностях `Ufw` можно на <https://help.ubuntu.com/community/UFW>.

Настройка брандмауэра

После включения `Ufw` запускается при загрузке, блокируя все входящие соединения и разрешая все исходящие. Это может быть не идеально, если вы хотите подключаться к Pi через `SSH` или `VNC`. Чтобы настроить `Ufw`, используйте терминал своего Pi. `Ufw` распознает большинство служб, таких как `SSH`, и в большинстве случаев вы можете разрешить доступ с помощью `ufw allow`, например:

```
sudo ufw allow ssh
```

Чтобы увидеть полный список служб, включаемых таким образом, просто выполните команду

```
nano /etc/services
```

Если ваш сервис отсутствует в списке, откройте порт, который он использует. Например, `RealVNC`, который может применяться для удаленного подключения к рабочему столу Pi, использует порт 5800. Чтобы `Ufw` открыл его, выполните

```
sudo ufw allow 5800
```

Чтобы разрешить подключения только с определенного IP-адреса, используйте `from`, например:

```
sudo ufw allow from 192.168.1.12
```

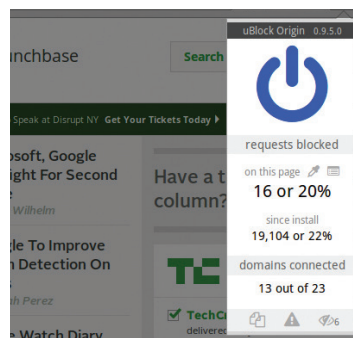
Для просмотра всех параметров брандмауэра в Ubuntu скомандуйте

```
sudo ufw status verbose
```

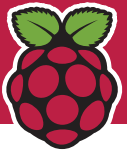
Еще больше упростит вам жизнь графический напарник `Ufw` — `Gufw`. Чтобы открыть его, выполните:

```
sudo gufw
```

`U Gufw` есть удобный переключатель и список открытых и заблокированных портов. Нажмите кнопку +, чтобы добавить дополнительные правила. Нажмите Show extended actions, чтобы увидеть преднастроенные параметры для некоторых популярных приложений, таких как `Skype`.



Удобное дополнение `uBlock Origin` браузера в действии: блокирует всплывающую рекламу.



Создаем брандмауэр с помощью Ufw

```
sudo /home/pi
File Edit Tabs Help
pi@raspberrypi -> sudo apt-get install dnsmasq hostapd
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  dns-root-data dnsmasq-base libn10 libnetfilter-contrack3 libnl-route-3-200
The following NEW packages will be installed:
  dns-root-data dnsmasq dnsmasq-base hostapd libn10 libnetfilter-contrack3
  libnl-route-3-200
0 upgraded, 7 newly installed, 0 to remove and 11 not upgraded.
Need to get 1,015 kB of archives.
After this operation, 2,523 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```

1 Установите базовые программы

Подключите Pi к порту Ethernet вашего маршрутизатора, затем подключитесь через SSH, или, в идеале, откройте терминал на Pi. Установите необходимые программы, выполнив

```
sudo apt-get install dnsmasq hostapd
```

Затем откройте файл конфигурации:

```
sudo nano /etc/dhcpd.conf
```

Добавьте следующие строки в самый конец файла:

```
interface wlan0
static ip_address=172.24.1.1/24
```

Нажмите Ctrl+X, Y, затем Enter, чтобы выйти и сохраниться.

3 Создайте точку доступа

Откройте файл `hostapd.conf` командой

```
sudo nano /etc/hostapd/hostapd.conf
```

и вставьте следующее:

```
interface=wlan0
driver=nl80211
ssid=piVPN
hw_mode=g
channel=1
macaddr_acl=0
auth_algs=1
ignore_broadcast_ssid=0
wpa=2 wpa_key_mgmt=WPA-PSK
wpa_passphrase=raspberrypi231
wpa_pairwise=TKIP
rsn_pairwise=CCMP
```

Измените SSID, пароль и шифрование сети как считаете нужным. Затем запустите

```
sudo nano /etc/default/hostapd
```

Найдите строку, начинающуюся с `DAEMON_CONF=""`. Удалите `#` в начале строки и замените ее на

```
DAEMON_CONF="/etc/hostapd/hostapd.conf"
```

```
pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberr... x pi@raspberr... x
pi@raspberrypi:~$ sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
pi@raspberrypi:~$ sudo iptables -A FORWARD -i eth0 -o wlan0 -m state --state RELATED,ESTABLISHED -j ACCEPT
pi@raspberrypi:~$ sudo iptables -A FORWARD -i wlan0 -o eth0 -j ACCEPT
pi@raspberrypi:~$ sudo sh -c "iptables-save > /etc/iptables.ipv4.nat"
pi@raspberrypi:~$
```

5 Настройте переадресацию IPv4

Откройте терминал на своем Pi и выполните

```
sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
sudo iptables -A FORWARD -i eth0 -o wlan0 -m state --state RELATED,ESTABLISHED -j ACCEPT
sudo iptables -A FORWARD -i wlan0 -o eth0 -j ACCEPT
sudo sh -c "iptables-save > /etc/iptables.ipv4.nat"
```

Затем выполните

```
sudo nano /etc/rc.local
```

Вставьте следующие две строки непосредственно над строкой с `exit 0`:

```
iptables-restore < /etc/iptables.ipv4.nat
/usr/sbin/hostapd /etc/hostapd/hostapd.conf
```

```
sudo /home/pi
File Edit Tabs Help
GNU nano 2.2.6 File: /etc/network/interfaces Modified
iface lo inet loopback
iface eth0 inet manual
allow-hotplug wlan0
iface wlan0 inet static
address 172.24.1.1
netmask 255.255.255.0
network 172.24.1.0
broadcast 172.24.1.255
wpa-conf /etc/wpa_supplicant/wpa_supplicant.conf
```

2 Сделайте себе статический IP-адрес

Откройте свою конфигурацию сетевых интерфейсов:

```
sudo nano /etc/network/interfaces
```

Найдите строку `iface wlan0 inet static` и замените ее на `iface wlan0 inet manual`. Нажмите Enter, чтобы начать новую строку, и вставьте следующее:

```
address 172.24.1.1
netmask 255.255.255.0
network 172.24.1.0
broadcast 172.24.1.255
```

Поставьте `#` в начале строки, начинающейся с `wpa-conf`. Сохранитесь и выйдите, как в прошлый раз.

```
GNU nano 2.2.6 File: /etc/dnsmasq.conf Modified
interface=wlan0
listen-address=172.24.1.1
bind-interfaces
server=8.8.8.8
```

4 Сконфигурируйте dnsmasq

Переименуйте файл конфигурации `dnsmasq`:

```
sudo mv /etc/dnsmasq.conf /etc/dnsmasq.conf.orig
```

Затем создайте новый, командой

```
sudo nano /etc/dnsmasq.conf
```

Вставьте следующий текст:

```
interface=wlan0
listen-address=172.24.1.1
bind-interfaces things elsewhere
server=8.8.8.8
domain-needed
bogus-priv
dhcp-range=172.24.1.50,172.24.1.150,12h
```

Сохраните и выйдите. Затем запустите

```
sudo nano /etc/sysctl.conf
```

Найдите строку, начинающуюся с `"net.ipv4.ip_forward=1"`, и удалите `#` перед этим.

Сохраните, выйдите и перезагрузите Pi.

```
pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberr... x pi@raspberr... x
pi@raspberrypi:~$ sudo ufw status
Status: inactive
pi@raspberrypi:~$ sudo ufw enable
Firewall is active and enabled on system startup
pi@raspberrypi:~$
```

6 Включите брандмауэр

Выполните следующие команды, одну за другой:

```
sudo update-rc.d hostapd enable
sudo update-rc.d dnsmasq enable
```

Перезагрузите Pi.

Чтобы установить, а затем включить брандмауэр, надо выполнить следующие команды:

```
sudo apt-get install ufw gufw
sudo ufw enable
```

На этом этапе вы можете настроить свой брандмауэр — см. врезку «Настройка брандмауэра» на стр. 90.

Итак, отныне ваша сеть надежнее защищена от угроз Интернета.



LXF Hot Picks



Александр Толстой

Shotcut » Makehuman » Rapid Photo Downloader » WINE » Qt5ct » Parlatype » PulseEffects » Doublespeak » Tilix » Stunt Rally » Dead Ascent

Услышав, как кто-то упоминает приложение с открытым кодом, всякий раз многозначительно кивает с понимающим видом.

Видеоредактор

Shotcut

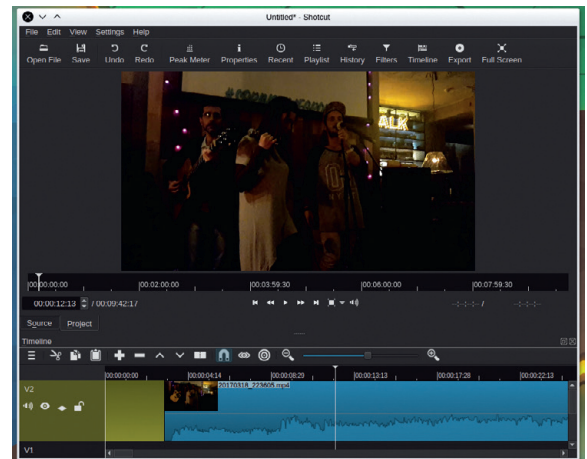
Версия: 17.06 Сайт: <https://shotcut.org>

Исследование мира программ для творчества с открытым кодом очень быстро покажет, что здесь имеется явный спрос на нелинейное видеоредактирование. Независимо от способа применения этого ПО, появляется всё больше подобных программ, и список уже включает такие уважаемые имена, как *Kdenlive*, *OpenShot* и *Flowblade*. Но есть еще одно приложение, которое мы незаслуженно пропустили: *Shotcut*.

В бесконечном ряду программ с названиями с 'shots' и 'cuts', этот новый редактор объединяет простоту с достойным набором функций, и, мы уверены, привлечет преданных пользователей, для которых

другие редакторы слишком сложны либо слишком просты.

Shotcut предлагает опрятный и компактный интерфейс на *Qt5* и использует такие технологии, как *FFmpeg*, *MLT*, *x264* и *LADSPA*, а также разнообразные кодировщики. Список поддерживаемых функций имеет много общего с *Kdenlive*, но мы хотим выделить интересные элементы. Например, встроенное редактирование видео



» *Shotcut* — редактор с открытым кодом с большим количеством функций, идеальный для редактирования домашнего видео.

Обработка в GPU на многих видеочипах работает неплохо.

без необходимости заранее импортировать клипы. В *Shotcut* вы просто открываете один или несколько клипов, как в видеоплеере, и работаете с ними напрямую.

Разработчики приложения обладают собственным видением удобного и интуитивного рабочего пространства. Большая часть функций *Shotcut* доступна из главной панели инструментов, способной и показывать, и скрывать такие панели, как *Filters* [Фильтры], *Timeline* [Временная шкала] или *file properties* [свойства файлов]. По умолчанию *Shotcut* панели не показывает, и после первого запуска надо включить то, что вы собираетесь использовать, например, добавление видео- и аудиотреков и организация панелей по вашему вкусу. Выглядит это проще, чем звучит, но часто говорят, что новичкам *Shotcut* труден в освоении.

По части всего прочего мы попробовали экспериментальную опцию обработки в GPU, из меню *Settings*. Эта функция пока что нестабильна, но на многих современных видеочипах работает очень неплохо. В реальном сценарии редактирования при рендеринге крупных проектов видеускорение GPU может сэкономить вам немало часов и некоторые деньги на оплату электричества, и если это — ваш случай, то определенно стоит дать *Shotcut* шанс.

Проект выпускает промежуточные версии каждый месяц и постоянно улучшается в плане стабильности и переводов.

Исследуем интерфейс Shotcut

Применение фильтров и эффектов

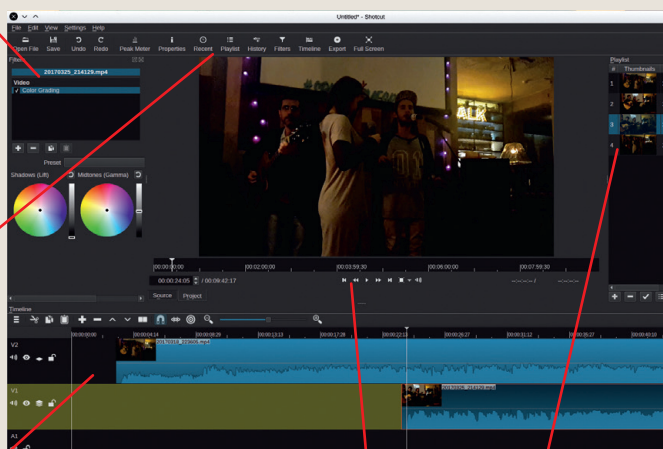
Shotcut подсвечивает файл, к которому будут применяться эффекты и другие исправления.

Всё под рукой

Очень рациональное дизайнерское решение, в котором все наиболее нужные действия всегда доступны.

Временная шкала

Место, где делается реальная работа. Временная шкала *Shotcut* отзывчива и интуитивна.



Управление воспроизведением

Область предпросмотра рендеринга содержит кнопки для перемещения по проекту и воспроизведения отдельных частей.

Плей-лист

Shotcut работает с файлами напрямую, так что нет необходимости сначала их импортировать.

Инструмент 3D-моделирования

Makehuman

Версия: 1.1.1 Сайт: www.makehuman.org

Большинство инструментов с открытым кодом попадают в ограниченное количество категорий, например, плейер, редактор, клиент чего-то, и т.д. Поэтому нас особенно радует возможность взглянуть на оригинальный класс программ — или, по крайней мере, на такой, который не имеет явных аналогов.

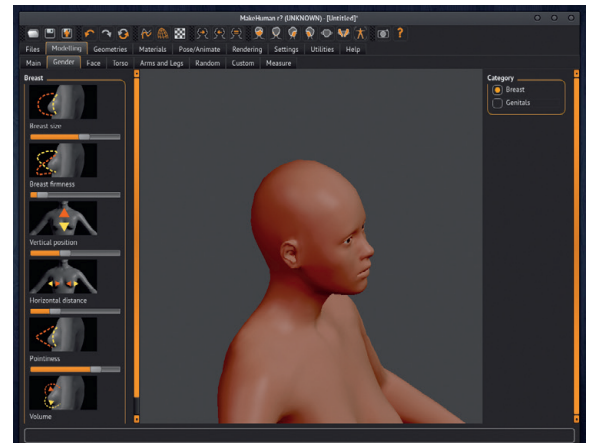
Makehuman — прекрасный инструмент 3D-моделирования, способный создавать изображение человека практически с любыми параметрами, при условии, что он остается прямоходящим существом. Самое лучшее — что у *Makehuman* более низкий входной барьер, чем у других профессиональных инструментов этого вида (вроде *Blender*), и вы можете начать эксперименты с телом прямо сразу.

Вид по умолчанию показывает человеческое тело, которое вы поворачиваете и изменяете его масштаб с помощью мыши. Можно изменить положение камеры, перетаскивая мышью при нажатой правой кнопке, и перейти к любой части модели. Или,

например, увеличить лицо и поиграть с мимикой или прической.

В интерфейсе приложения содержится на удивление большое число функций всего в двух рядах вкладок, с рядом ползунков и переключателей в каждой вкладке. Вы начинаете с ввода пола, возраста, веса и роста, и играете с настройками расы и пропорций. Вкладка *Gender* [Пол] позволяет индивидуально задать мышцы и формы мужского или женского тела (рекомендуется родительский контроль!), и затем можно поднастроить определенные части тела, например, лицо, туловище, ноги, руки и т.д.

Вам предоставляется безграничная свобода в использовании физических данных человеческого тела; это также имеет



› Настроив черты лица, можно добавить брови и волосы.

образовательную ценность для студентов. Помимо поворота обнаженного тела, вы можете потом изменить его позу и одеть его в разные костюмы, платья и т.д.

Makehuman — компактная программа, которая использует Python и возможности OpenGL, и, кроме того, предлагает большой выбор причесок, оттенков кожи и одежды. Короче, отличная игровая площадка для создания анимированных персонажей, изучения анатомии человека, дизайна одежды и многого другого.

Можете начать эксперименты с телом прямо сразу.

Инструмент работы с изображениями

R.P. Downloader

Версия: 0.9.1 Сайт: <http://demonlynch.net/rapid>

Этот очень полезный инструмент для фотографов Linux и всех, кто работает с большим количеством изображений. Поддержание порядка в библиотеке изображений может показаться не слишком трудной задачей, но когда у вас появляется несколько тысяч файлов, это меняет дело.

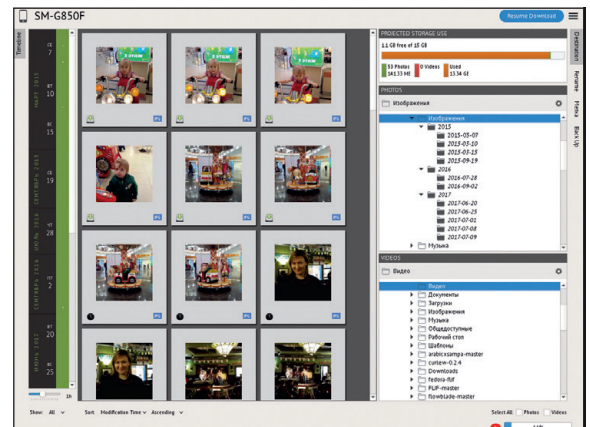
Rapid Photo Downloader (RPD) предназначен для того, чтобы упростить скачивание и сортировку файлов с вашей цифровой камеры или смартфона. Коротко говоря, он автоматически сканирует ваши удаленные устройства USB и быстро создает эскизы предпросмотра для графических и видеофайлов. Затем он позволяет скачивать файлы и организовать их в папки по дате.

На первый взгляд, ничего особенного, ведь у специальных фотоменеджерах типа *Digikam* или *Shotwell* функции такие же; но *RPD* всё же имеет свою нишу. Преимущество номер 1 — в том, что

он обеспечивает вас невероятно быстрым и при этом мощным решением для фотографий на настольном Linux, и с ним вы сможете спокойно избавиться от тяжелых фотоменеджеров и исследовать свою коллекцию в любимом менеджере файлов.

Верхний левый угол окна приложения показывает переключатели для автоматического сканирования внешних носителей, а также вашу локальную файловую систему. Правая панель отображает информацию о свободном месте на выбранном устройстве и показывает раскладку предлагаемой директории, где после импорта будут храниться ваши изображения и видеофайлы. Это отличный холостой прогон,

Невероятно быстрое и мощное решение для фотографий.



› Настройте его один раз, и вы сможете автоматически импортировать ваши фото с устройств USB.

который позволяет изменить настройки и имена, прежде чем взяться что-нибудь скачивать.

Но, поделившись этими преимуществами, мы всё же думаем, что *RPD* еще есть что предложить. Перейдите в меню *Preferences* (со значком-гамбургером) и затем в раздел *Automate*. Здесь вы можете дать *RPD* задание скачать ваши фото и видео после того, как вы подключите свое устройство, размонтировать или извлечь носитель по завершении процесса скачивания, и т.д.

Чем больше вы работаете с *RPD*, тем больше вам открывается преимуществ.

Уровень приложений Windows

WINE

Версия: 2.12 Сайт: www.winehq.org

WINE — отличная программа запуска исполняемых файлов Windows в Linux, и она существует уже почти 20 лет. Это одно из самых успешных и высоко ценимых приложений с открытым кодом, которое развивается весьма быстро, так что это далеко не первый раз, когда WINE попадает в обзоры Hotpicks.

В наши дни мира Windows и Linux сближаются, особенно с тех пор, как подсистема Linux стала доступна в Windows 10; но хотя мы предпочитаем запускать приложения Windows в Linux (а не наоборот), нам всё же требуется стабильная поддержка для API Win32 в Linux, что и реализуется в WINE.

Удивительно, что даже первые релизы WINE в начале 2000-х гг. были способны запускать такие сложные программы, как *Adobe Photoshop* или *Microsoft Office*, и с течением времени поддержка разных приложений Win32 крепла. Современный релиз WINE 2.12 делает заметный упор

на улучшение поддержки Android, что очень радует.

Использование WINE с Android означает, что вы можете заставить тысячи приложений Windows работать на устройствах с отличающейся от x86 архитектурой; это не просто приводит сюда смартфоны и планшеты, основанные на ARM, но и обеспечивает, чтобы для людей с Android на ноутбуках и планшетах с внешней клавиатурой их работа превратилась в более дружелюбную к настольным ПК.

Стоит также отметить свободный проект Android-x86, ОС, которая, вероятно, будет очень гладко работать на вашем ПК. С WINE она может стать заменителем Linux в определенных задачах.

Обновление делает упор на улучшение поддержки Android.



► В WINE улучшена поддержка Android, что позволяет запускать ваши любимые приложения на настольном ПК.

Еще один интересный момент — расширенная поддержка современных коммерческих игр. С каждым релизом WINE (приблизительно раз в три недели) все могут видеть закрытие ошибок по части поддержки Direct3D и улучшение взаимодействия с OpenGL в играх Windows, которые работают через WINE в Linux. Не забудьте познакомиться с утилитой *winecfg*: в ней имеется много полезных настроек, которые могут вас заинтересовать.

Проекты WINE к тому же продвигают сборки «этапов wine» с дополнительными вкладками и функциями, скрытыми в *winecfg*, хотя они и экспериментальные.

Инструмент настройки

Qt5ct

Версия: 0.33 Сайт: <https://github.com/quiroy9/qt5ct>

Иногда, когда мы ищем приложение, обязанное выполнять наши задачи, у нас налицо выбор из нескольких, и мы частенько предпочитаем то, которое лучше всего подходит к нашему рабочему столу.

Иными словами, пользователи Gnome 3, Unity и Pantheon выбирают приложения GTK3, а те, кто предпочитает KDE Plasma 5, заполняют свои системы приложениями на Qt5. Все перечисленные среды рабочего стола имеют свои средства интеграции приложений, написанные с помощью других библиотек или сред, поэтому большинство тех, кто работает на Ubuntu Unity, не замечают, основаны ли их любимые приложения на GTK или Qt, по большей части благодаря точной стилизации тем обоих.

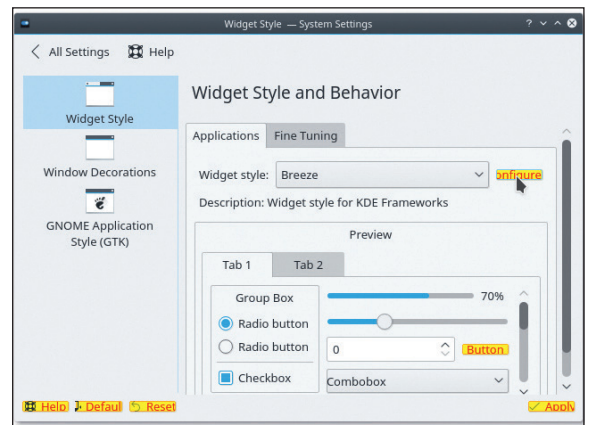
Но мы забыли о других рабочих столах, например, таких как Xfce, или минималистских вручную настроенных сессиях на OpenBox, или о различных мозаичных менеджерах окон, у которых отсутствует интеграция тем Qt.

Qt5ct — небольшая, но очень удобная утилита, которая заполняет этот пробел и предлагает инструмент стилизации и выбора тем для всех, независимо от рабочего стола. Qt5ct может даже заменить собственные настройки Plasma, когда речь идет о внешнем виде и работе элементов интерфейса, таких как кнопки, панели инструментов и окна-флажки. Это делается посредством настройки специальной переменной Qt:

```
$ export QT_OPA_PLATFORMTHEME=qt5ct
```

Затем можно запустить исполняемый файл *qt5ct* и начать настраивать свой стиль. В приложении имеется пять вкладок для изменения внешнего вида, шрифтов, значков, тем, эффектов интерфейса, а также применения персональных таблиц стилей.

Задаваемые правила отменяют настройки по умолчанию.



► С помощью Qt5ct мы сделали кнопки более тонкими и цветными.

Последней вкладке стоит уделить несколько больше слов, поскольку она мощнее, чем можно ожидать. Qt может изменять элементы с помощью задаваемых правил, которые отменяют настройки по умолчанию выбранного стиля. Скажем, не нравятся вам широкие поля кнопок в стиле Breeze — тогда всего лишь надо создать свой файл *.qss* и добавить нечто вроде

```
QPushButton {max-height: 10px;}
```

Возможность повсеместного применения индивидуальных таблиц стилей является уникальной функцией Qt5ct, которая порадует тех, кто любит навести красоту на рабочем столе.

Аудиоплеер

Parlatype

Версия: 1.5.1 Сайт: github.com/gkarsay/parlatype

Недavno мы наткнулись на выдающееся приложение в мире музыкальных и аудиоплееров. *Parlatype* — небольшая программа, и вид его компактных окон развеивает любую надежду на то, что этот плеер скрывает дополнения вроде медиа-библиотек или интеграцию с радиосервисами Интернета.

Вместо всего этого *Parlatype* предлагает удобный способ прослушивать интервью, лекции или иные образчики записанной речи. Будучи приложением *GTK3*, *Parlatype* является отличным дополнением к рабочему столу *Gnome*, но вы, естественно, можете использовать его любым способом по своему усмотрению.

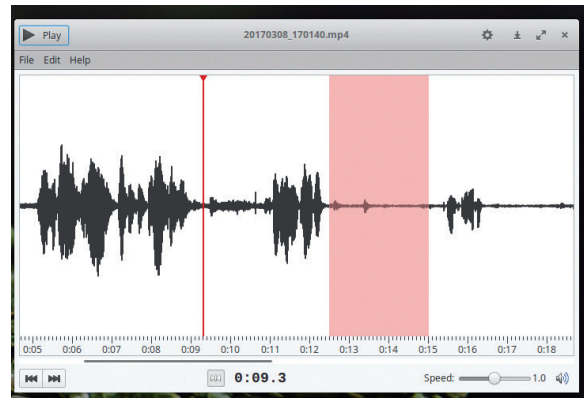
При ближайшем рассмотрении *Parlatype* обнаруживает функции, которые его отличают. То, что все заметят вскоре после загрузки аудиофайла в *Parlatype* — видимая форма волны. Такие отображения волны весьма характерны для аудиоредакторов, таких, как *Eko*, *Kwave* и *Audacity*,

но в *Parlatype* оно играет другую роль. Вам не надо редактировать звук, но приятно узнать, какие части вашей записи содержат паузы и тишину, благодаря чему легко их прогнать.

Две кнопки в нижнем правом углу позволяют перепрыгивать вперед и назад на десять секунд (по умолчанию), а ползунок рядом с ними изменяет скорость воспроизведения без изменения тона.

Окно Preferences в *Parlatype* не может похвастаться большим количеством настроек, однако любая из них полезна и практична. Например, можно зафиксировать статичный курсор (чтобы волна проходила через него); установить, сколько секунд пропускать при переходе; настроить

Удобный способ прослушивать интервью, лекции и т. д.



► Аудиоплеер *Parlatype* чуть упростил преобразование монотонной речи в текст.

масштаб волны или скомандовать приложению всегда запускаться поверх всего (это потребует перезапуска).

Очевидно, что *Parlatype* был написан тем, кому надо было лично преобразовать аудиофайл в текст, и если вы периодически делаете то же самое, дайте *Parlatype* шанс.

Вы можете открывать в *Parlatype* также и фильмы, если они основаны на *Gstreamer*. Приложение извлечет звук и покажет видео в отдельном окне, которое спокойно можно минимизировать.

Эквалайзер

PulseEffects

Версия: 2.0.7 Сайт: github.com/wwmm/pulseeffects

Большинство аудиоплееров в Linux снабжены программными эквалайзерами — хорошая функция, позволяющая изменять определенные частоты и настраивать звук по своему усмотрению. Однако на свете немало людей, не знакомых с *PulseEffects*, прекрасным отдельным эквалайзером и обязательным инструментом для всех, кто любит слушать, записывать и сводить звук в Linux.

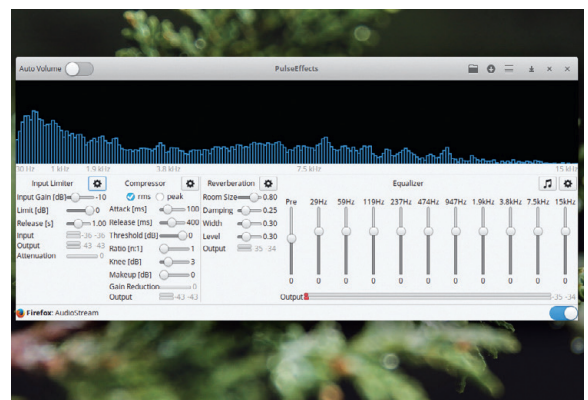
PulseEffects дополняет *PulseAudio*, звуковую систему по умолчанию практически всех современных дистрибутивов Linux. В приложении имеется невероятное количество настроек в одном окне, и если вы не профессионал в мастеринге записи, вам понадобится некоторое время на привыкание к ним.

PulseEffects — не только эквалайзер, но также ограничитель, инструмент реверберации и компрессор. Мы не можем гарантировать, что улучшение звука будет простой задачей, но с таким изобилием опций, пресетов, ограничителей, усилителей

и прочих слайдеров и переключателей, мы уверены, что он более чем способен справиться с этим.

При запуске *PulseEffects* его дисплей включает приложения, которые воспроизводят звук прямо сейчас (если таковых нет, *PulseEffects* останется спокойным). С помощью переключателей можно выбрать, хотите ли вы применить эффекты ко всему или только к определенному приложению. Фактически, этот подход очень удобен для многорожечной записи в Linux. С *PulseEffects* вы можете применить реверберацию или ограничить ввод для определенной части, оставив остальные нетронутыми. Также можно сохранить свои индивидуальные настройки в виде пресетов и загрузить их в любое время позднее.

Не только эквалайзер, но ограничитель, ревербер и компрессор.



► Полнофункциональный аудиозквалайзер *PulseEffects* отлично настроит ваши домашние записи.

В новый *PulseEffects* встроено 18 пресетов, а также сделаны некоторые отладки, например, унификация громкости. Хотя приложения в первую очередь предназначены для аудиофилов, простые смертные тоже найдут удовольствие в настройке своей звуковой системы и лучшего качества звука в своих динамиках.

Под капотом *PulseEffects* являет собой набор независимых от дистрибутива скриптов Python, но Ubuntu может легко скачать это приложение из [ppa:yunnxx/gnome3](https://ppa.yunnxx/gnome3).

HotGames Развлекательные приложения

Гонки

Stunt Rally

Версия: 2.6 Сайт: <http://stuntrally.tuxfamily.org>

Мы делаем обзоры тяжелых весных игр Linux в *HotPicks* лишь изредка — отчасти потому, что не так уж много есть игр с открытым кодом, способных втиснуть все ресурсы в ваш компьютер, в противоположность тому, что мы видим в мире коммерческих игр Linux.

Вот почему мы не могли скрыть удовольствие при виде *Stunt Rally*, превосходного кросс-платформенного гоночного симулятора, с полностью открытым кодом. Он, конечно, требует наличия хорошего GPU — как минимум, современного GPU от Intel, чтобы играть на средних настройках качества, и специальной видеокарты, чтобы наслаждаться детализованной графикой с отражениями и разумной частотой кадров.

После первого запуска *Stunt Rally* показывает полезнейший экран приглашения,

который объясняет основные привязки к клавишам и предлагает вам начать обучение в режиме руководства и попробовать сначала пройти четыре базовых трека. Это весьма разумный вариант, чтобы привыкнуть к физике игры и исследовать обычный ландшафт *Stunt Rally*, где полно камней и деревьев, и дороги по большей части гравийные.

Вы можете использовать клавиши Q и W, чтобы разворачивать машину влево и вправо, и перематывать свои трюки для повтора с помощью клавиши Backspace. В игре имеется 172 трека в 34 разных ландшафтах, а также масса расширенных



► В *Stunt Rally* вы можете перевернуть машину и постараться не задеть многочисленные препятствия на пути.

функций, уместных в профессиональных играх — таких как чемпионат, воспроизведение, призраки гонки, конкурсы на самых сложных треках, многопользовательский режим и много чего еще.

Если у вас мощный GPU, можете включить полноэкранный режим и играть с другом на том же компьютере. Плюс к этому, *Stunt Rally* позволяет переопределять настройки, изменять поведение камеры и проектировать свои собственные треки.

Если у вас хорошее оборудование и внешние средства управления игрой, то *Stunt Rally* будет вне конкуренции среди гонок с открытым кодом для Linux.

Вне конкуренции среди гонок с открытым кодом для Linux.

Приключения

Dead Ascent

Версия: 1.0.2 Сайт: <https://github.com/Larpon/>

Прошло уже немало времени с тех пор, как мы были в почти безвыходной ситуации: орды зомби загнали нашего героя в подвал, и теперь надо найти выход. Каждый уровень игры — это этаж башни, где надо решать загадки и выстраивать разнообразные конструкции из импровизированных объектов.

Dead Ascent — это классическая игра с управлением мышью, графикой, нарисованной вручную, и встроенным справочником. При каждом нажатии на объект появляется страница с объяснением, и если объект имеет хоть какую-то пользу, он проследует в ваш рюкзак. Таким образом вы можете собирать объекты и носить их с одного уровня на другой. Объекты могут взаимодействовать друг с другом — например, можно использовать жвачку, чтобы залепить дыру в ведре, или построить лестницу из палочек.

По мере восхождения задачи становятся интереснее и сложнее. Вы должны найти способ запустить генератор, очистив его от ненужных предметов и найдя источник энергии. Вряд ли будет большим спойлером, если мы скажем, что вам, вероятно, понадобятся шприц и хомяк, чтобы запустить лифт и подняться на этаж выше. Всплывающие подсказки очень полезны, поскольку они мягко направляют вас к нужному решению и дают подсказки и советы.

Игра мотивирует вас дойти до конца благодаря прекрасной графике, музыке и продуманному сюжету. Мы умудрились решить все задачи за час. Кстати, для *Dead*



► Этот зомби безопасный и к тому же полезный, потому что скоро вам потребуется его зараженная кровь.

Ascent нет никаких инструкций или читов, но если вы застрянете, можете поискать по ее QML файлам и посмотреть код, чтобы найти решение. Что за прекрасный способ изучения кодирования!

Хоть игра и про зомби, на пути к последнему этажу вам их попадает не так уж много. *Dead Ascent* — небольшая, но очень качественная и захватывающая игра для Linux. У нее также есть сборки для Android и iOS.

Вам понадобятся шприц и хомяк, чтобы запустить лифт.

Инструмент стеганографии

Doublespeak

Версия: GIT Сайт: <https://github.com/joshuaptfan/>

Если вы смотрели на YouTube видео Quirkology, то, вероятно, уже знаете, как выполнять некоторые крутые трюки, которые кажутся настоящей магией.

Можно натренировать гибкость пальцев, выяснить, как работают некоторые оптические иллюзии, и узнать, как отвлечь внимание зрителя от того, что вы хотите скрыть. А как насчет магии открытого кода?

Посмотрите, я могу отражением превратить R в Я, или N в И! Это легко, если большая часть текста, который вы читаете, в 8-битных символах Unicode или UTF-8, которые также содержат символы кириллицы. Есть еще одна функция в Unicode, которая позволяет создать *Doublespeak*, героя этого обзора. Это — непечатаемые символы, в частности, дополнительный пробел, соединитель, не-соединитель и дополнительный неразрывный пробел. Каждый из четырех имеет нулевую ширину, и это означает, что вы не можете его видеть. *Doublespeak* — простая web-страница демо, созданная с помощью HTML, CSS и Javascript,

где можно добавлять секретные текстовые сообщения в любой видимый текст и затем использовать их как форму стеганографии.

На странице есть несколько полей ввода, помогающие вам закодировать ваш текст с секретным сообщением и затем скопировать его в буфер, и еще один блок для дешифровки ранее закодированного текста. Это и правда выглядит, как магия, потому что секретные сообщения сохраняются даже при копировании и вставке между почти любыми сторонними приложениями и в разных операционных системах.

Единственная заметная улика — это что зашифрованный текст занимает больше байт. *Doublespeak* использует для кодирования данных числовую систему по основанию 4, известную как четверичная, поэтому

Добавьте секретный текст к видимому, как стеганографию.



➤ Невидимого мы тут показать не можем, но вот вам вместо этого страница *Doublespeak*!

секретному тексту понадобится вчетверо больше байт, чем такому же тексту в обычном UTF-8. В результате вы можете заметить, что ваше сообщение, сохраненное как текстовый файл, имеет подозрительно большой размер.

Однако шансы на то, что кто-то вдруг решит проверить каждый фрагмент текста из, скажем, окна чата на предмет наличия скрытых слов, крайне малы. Используйте его, чтобы загадать задачу своим друзьям — или каким-то другим способом на благо. *Doublespeak* поможет вам делать крутые фокусы, так что его стоит добавить в закладки.

Эмулятор терминала

Tilix

Версия: 1.5.8 Сайт: <https://github.com/gnunn1/tilix>

Tilix — мозаичный эмулятор терминала, ранее известный как *Terminix*. Он популярен среди пользователей Ubuntu и не только как простой, но мощный инструмент с функцией мозаичной раскладки. Не секрет, что мозаика — популярный метод организации рабочего пространства, и у нас уже есть ряд мозаичных менеджеров окон и прочих программ, например, мультитерминалов терминала.

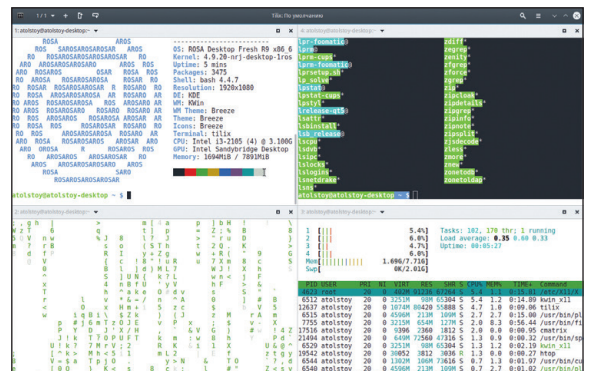
С *Tilix* вам не надо изменять всю сессию пользователя и использовать мозаику по всему рабочему столу. Вместо этого всё происходит внутри окна *Tilix* — это отлично проработанное приложение *GTK3*. Мы признаем, что *Tilix* очень дружелюбное к пользователю приложение, если учесть, что оно разработано для тех, кто живет в основном в интерфейсе командной строки.

Tilix позволяет начать работать, не требуя запоминать специальные комбинации клавиш. На заголовке имеются кнопки для управления терминалами и сессиями.

Не может быть ничего проще двух специальных значков, которые добавляют дополнительные терминалы справа или ниже текущего. Вы закрываете терминал, нажав на крестик, как вкладки в браузере; перетаскиваете любое окно терминала из приложения, чтобы работать с ним независимо; или перетаскиваете в другой терминал для слияния. Для тех, кому нужно больше терминалов, с левой стороны окна есть менеджер сессии. Переключатель сессии похож на переключатель во многих обычных средах рабочего стола и позволяет увеличивать число терминалов до очень большого количества.

И это еще не всё, потому что у *Tilix* есть еще один уровень абстракции. Перейдя

Для тех, кому нужно больше терминалов — менеджер сессии.



➤ *Tilix* приглашает вас использовать командную строку весьма изящным способом.

в предпочтения *Tilix*, вы сможете менять разные настройки терминала: цвета, звуки, режим мигания, работу прокрутки и т. д., и всё это вместе формирует ваш профиль. Можно сохранить сколько угодно профилей и применять разные профили к разным окнам терминала, так что в теории с помощью *Tilix* можно даже воссоздать жуткую голливудскую технодраму, о которой мы рассказывали в **LXF195**.

И, наконец, все настройки *Tilix*, в т. ч. настройки для окон, можно сохранить как файлы JSON и загрузить позднее. Почти не сосчитать комбинации настроек терминала, возможные в *Tilix*. Это как Lego для создания исключительной рабочей области терминала из отличных кирпичей. **LXF**

На диске

Дистрибутивы, приложения, игры, книги и всякое-разное...

Лучшее из Интернета, упакованное в 8 ГБ качественного DVD.



Дистрибутивы

Все три дистрибутива на DVD этого месяца имеют общего предка — Red Hat Linux, хотя они и предлагают разное взаимодействие с пользователем. О проектах-ответвлениях мы слышим нередко; иногда их делают недовольные разработчики, полагающие, что они лучше знают (а бывает и так — вспомните, что случилось, когда X.Org отделился от XFree86). Одна из жалоб заключается в том, что ответвления приводят к фрагментации, и на краткий срок, возможно, это верно. Но в результате, как правило, процветает один — или ответвление, или оригинал — обычно благодаря лучшим коду и поддержке; а другой увядает. Создание ответвлений — лишь форма эволюционного процесса, ПО-эквивалент мутации; но у проектов есть не только ответвления. Два дистрибутива на DVD этого месяца — результат некогда произошедшего слияния. LXQt появился, когда разработчики двух легковесных рабочих столов поняли, что им будет лучше работать вместе, а Mageia хоть и является ответвлением, но это ответвление массивного слияния двух крупных дистрибутивов.

Эти ответвления и слияния — лишь один пример знаменитого «базара» Эрика С. Реймонда в действии; пусть же они продолжатся.

Neil

» Важно ВНИМАНИЕ!

Порченые диски

В маловероятном случае какого-то дефекта вашего LXF DVD обращайтесь, пожалуйста, по адресу disks@linuxformat.ru или телефону +7 (812) 309-0686.

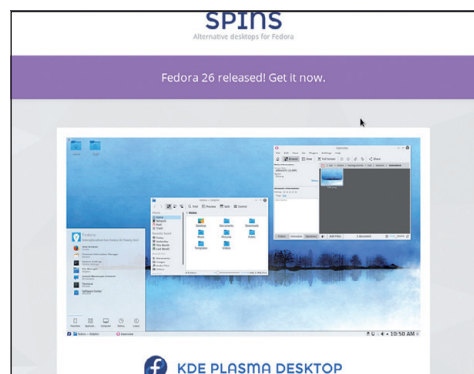
Самый свежий релиз

Fedora 26

Fedora был создан в Red Hat, но в виде дистрибутива, поддерживаемого сообществом, и задумывался не как бесплатная версия основного продукта Red Hat — если вам нужен Red Hat Enterprise Linux без договора поддержки, вы можете установить CentOS. Вместо этого Fedora служит своего рода тестовым полигоном для технологий, которые Red Hat хочет включить в RHEL, и в этом качестве обрел репутацию самого технологически передового дистрибутива; масса блестящих функций появляется здесь намного раньше, чем в других дистрибутивах — в полную противоположность консервативному и стабильному Debian с диска прошлого месяца.

Fedora был первым дистрибутивом, который принял *Pulse Audio* и *systemd*. Что и неудивительно, ведь они были разработаны людьми из Red Hat. Однако он также экспериментировал с *Wayland* раньше всех остальных, и это по-прежнему единственный дистрибутив, который приходит на ум, где по умолчанию для подготовки диска во время установки применяется LVM.

Эта live-версия Fedora использует рабочий стол Gnome, который давно уже был выбран рабочим столом для Red Hat и Fedora, хотя после установки вы можете добавить любой рабочий стол, который хотите.



Желая попробовать нечто полегче, чем Gnome, попробуйте Fedora LXQt, описанный на соседней странице. Установка должна пройти быстро и не больно.

Давайте признаем, они занимались этим долгие годы, и установщик *Anaconda* отлично отлажен; и то же описание применимо и ко всему дистрибутиву. Если когда-либо какой-то дистрибутив и противостоял Ubuntu и его птенцам на пути к мировому господству среди домашних пользователей, то это Fedora. И пусть ответвлений Ubuntu не считать — в частности, из-за того, что их программа очень упрощает создание таковых; у Fedora тоже немало веток. Если вы сочтете, что работа со стандартным Fedora вам не особо подходит, загляните на <https://spins.fedoraproject.org>, где вы найдете ряд ссылок на скачивание live-дисков Fedora с разнообразными рабочими столами.

Логин: **liveuser**, пароль: пустой.



» На соревнованиях за простоту в работе Fedora явно заставит Ubuntu попрыгать.



Новичок в Linux? Начните отсюда!

- » Что такое Linux? Как его установить?
- » Есть ли в нем эквивалент *MS Office*?
- » Зачем нужна командная строка?
- » Как устанавливать программы?

Ответы приводятся в [Index.html](#) на диске.

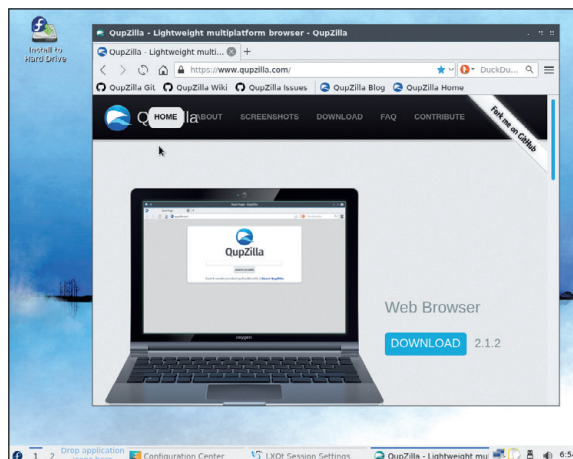
Дистрибутив на LXDE/Razor-Qt

Fedora LXQt 26

Это одно из ранее упомянутых ответвлений Fedora, но с рабочим столом, который вы, возможно, еще не пробовали. *LXQt* — это союз *LXDE*, легковесного рабочего стола на *GTK*, с которым вы, вероятно, не встречались, и *Razor-Qt*, столь же легковесного, но куда менее зрелого рабочего стола, использующего инструментарий *Qt*. Появившийся в результате дистрибутив предлагает пользователям лучшее от обоих: зрелый легковесный рабочий стол от *LXDE* с привлекательным и настраиваемым интерфейсом на *Qt*. Хотя многие люди считают

легковесные рабочие столы идеальными для старого оборудования — и это, безусловно, правда — им есть место и на более современных системах. Зачем тратить драгоценные производительные ресурсы и память на лишние функции, если можно наслаждаться куда более быстрым функционированием рабочего стола, который продолжает делать всё, что вам надо? И это сущий подарок, если вы применяете свой компьютер для задач с большей нагрузкой на CPU, например, для обработки видео.

Логин: *liveuser*, пароль: пустой.



» Охват *Qt* настолько широк, что включает даже браузер *QupZilla*.

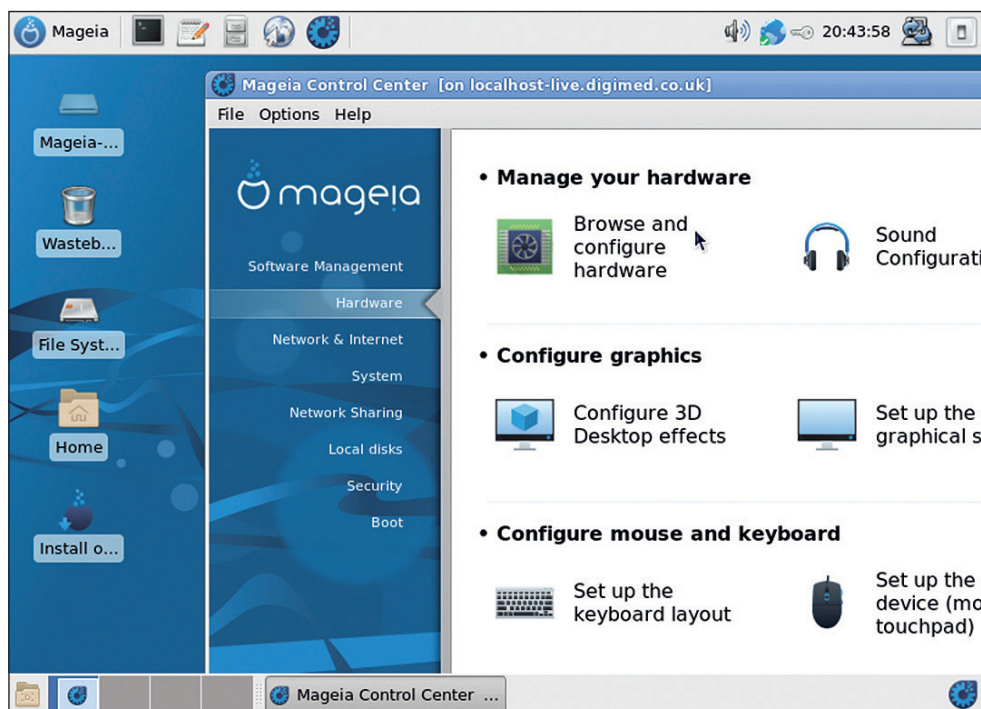
Ответвление Mandriva

Mageia XFCE 6.0

Mageia — дистрибутив с обширным и развесистым генеалогическим деревом. Это ветвь Mandriva, который был хорошим дистрибутивом с прискорбной привычкой метаться от одного финансового кризиса к другому. Сам Mandriva был создан объединением Mandrake и Connectiva, и многие долгосрочные пользователи Linux, включая нас самих, лелеют добрые воспоминания о Mandrake — это был Ubuntu своего времени, и начинался он как ответвление Red Hat, но с рабочим столом KDE.

Отложим уроки истории в сторонку: на DVD этого месяца — самая свежая версия Mageia, чтобы вы познакомились с ней и, если она вам понравится, установили. Если будете ее устанавливать, не забудьте, что программа установки попросит вас ввести пароль; это касается загрузчика, а не пользователя или *root*. Обычно его можно оставить пустым. Установка пароля *root* и создание пользователя происходит только при первой загрузке в установленную систему, а запустившийся мастер позаботится о деталях финальной настройки.

Логин: *live*, пароль: пустой.



» Mageia происходит из приличной семьи незаурядных дистрибутивов.

Пропустили номер?



Закажите его на сайте 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!

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

LXF DVD: ClearOS, Debian, Koozali, NethServer, Voyager, Zentyal, 12 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_227/
PDF-версия: shop.linuxformat.ru/elxf_227/

LXF226 Август 2017



- » **Уходим на Linux** Выбираем свободу
- » **Раскидываем сети** Ну-ка, проследи!
- » **Анимация** Почувствуйте себя Диснеем
- » **Меньше ада** Doom на шоколадном движке

LXF DVD: BackBox Linux, Elementary OS, Manjaro, Tails, Zorin, горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_226/
PDF-версия: shop.linuxformat.ru/elxf_226/

LXF225 Июль 2017



- » **Притворимся хакерами** Но не корысти ради
- » **Мультки и не только** Программы анимации
- » **Параллельный Интернет** Сеть для избранных
- » **Лучше перебдеть** Двухфакторная аутентификация

LXF DVD: Ubuntu, Solus, Android-x86, 12 книг о Linux (на английском), горячие новинки, архив LXF за 2005–2016 гг. и прочее...

Печатная версия: shop.linuxformat.ru/lxf_225/
PDF-версия: shop.linuxformat.ru/elxf_225/

LXF224 Июнь 2017



- » **Конец Unity** Ubuntu перешел на Gnome
- » **Где KDE краше** Лучший дистрибутив с KDE
- » **Боронися, бабка** Брандмауэр + роутер = защита
- » **Кому верить?** Надежные ключи шифрования

LXF DVD: Ubuntu, Linux Lite, Manjaro, Netrunner, 12 книг о Linux (на английском), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_224/
PDF-версия: shop.linuxformat.ru/elxf_224/

LXF223 Май 2017



- » **Ставим на Zero** Raspberry Pi Zero W уже с Wi-Fi и Bluetooth
- » **Дизайн для 3D-принтера** Создание моделей
- » **В стиле Blu-ray** Заставим работать в Linux
- » **Стеганография** И чтоб никто не догадался

LXF DVD: feren OS, IPFire, openSUSE Tumbleweed, RancherOS, Rescatux, Ubuntu Studio, XenialDog, 12 книг о Linux (на английском), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_223/
PDF-версия: shop.linuxformat.ru/elxf_223/

LXF222 Апрель 2017



- » **Стать Linux-джедаем** Изучаем терминал
- » **В Интернет — невидимкой** Дистрибутивы конфиденциальности
- » **Серверы на все лады** Два учебника и статья
- » **Сайт с галереей** Адаптивный web-дизайн

LXF DVD: Devuan, Linux Mint, ROSA, Scientific Linux, Ubuntu Budgie, 12 книг о Linux (на английском), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_222/
PDF-версия: shop.linuxformat.ru/elxf_222/

Подпишитесь на печатную версию журнала на www.linuxformat.ru/subscribe
или 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

» Содержание

Linux
Format
FOUNDED 2005

Страница 1

ДИСТРИБУТИВЫ

Fedora 26 Gnome (64-битный)
Fedora 26 LXDE (32-битный)
Дистрибутив Linux, разработываемый сообществом при поддержке проекта Fedora. Имеет репутацию дистрибутива с упором на инновации, интеграцию новых технологий на ранней стадии и тесное сотрудничество с другими сообществами Linux.
Mageia 6.0 (32-битный)
Дистрибутив с разветвленным генеалогическим древом — ответвление сообщества от популярного французского дистрибутива Mandriva Linux. Разрабатывается с упором на дружелюбие к начинающим.

Страница 2

ДИСТРИБУТИВЫ

SAINE (64-битный)
SAINE (Смарт-Айдед Investigative Environment) — дистрибутив GNU/Linux на базе Ubuntu, созданный как проект цифровой криминалистики. Предлагает полноценную среду с поддержкой чехарк, этапов цифрового расследования, организованную для интеграции существующих программных средств в виде программных модулей, с дружественным графическим интерфейсом. Обеспечивает полную оперативную компиляцию ит.о.в.о.т.ч.а.

СРАВНЕНИЕ: СЕРВЕРЫ СИНХРОНИЗАЦИИ ФАЙЛОВ

Nextcloud 12.0
www.nextcloud.com
Pydio 8.0
www.pydio.com
Seafile 6.11
www.seafile.com
SparkShare 1.5
www.sparkshare.org
Syncthing 0.14.31
www.syncthing.net

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

Внимательно прочтите это перед тем, как использовать LXF DVD!

Fedora 26

НОВЫЙ РЕЛИЗ



32- и 64-битный

Передовой
дистрибутив
с новейшими
технологиями:

- » Wayland
- » Ядро 4.11
- » Gnome 3.24
- » GCC 7
- » Python 3.6 + еще куча всего!

НОВЫЙ РЕЛИЗ

Mageia 6.0

32-битный

После двух лет разработки — солидное обновление короля настольного дружелюбия!



Linux
Format
FOUNDED 2005

LIVE-ДИСК: ГОТОВ К РАБОТЕ
3 ПОЛНЫХ ДИСТРИБУТИВА ДЛЯ СТАРТА В LINUX

Содержание



НОТРИСКИ

- Dead Ascent 1.0.2** Игра с приключениями dl.dropbox.com/u/14900/
- Doublespeak** (Gty) Инструмент стеганографии dl.dropbox.com/u/10514914/
- Makehuman 1.11** Инструмент 3D-моделирования www.makelinux.org
- Paratype 1.51** Аудиоплеер dl.dropbox.com/u/146349/
- PulseEffects 2.0.7** Эмулятор dl.dropbox.com/u/10514914/
- Qibet 0.33** Инструмент настройки с изображениями dl.dropbox.com/u/10514914/
- Rapid Photo Downloader 0.9.1** Инструмент работы с изображениями demonstrations.net/rapid
- Shotcut 17.06** Видеоредактор shotcut.org
- Stunt Rally 2.6** Игра-гонки stuntrally.linuxfamily.org

- Tiik 1.5.8** Эмулятор терминала dl.dropbox.com/u/10514914/
- Wine 2.12** Уровень приложений Windows www.winehq.org

ПОМОЩЬ

- Руководство новичка
- Руководства
- Ответы
- ЧаВо (FAQ)

ПЛАВНОЕ

- CheckInstall
- Coreutils
- HardInfo
- Kernel
- Memtest86+
- Plop
- SBM
- WvDial

ДОКУМЕНТАЦИЯ. 12 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 Реймонда [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 специфическую терминологию

- Linux Kernel in a Nutshell** Описание ядра Linux, созданное одним из его выдающихся разработчиков — Грегом Кроа-Хартманом [Greg Kroah-Hartman]
- Route** Знаменитый справочник System Administrators Guide Руководство по базовому администрированию Linux
- GNU Tools Summary** Руководство по работе в командной строке и обзор основных утилит GNU

Все дистрибутивы представлены ISO-образами, который можно записать на отдельный носитель, и загрузить в live-режиме прямо с LXF DVD. У всех приложений возможность установки на жесткий диск.

Пожалуйста, перед использованием Аджунто Диска ознакомьтесь с инструкцией, опубликованной в журнале на стр. 109!

КОММЕНТАРИЙ Присылайте ваши пожелания и предложения по электронной почте: info@linuxformat.ru

ДЕЯТЕЛЬНЫЕ ДИСКИ В маловероятном случае обнаружения дефектов на дисках, обращайтесь, пожалуйста, по адресу disk@linuxformat.ru

Настоящий диск тщательно тестировался и проверялся на всех старых платформах, однако, как и в случае с любым новым ПО, мы рекомендуем вам использовать аэрированный 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, предоставленных на нем программ или данных. Прежде чем устанавливать какие-либо ПО на компьютер, пожалуйста, с сайта www.linuxformat.ru скачайте все файлы, необходимые для установки.

Тираж изготовлен ООО «Марком», 186852, Россия, Ленинградская область, Всеволожский р-н, дер. Юрки, Школьная ул., 7-а. Лицензия ИМПР ВАР N 77-03.

Создание установочных дисков при помощи cdcrecord

Самый быстрый способ записать ISO-образ на чистую матрицу — это обратиться к программе *cdcrecord*. Для всех перечисленных ниже действий потребуются права суперпользователя-*root*. Сначала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdcrecord -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:

```
cdcrecord dev=0,3,0 -v /путь к образу/image.iso
```

Чтобы упростить дальнейшее использование *cdcrecord*, сохраните некоторые настройки в файле */etc/default/cdcrecord*. 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

```
cdcrecord -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 *cdcrecord* 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.

ИТ и СПО в образовании

Александр Недбайлов оценивает применение компьютерных технологий и свободного программного обеспечения в высшей школе.



Наш эксперт

Александр Недбайлов — доцент, старший преподаватель кафедры «Прикладная математика и информатика» Дальрыбвтуза, г. Владивосток.

Рбота с приложениями в среде свободного программного обеспечения технологически мало чем отличается от работы в приложениях Windows. Этот факт подтверждается результатами, которые были получены в ходе проведенных исследований, касающихся технологий поддержки проектного обучения.

Эти исследования, о которых речь пойдет ниже, начались достаточно тривиально — с распоряжения проректора по учебной работе о проведении занятий по компьютерным технологиям с преподавателями общетехнических и выпускающих инженерных кафедр ВУЗа в рамках курсов повышения квалификации.

Опрос потенциальных слушателей на предмет «а что вы хотели бы узнать и чему научиться» показал, что необходимы (в основном) навыки подготовки эскизов технического характера и анимации механизмов и процессов, создания и исследования их компьютерных моделей. В отношении программирования мнения разделились в выборе языка, который студентам хотелось бы освоить. Общим же было предложение «Нам не нужен традиционный курс информатики» (имелся в виду тот ее вариант, который изучался студентами).

Контингент слушателей и их предпочтения не предполагали проведения занятий в традиционном варианте (лекции и лабораторные/практические занятия). Поэтому за основу были приняты такие варианты:

- » «лекция — практика» (базовая теория и освоение технологии по принципу «выполняем сначала вместе, затем делаем самостоятельно»);
- » самостоятельная практическая работа в двух вариантах — репродуктивном (лист с заданием) и творческом (эскиз по теме-тике одной из читаемых дисциплин).

В процессе обсуждения понятия «компьютерная технология», которое наиболее часто цитировалось в учебной литературе, слушателями были высказаны мнения о том, что оно относится к устоявшемуся технологическому процессу выпуска определенного изделия или автоматизированной системе управления. Слушателей больше интересовали компьютерные технологии, которые обеспечивали бы инженерную проектную (т. е. творческую) деятельность.

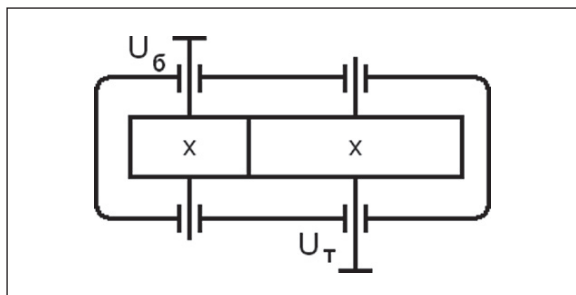
Обращение к литературе для поиска необходимых базовых понятий не принесло желаемых результатов. Поэтому пришлось обратиться в библиотеку за учебными пособиями, авторами которых являются преподаватели ВУЗа, и к архивам курсовых и дипломных проектов — с тем, чтобы выявить и классифицировать объекты, которые в них содержатся, и предложить способы (технологии) их подготовки (создания).

Так, графические эскизы, которые представлены в студенческих работах, были классифицированы по трем видам:

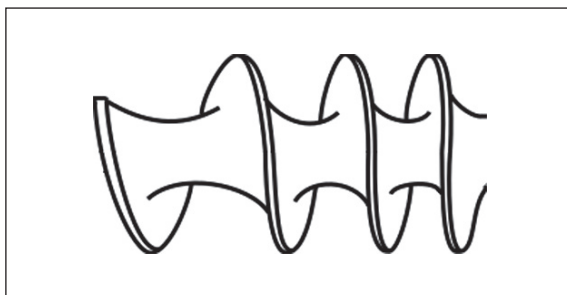
- » содержащие типовые графические примитивы и их части (пример — на рис. 1);
- » содержащие кривые (пример — на рис. 2);
- » комбинированные эскизы.

Стало понятно, что для подготовки таких принципиально разных изображений нужны и отличающиеся технологические процессы. В доступной литературе таких технологий обнаружено не было. Поэтому были разработаны и представлены слушателям технологии подготовки эскизов в среде графического редактора *Paint*, названные в дальнейшем методами.

Первое их применение показало, что требуется «идентификатор», который однозначно ассоциировался бы с технологическим процессом и был бы связан с чем-то уже знакомым пользователям.



› Рис. 1. Кинематическая схема редуктора.



› Рис. 2. Деталь механизма.

Поэтому внимание было обращено к методу опорных сигналов В. Ф. Шаталова.

Каждый метод получил название:

- › метод конструктора (работает с примитивами и их частями);
 - › метод изменения кривых;
 - › метод дополнительных элементов (вспомогательный);
- и дополнительно графическое изображение.

Были подготовлены учебное пособие и сборник заданий для аудиторной и самостоятельной работы.

По итогам трехлетней работы можно сделать вывод, что переход к обучению преподавателей, студентов 1-го и 2-го курсов, учащихся лица, на основе технологического подхода себя вполне оправдал. К тому же студенты начинают лучше понимать, что такое технология, как формируется технологическая последовательность операций, что такое технологическая карта, почему нарушение технологии приводит к результатам, не соответствующим критериям оценки, почему отсутствие технологической проработки задания существенно увеличивает время его выполнения.

В дальнейшем было проверено, как предложенные методы работают в векторной среде, в частности, во *Flash*. Оказалось, проблем при этом почти не возникает.

Появившееся вскоре любопытство к работе в *Ubuntu* позволило применить разработки в графическом редакторе *KolourPaint*. Отличий от работы в среде *Paint* практически не было.

Следовательно, технологический подход к подготовке графических эскизов себя вполне оправдал. Он оказался универсальным и кросс-платформенным.

Сопоставить его можно с работой за слесарным (столярным) верстаком. На каждом есть набор инструментов. Сначала осваиваются приемы работы инструментами, затем изготавливается какая-либо конструкция по образцу, затем можно переходить к самостоятельной проектной (по сути, творческой) деятельности. Если инструментов на верстаке достаточно, нужен ли другой верстак? На него можно перейти в случае необходимости. Если же инструментов больше, чем обычно требуется, их можно изучать и использовать по мере необходимости, выстроив нужную технологическую последовательность, что при наличии опыта будет не так уж и сложно.

Когда в среднем образовании появился *Linux*, видимых изменений в курсе информатики делать не пришлось. Тогда как имевший ранее место подход, связанный с изучением возможностей программных средств, привел к сложностям с методическим обеспечением, что было отражено в ряде публикаций преподавателей школ и методических объединений.

Появившийся опыт работы с графикой позволил провести исследование, как работают предложенные методы подготовки эскизов в разных средах, завершившееся в 2010 г. подготовкой учебных пособий по технологиям подготовки эскизов и кадров для создания анимации сначала в *Paint* и *KolourPaint*, затем *Photoshop* и *GIMP*.

В дальнейшем было высказано предположение о том, что и обработка другой (текстовой, числовой) информации тоже может

подлежать технологизированию, что привело к следующим сопоставлениям:

- › технологий подготовки текстовых документов (*Word*, *Writer*, *AbiWord*);
- › технологий выполнения расчетов (*Excel*, *Calc*, *Gnumeric*);
- › технологий подготовки презентаций (*PowerPoint*, *Impress*);
- › технологий работы с файлами (операционные системы *Windows* и *Linux*, файловые менеджеры «Проводник», *Nautilus*, *Konqueror*, *Thunar*, *Midnight Commander*).

Когда появился *Linux*, видимых изменений в курсе информатики делать не пришлось.

Результатом явились учебные пособия, подготовленные как для преподавателей, так и для студентов и учащихся вузовского лица.

Естественно, как и при работе с графикой, речь не шла о том, чтобы раскрыть все возможности перечисленного выше программного обеспечения. В этом для студентов технических специальностей на данном этапе нет большой необходимости. Речь идет о том, чтобы обеспечить решение тех задач, с которыми сталкиваются преподаватели и студенты в ходе учебной и проектной деятельности.

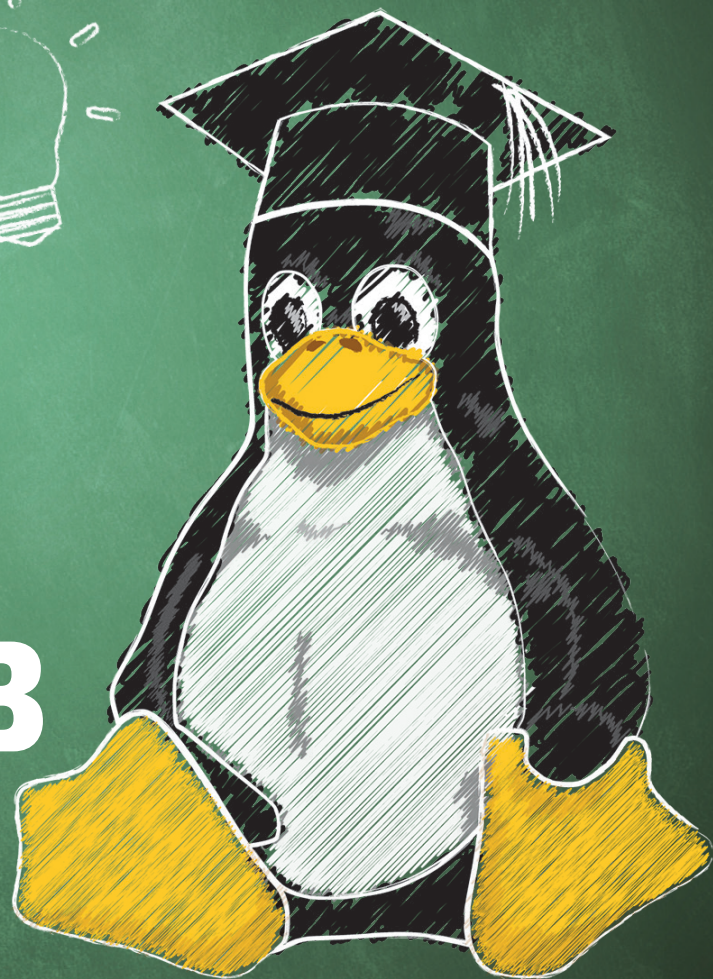
Как показала практика, после знакомства с возможностями программных средств, в дальнейшем не востребованными в ходе использования этих программ, они забываются (теория усвоения П. Я. Гальперина).

Результаты использования предложенного технологического подхода:

- › сокращается время подготовки эскиза;
- › критерии оценивания, которые предложены студентам, в значительной мере соблюдаются ими в дальнейшем;
- › студенты получают дополнительные возможности для самовыражения, а также поиска работы или получения второго высшего образования, связанного с веб-дизайном, графикой и анимацией;
- › студенты достаточно легко выбирают программную среду для решения текущих задач;
- › освоение свободного программного обеспечения в среде *Windows* и операционной системы *Linux* проходит быстрее;
- › появляется интерес к администрированию *Linux*, в том числе и серверных дистрибутивов.

Через три года была проведена оценка того, запомнили ли студенты технологии подготовки эскизов. Так, например, анкетирование студентов 5-го курса после защиты ими диплома показало, что более 60% заполнивших анкету совершенно точно ассоциируют название «метод конструктора» с технологией подготовки эскизов и в состоянии объяснить, как она реализуется. LXF

Linux
Rulez!



Linux глазами педагогов

I  Linux

Под чутким руководством Людмилы Метлиной было исследовано влияние смены операционной системы на работу школы.



Наш эксперт

Людмила Метлина — директор школы МОУ СОШ № 11 г. Искитим Новосибирской области.

В нашей стране, как и во многих странах мира, принята государственная программа перехода на «альтернативное» ПО. Процесс легализации ПО на большинстве предприятий, и в том числе — в сфере высшего и среднего образования, уже начался.

Речь идет, конечно же, об использовании СПО на базе становящейся всё более популярной ОС Linux (Линукс). Само по себе это дело хорошее. Но, к сожалению, в постперестроечной России сложилась парадоксальная ситуация, когда наиболее активные потребители информационных технологий — наука и образование — финансируются слишком слабо, чтобы приобретать лицензионное ПО по коммерческим ценам. И даже при предоставляемых фирмами-производителями скидках на «education-версии» оснащение всего парка компьютеров «лицензиями» — бремя очень тяжелое.

» **Цель** Доказать, что ОС Linux может равноправно заменить ОС Windows в образовании РФ.

» Задачи

- 1 Исследовать операционные системы ОС Linux, ОС Windows.
 - 2 Сравнить данные ОС.
 - 3 Рассмотреть возможность установки различных видов ОС ALT Linux на компьютеры школы № 11 (анализировать технопарк школы).
 - 4 Провести социологический опрос среди учащихся 9–11-х классов школы № 11 на выявление отношения пользователей к ОС Linux, ОС Windows.
- » **Гипотеза** Если мы введем операционную систему Linux, то ухудшится информационно-образовательный процесс.
- » **Объект исследования** Процесс перехода на СПО.
- » **Предмет исследования** ОС Linux, ОС Windows.
- » **Метод исследования** Теоретический и практический (изучение литературы по ОС Linux, ОС Windows, анкетирование, анализ компьютерного парка школы).

Теоретическая часть

Свободное программное обеспечение

Свободное программное обеспечение (СПО) — это широкий спектр программных решений, в которых права пользователя («свободы») на неограниченные установку и запуск, а также свободное использование, изучение, распространение и изменение (совершенствование) программ защищены юридически авторскими правами при помощи свободных лицензий.

Особенность программного обеспечения состоит в том, что оно производится в одной форме — в виде исходного текста, а распространяется и используется в другой — в виде исполнимых программ, машинных кодов, по которым невозможно однозначно восстановить исходный текст. Чтобы эффективно изменять программу, исправлять ошибки или даже просто точно установить, что и как делает программа, необходимо располагать ее исходным текстом. Именно предоставление исходного кода является отличием СПО от платного или коммерческого ПО.

История создания фонда Свободного программного обеспечения

В 1984 г. имел место переломный момент в сфере программного обеспечения: появилось много коммерческих фирм, которые не предоставляли исходный код. И тогда Ричард Столлмен [Richard Stallman] в поисках единомышленников создал некоммерческую организацию «Фонд свободного программного обеспечения».

Своей основной целью Фонд ставил сохранение программного обеспечения, процесс разработки которого всегда будет гарантированно открытым, а исходные тексты всегда доступны. Более масштабная цель Фонда — разработка операционной системы, целиком состоящей из открыто разрабатываемого программного обеспечения.

Ричард Столлмен сформулировал понятие «свободное программное обеспечение», в котором отразились принципы открытой разработки программ в научном сообществе, сложившемся в американских университетах в 1970-е гг. Столлмен сформулировал также принципы — критерии свободного программного обеспечения. Эти критерии оговаривают те права, которые авторы свободных программ передают любому пользователю:

- » Программу можно свободно использовать с любой целью («*нулевая свобода*»).

- » Можно изучать, как программа работает, и адаптировать ее для своих целей («*первая свобода*»). Условием этого является доступность исходного текста программы.
- » Можно свободно распространять копии программы в помощь товарищу («*вторая свобода*»).
- » Программу можно свободно улучшать и публиковать свою улучшенную версию с тем, чтобы принести пользу всему сообществу («*третья свобода*»). Условием этой третьей свободы является доступность исходного текста программы и возможность внесения в него модификаций и исправлений.

Linux — операционная система в СПО: История создания

Линус Торвалдс — создатель ядра «Linux».

Апрель 1991 г. — 21-летний Линус Торвалдс начал работу над некоторыми механизмами операционной системы. Он начал с эмулятора терминала и планировщика задач.

В отличие от коммерческих систем, таких как Microsoft Windows или Mac OS X, Linux не имеет географического центра разработки. Нет и организации, которая владела бы этой системой; нет даже единого координационного центра. Программы для Linux — результат работы тысяч проектов. Некоторые из этих проектов централизованы, некоторые сосредоточены в фирмах. Многие про-

Linux не имеет географического центра разработки и даже координационного центра.

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

Linux является UNIX-совместимой, однако основывается на собственном исходном коде. Именно такая гибкая и динамичная

»

Плюсы и минусы ОС Linux

Плюсы

- » Сравнительно низкая стоимость. В более или менее большом городе вполне реально получить диск с каким-либо дистрибутивом Linux по цене чистого CD/DVD-носителя, не говоря уже о бесплатном скачивании из Интернета. Можно заказать и купить наиболее популярные дистрибутивы через Интернет с доставкой по почте, что можно сделать в любом населенном пункте, где работает почта. При этом, имея всего одну физическую копию дистрибутива Linux, вы получаете право установить его на любое количество компьютеров.
- » Малое количество вредоносных программ для этой платформы, что позволяет избежать дополнительных расходов по предотвращению или ликвидации ущерба от вредоносных программ.
- » Независимость от разработчика. Если вам потребовалась какая-то функциональность,

отсутствующая в ОС Linux, вы можете ее добавить своими собственными усилиями. Такая возможность существует благодаря тому, что ОС Linux распространяется не только в двоичном виде, но и в исходных кодах, причем нет никаких запретов на модификацию этих исходных кодов.

Минусы

- » Худшая по сравнению с платформой Windows поддержка компьютерного оборудования — в частности, внешнего, такого как принтеры или USB-устройства. Пожалуй, хуже всего дело обстоит со сканерами и USB, а также внутренними HSF/HCF модемами. Вполне реальна ситуация, когда под ОС Linux какое-то оборудование вам использовать не удастся. Поэтому перед приобретением «железки» следует поискать в Интернете информацию о возможности ее работы под ОС Linux.

- » Меньшее, чем для платформы Windows, количество профессиональных прикладных программ. Более того, если речь идет о некоторых программах — безусловных лидерах в своих областях применения, то под операционной системой Linux, скорее всего, не будет самых свежих версий этих программ, а сопоставимые по функциональности программы для Linux пока что от них отстают. Частично этот недостаток компенсируется за счет того, что некоторые (но не все) Windows-программы можно запустить на платформе Linux через эмуляторы или прослойку WINE.
- » Меньшее, чем для платформы Windows, количество хороших или приличных специалистов. Иными словами, если вам потребуется помощь в решении технических проблем, то найти человека, достаточно хорошо разбирающегося в Linux, будет не так просто.

Плюсы и минусы ОС Windows

Плюсы

- » Поддержка очень большого ассортимента компьютерного оборудования. Какая бы экзотическая «железяка» вам ни попалась, почти наверняка вы сможете ее использовать под Windows. Хотя, быть может, вам и потребуется время на поиски нужной программы-драйвера.
- » Огромное количество прикладных программ: ныне это уже, наверное, более ста тысяч наименований. Для любой прикладной задачи на платформе Windows есть как минимум несколько десятков

программ, а для популярных задач их существует сотни. Выбрать есть из чего. На любой вкус.

- » Большое количество специалистов, которые более или менее хорошо знают семейство ОС Windows. То есть, если вам потребуется помощь, вы ее найдете легко и за умеренную цену.

Минусы

- » Сравнительно высокая стоимость. В самом дешевом варианте это более \$50, при том, что такая «дешевая» Windows, приобретаемая в комплекте

с новым компьютером, «привязана» к этому компьютеру. А это значит, что при смене компьютера вам снова придется тратить деньги на Windows.

- » Очень большое количество вредоносных программ (так называемые компьютерные вирусы). Это особо серьезная проблема для версии Windows XP.
- » Жесткая зависимость от разработчика. ОС Windows распространяется только в двоичном виде, который труднодоступен для изменения; но более того, компания Microsoft вообще запрещает вносить какие-либо изменения в рабочие коды ОС Windows.

система разработки, невозможная для проектов с закрытым кодом, определяет исключительную экономическую эффективность Linux.

Сравнение операционных систем Linux и Windows

Windows и Linux трудно сравнивать на равных из-за различия между системами.

1 Полный доступ или отсутствие такового

Пожалуй, одно из самых существенных различий между Linux и Windows — наличие или отсутствие доступа к исходному программному коду. Linux разрабатывается в соответствии с открытым лицензионным соглашением GNU (GPL), поэтому все пользователи имеют право и возможность просматривать и изменять

исходный программный код вплоть до самого ядра, которое служит основой операционной системы Linux. Увидеть исходный код ОС Microsoft вам никогда не удастся, тем более что в нем так или иначе содержатся двоичные фрагменты.

2 Свободное лицензирование или лицензионные ограничения

Другое фундаментальное различие между Linux и Windows проходит на уровне лицензий на использование программного обеспечения. Открытое лицензионное соглашение на ОС Linux позволяет свободно модифицировать программное обеспечение, использовать, публиковать его от своего имени и даже продавать — главное, чтобы исходный код по-прежнему оставался открытым. К тому же, GPL позволяет загрузить одну копию дистрибутива или приложения Linux и установить ее на неограниченном количестве компьютеров. Лицензия Microsoft ничего подобного не допускает.

Сравнение ОС GNU Linux и ОС MC Windows

Название характеристики	Windows	Linux
Доля установки при продаже компьютеров (OEM)	9,7	5,0
Оконные менеджеры/графическая среда	7,0	9,5
Системная консоль/командная строка	8,0	9,7
Размер инсталлятора	9,0	9,0
Простота установки	9,0	6,0
Время, затрачиваемое на установку	9,0	9,0
Наличие драйверов устройств	8,8	4,7
Инсталляция с помощью ознакомительного CD (Live CD)	7,0	9,3
Поставляемое программное обеспечение	4,0	8,9
Программы, которые можно установить дополнительно	8,7	8,8
Подготовка диска	7,0	9,0
Программа-загрузчик	7,9	8,0
Единообразие интерфейса	8,5	8,6
Единообразие между различными версиями	7,5	8,0
Единообразие между программами	8,9	8,1
Единообразие процедуры обновления программ и ОС	9,9	9,9
Настройка	7,0	9,9
Безопасность	7,5	9,7

Техническую поддержку по Linux можно получить в огромном сообществе пользователей — на форумах, в поисковиках и на сотнях специализированных сайтов. Но данная техническая помощь у компании Microsoft немного хуже, чем у Linux, о чем красноречиво повествуют форумы и блоги.

3 Полная или частичная аппаратная совместимость

Одна из вечных проблем Linux, постепенно уходящая в прошлое — это проблема аппаратной совместимости. Несколько лет назад для успешной установки Linux на настольном компьютере все компоненты системы приходилось подбирать вручную, иначе не было никакой гарантии, что ОС заработает. С Windows совместимо практически любое оборудование.

4 Командная строка или ее отсутствие

Как бы далеко операционные системы Linux ни зашли в своем развитии, и как бы ни был великолепен их графический интерфейс пользователя, командная строка всегда будет оставаться незаменимым инструментом для выполнения любых административных задач. В Windows командной строкой пользоваться тоже можно, но отнюдь не так широко, как в Linux. К тому же Microsoft изо всех сил старается спрятать командную строку от пользователей, и многие даже не подозревают о ее существовании.

Приспособляемость или негибкость

Linux — это комната, высоту пола и потолка в которой можно регулировать именно так, как пользователю будет удобнее. В Windows «пол» и «потолок» зафиксированы навсегда.

Автоматизированное или ручное подключение съемных носителей

Одна из вечных проблем начинающих пользователей Linux — подключение съемных носителей. Сегодня конфигурация подсистем Linux позволяет пользоваться съемными носителями

Linux можно регулировать именно так, как пользователю будет удобнее.

точно так же, как в Windows, но вообще это не является нормой. В Windows подключение съемных носителей практически полностью автоматизировано.

Один или несколько уровней запуска

Речь идет о присущей Linux способности останавливать работу системы на разных уровнях запуска. В Linux допускается работа как на уровне командной строки (уровень запуска 3), так и на уровне графического интерфейса пользователя (уровень запуска 5). Это здорово спасает положение, когда Linux по какой-то причине перестает функционировать и требуется выявить источник проблемы.

В Windows, если повезет, до командной строки можно добраться в безопасном режиме.

Практическая часть

Установка ОС и ПО

С целью последующей установки операционной системы ALT Linux Школьный 5.0.1 и соответствующего образовательному процессу программного обеспечения был проведен анализ компьютерного парка МОУ СОШ № 11.

В нашей школе имеется 38 компьютеров, предназначенных для обучения. По соответствию требованиям ALT Linux они разделились на 3 группы. Результаты показаны в таблице 1.

Таблица 1

Вид ОС	Системные требования	Число ПК
ALT Linux Легкий	От 128 МБ ОЗУ (рекомендуется 256 МБ), дисковое пространство: от 7 ГБ	12
ALT Linux Юниор	От 256 МБ ОЗУ (рекомендуется 512 МБ), дисковое пространство: от 11 ГБ	10
ALT Linux Мастер	От 1 ГБ ОЗУ, дисковое пространство: от 12 ГБ	16

Таблица 2

Марка принтеров	Количество
Samsung SCX-4200	3
HP Color Laser Jet	3
Xerox Phaser 3116	2
Epson EPL 6200	1

Установка ALT Linux Мастер и ALT Linux Легкий на компьютер, отвечающий системным требованиям данной ОС, прошла успешно,

без каких-либо осложнений, и производилась согласно инструкции по установке программы.

Установка принтеров марки Samsung SCX-4200 и HP Color Laser Jet прошли без проблем, при аккуратном следовании инструкции операционной системы. Марки проверенных принтеров представлены в таблице 2.

По итогам данного исследования было выявлено, что в ОС ALT Linux Мастер есть необходимые драйверы для установки принтеров, которые имеются в нашей школе.

В школе имеются 3 интерактивных доски марки SMART Board software. Установка данного оборудования не производилась, но по описанию системных требований компонента видно, что данная марка вполне совместима с ОС ALT Linux Мастер.

Совместимость ПО

Было проведено исследование по выявлению совместимости ПО (таблица 3).

Из таблицы видно, что работают не все диски с цифровыми образовательными ресурсами (ЦОР).

Вывод:

Для более корректной работы программы WINE (программа, необходимая для открытия файлов Windows) рекомендуется скопировать устанавливаемый файл в формате .exe в папку Wine_c, которая находится в Домашней папке.

Причины, по которым программы не смогли установиться на ОС Linux:

- » Экспериментальная машина не отвечала системным требованиям устанавливаемых программ;
- » Системное оборудование экспериментальной машины не совместимо на 100%;
- » Имеются повреждения на дисках с программами или в самих программах.

Дистрибутивы Alt Linux для школы

Junior

Дистрибутивы серии Junior предназначены для решения различных прикладных задач пользователей. В Junior входят текстовые редакторы, словари, шрифты, несколько графических операционных сред, языки программирования и среды разработки, системы управления базами данных, образовательные программы, игры и утилиты администрирования системы. Одной из задач при составлении дистрибутива ALT Linux Junior было создание на базе Linux платформы для обучения информатике и информационным технологиям. Поэтому в нем присутствуют дополнительные учебные и методические пособия, книги и статьи по Linux и смежным темам.

Master

Задача дистрибутивов серии Master — служить универсальным инструментарием для разработчика, администратора и пользователя. Master — наиболее полный дистрибутив ALT Linux, в котором «всё под рукой». На основе Master можно строить

корпоративные информационные системы, использовать его в качестве серверной платформы, для разработки клиентских и серверных приложений, в образовательных учреждениях, а также применять в качестве профессиональной настольной операционной системы.

Server

Начиная с ветки 4.0, доступен специализированный серверный выпуск под названием Server. Сертифицирован ФСТЭК России:

- » Классификация по уровню контроля отсутствия недеklarированных возможностей — 4 уровень.
- » Показатели защищенности от несанкционированного доступа к информации — по 5 классу защищенности.

Учитель

Операционная система, содержащая расширенный набор приложений, основана на графической среде Xfce. Также дистрибутив включает в себя:

- » систему управления учебным процессом iTALC;

- » антивирус ClamAV;
- » клиент резервного копирования Bacula;
- » клиент централизованного обновления ПО

Перечисленные дистрибутивы включены в Регистр Российского программного обеспечения для закупок по 188-ФЗ.

В настоящее время комплект Школьный вошел в дистрибутив для образовательных учреждений Альт Образование (последняя версия — 8, обновления по безопасности поддерживаются до 16 декабря 2019 г.). На базе этого дистрибутива можно построить инфраструктуру для предоставления учащимся учебных материалов на стационарных компьютерах, ноутбуках, нетбуках, планшетах и электронных книгах. Эта особенность позволяет не только централизованно управлять учебным процессом, но и осуществляет взаимодействие учеников и учителя в ставшей привычной для всех форма чатов и форумов. Сообщения могут содержать задания, их решения и комментарии. Также возможен обмен файлами любых форматов, как между преподавателем и учащимся, так и между учащимися.

Таблица 3

Название программ в ОС Windows	Совместимо ли с Linux?
Microsoft Office Word 2007/2003	Да
Microsoft Office PowerPoint 2007/2003	Да
Microsoft Office Excel 2007/2003	Да
Microsoft Office Access 2007/2003	Нет
Конструктор сайтов	Да
Учебная программа (Информатика 3–4)	Да
Учебная программа по экологии	Да
Тестовые задания ЕГЭ по обществознанию	Да
Электронная библиотека по географии	Нет
Обществознание 11-й класс	Нет
Математика 5–9-й класс	Да
Физика	Нет
Экология 11-й класс	Да
Право	Да
Российская символика	Да
Химия 11-й класс	Нет
От пуга до лазера	Нет
От Москвы до Рейхстага	Да
Химия 7-й класс	Да
Древнейшая история 5-й класс	Нет

Проблема, выявленная при установке программного обеспечения:

- » На экране монитора в некоторых его частях появлялись черные линии и принимались мигать, вызывая специфическую неприятную рябь.

При открытии документов Microsoft Office 2007 (Word, Power Point, Excel) не требуется конвертация их в формат Microsoft Office 2003.

Анкетирование

- Был проведен опрос учащихся 9–11-х классов, выявляющий знания о различных ОС, отношение к переходу на операционную систему Linux.

Данные о знаниях ОС среди школьников 9–11-х классов

Название ОС	Количество
Linux	25
Vista	30
Mac OS	5
Windows 7	20
Macintosh	5

Из таблицы видно, что учащиеся имеют представление о разных ОС.

Впечатление школьников после работы на ОС Linux

На первом этапе учащимся предлагалось работать только на уровне офиса.

Впечатление	Количество
Положительное	38
Отрицательное	46
Нет впечатлений	1

Данные опроса показывают, что у учащихся налицо интерес к новой ОС.

Отношение к переходу на ОС Linux (ученики)

Переход...	Количество
Нужен	15
Не нужен	50
Безразличен	30

Переход на новую ОС учащиеся посчитали ненужным, т. к. на домашних компьютерах в большинстве случаев стоит ОС Windows и надо постоянно перестраиваться.

2 Так как переход на новую систему вызвал беспокойство, был проведен опрос учителей школы касательно перехода на новую ОС Linux.

Отношение к переходу на ОС Linux (учителя)

Переход...	Количество
Нужен	0
Не нужен	47
Безразличен	7

Резкое отрицание перехода на новую ОС объясняется тем, что учителя беспокоятся о уже наработанных материалах и будут ли они совместимы с ОС Linux. Вторая причина — это необходимость переучивания. У всех опрошенных на домашних ПК стоит ОС Windows. Возникнет необходимость организации курсов.

Заключение

Пожалуй, одно из самых существенных различий между Linux и Windows — наличие или отсутствие доступа к исходному программному коду. Linux разрабатывается в соответствии с открытым лицензионным соглашением GNU (GPL), поэтому все пользователи имеют право и возможность просматривать и изменять исходный программный код вплоть до самого ядра, которое служит основой операционной системы Linux. Увидеть исходный код ОС Microsoft вам никогда не удастся.

При сравнении двух систем выяснилось, что:

- » ОС ALT Linux Легкий и Мастер отвечают запрошенным системным требованиям. Установка на экспериментальные компьютеры прошла успешно, при последовательном выполнении инструкции, которая понятна любому пользователю.
- » Установка дополнительных устройств (принтеров) прошла успешно. Драйвера, необходимые для установки всех школьных принтеров, имеются в репозиториях ОС.

Системные требования для дистрибутивов

Дистрибутив	ОЗУ		Дисковое пространство (минимум), ГБ
	Минимум, МБ	Рекомендуется, ГБ	
Альт Линукс 7.0.5 Школьный Сервер	512	1	10
Альт Линукс 7.0.5 Школьный Юниор	512	1	11
Альт Линукс 7.0.5 Школьный Мастер	512	2	20
Альт Линукс 7.0.5 Школьный Учитель	512	1	11

Очевидно, что с дистрибутивом Альт Линукс справится самое скромное оборудование (данные приведены для 32-битных систем).

- » Установленные нами ЦОР и электронные образовательные ресурсы (ЭОР), отвечающие системным требованиям экспериментальной машины, прошли успешно и без осложнений.
 - » Документы, созданные в Microsoft Office 2003/2007, открываются в Linux без дополнительного конвертирования.
- По результатам анкетирования видно, что учащиеся интересуются новыми ОС, но подходят к этому вопросу с осторожностью, из-за неудобства работы на разных ОС — дома и в школе.

Учащиеся интересуются новыми ОС, но подходят к этому вопросу с осторожностью.

Данные обстоятельства показывают, что между ОС ALT Linux и ОС Windows нет кардинальных различий, а значит, переход на ОС ALT Linux пройдет безболезненно для педагогов и учащихся образовательных учреждений.

Из вышеизложенного следует: так как в ОС ALT Linux можно проводить все те действия, что и в ОС Windows; значит, при переходе на СПО информационно-образовательный процесс не изменится. Таким образом, выдвинутая мною гипотеза об ухудшении образовательного процесса при переходе на Linux полностью опровергается. **LXF**

Список литературы

Полезные для изучения источники:

- 1 Баррет Д. Linux — основные команды. Карманный справочник. — М.: Кудиц-образ, 2005
- 2 Колисниченко Д. Н., Аллен Питер В. LINUX: полное руководство. — СПб: Наука и Техника, 2006
- 3 Курячий Г. В., Маслинский К. А. Операционная система Linux. Курс лекций. Учебное пособие. — М.: Интернет-университет информационных технологий, 2005
- 4 Маслинский К. А. Операционная система Linux. — ИНТУИТ.ру, 2005
- 5 ALT Linux 2.3 Junior. Справочник пользователя. docs.altlinux.org/ru-RU/archive/2.3/html-single/junior/alt-docs-schooljunior/ch04.html
- 6 Федосеев А. UNIX: учебный курс. fedoseev.net/materials/courses/admin/index.html

- 7 Сборник наиболее часто задаваемых вопросов (FAQ), касающихся использования программных продуктов ALT Linux Team. www.altlinux.org/FAQ
- 8 Alt Linux Wiki. www.altlinux.org/
- 9 Костромин В. А. Основы работы в ОС Linux. Курс ИНТУИТ.ру www.intuit.ru (эл. в.)
- 10 www.opennet.ru/ — различная документация по Linux
- 11 docs.kde.org — официальная документация по KDE
- 12 ru.wikipedia.org — свободная энциклопедия
- 13 server.179.ru/wiki/ — сайт школы 179 МИИО
- 14 www.gnu.org/ — официальный сайт проекта GNU
- 15 www.linuxcenter.ru/lib/ — сайт ГНУ/Линуксцентра
- 16 www.otstavnov.com/ — личный сайт Максима Отставнова
- 17 www.vdel.ru/ru — официальный сайт VDEL

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





В ноябрьском номере

Стать экспертом...

По безопасности Linux!

Воспользуйтесь новейшими технологиями защиты и проведите в своей сети тесты на вторжение.

Инструменты шифрования

С помощью свободного ПО превратите свои самые ценные файлы в невинный с виду фоновый шум.

Snap атакует

Почему все так расшумелись насчет Snap? Пора разобраться и оценить новый способ упаковки программ.

Swar умер, да здравствует Swar

В Ubuntu объявили раздел подкачки покойником, и мы пробуем новую (хотя и старую) систему на основе файлов.

OggCamp, детка!

Мы (точнее, Лес Паундер) ведем репортаж с оживленной неконференции, которая вернулась под новым руководством!

Содержание будущих выпусков может меняться — вдруг из-за паранойи мы побьемся выйти из тени...

LINUX FORMAT

Главное в мире Linux

Журнал зарегистрирован Федеральной службой по надзору за соблюдением законодательства в сфере массовых коммуникаций и охране культурного наследия ПИ № ФС77-21973 от 14 сентября 2005 года. Выходит ежемесячно. Тираж печатной версии 1200 экз., распространение электронной версии 30000 экз.

РЕДАКЦИЯ РУССКОЯЗЫЧНОЙ ВЕРСИИ

Главный редактор

Кирилл Степанов info@linuxformat.ru

Литературный и выпускающий редактор

Елена Толстякова

Переводчики

Елена Ессяк, 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

Генеральный директор

Павел Фролов

Учредители

Частные лица

Издатель

ООО «Линукс Формат»

Отпечатано в типографии ООО «ЛД-ПРИНТ»

196644, Санкт-Петербург, Колпинский р-н, пос. Саперный, территория предприятия «Балтика», д. 6/н, лит. Ф
Тел. (812) 462-8383, e-mail: office@ldprint.ru

РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ

Редактор Нейл Мор [Neil Mohr] neil.mohr@futurenet.com

Научный редактор Джонни Бидвелл [Jonni Bidwell]

jonni.bidwell@futurenet.com

Художественный редактор Эфраин Эрнандес-Мендоза

[Efrain Hernandez-Mendoza] efrain.hernandez-mendoza@futurenet.com

Выпускающий редактор Клиффорд Хоуп [Cliff Hope]

clifford.hope@futurenet.com

Директор редакции По Ньюмен [Paul Newman]

Старший художественный редактор Джо Гулливер [Jo Gulliver]

ПОДГОТОВКА МАТЕРИАЛОВ

Джонни Бидвелл [Jonni Bidwell], Нейл Ботвик [Neil Bothwick], Алекс Коок [Alex Cox], Нейт Дрейк [Nate Drake], Дан Фрост [Dan Frost], Том Маркс [Tom Marks], Бобби Мосс [Bobby Moss], Адам Оксфорд [Adam Oxford], Ник Пирс [Nick Peers], Марк Пикаванс [Mark Pickavance], Лес Паундер [Les Pounder], Симон Раффайнер [Simon Raffener], Маянк Шарма [Mayank Sharma], Шашанк Шарма [Shashank Sharma], Александр Толстой [Alexander Tolstoy], Михалис Цукалос [Mihalis Tsoukalos], Энди Уорбертон [Andy Warburton], Евгений Балдин, Андрей Гондаренков, Людмила Метлина, Александр Недбайлов, Дмитрий Пантеличев, Алексей Федорчук, Максим Черепанов

Иллюстрации Шейн Коллиндж [Shane Collinge]

Иллюстрация с обложки www.magicforch.com

КОНТАКТНАЯ ИНФОРМАЦИЯ

UK: Linux Format, 30 Monmouth Street, Bath BA1 2BW

Тел. +44 (0) 1604 251045, email: linuxformat@myfavouritemagazines.co.uk

РОССИЯ: Санкт-Петербург, пр. Медиков, 5, корп. 7

Тел. +7 (812) 309-0686, e-mail: info@linuxformat.ru

По вопросам сотрудничества, партнерства, оптовых закупок:
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

За информацией о журналах, издаваемых Future plc group company, обращайтесь на сайт www.futurepic.com.



© Linux Format 2005

© Future Publishing Ltd 2005

BATH • LONDON • MILAN • NEW YORK • PARIS • SAN DIEGO • SAN FRANCISCO

16+

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

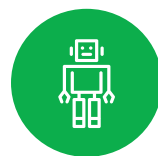
- Праздничные мастер-классы
- Городской лагерь на время каникул
- Курсы повышения квалификации для педагогов



Инновационная программа обучения

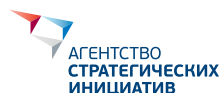


Собственная методическая база



Российские технологии и оборудование

ВООБРАЖАЙ! ПРОГРАММИРУЙ! СОЗДАВАЙ!



www.robboclub.ru

8 800 777 2985

vk.com/robboclub ● info@robboclub.ru

HETZNER
ONLINE

БЫСТРЫЙ КАК МОЛНИЯ

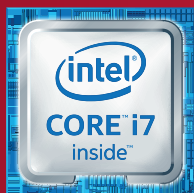
ОСТРЫЙ КАК ЛЕЗВИЕ

НОВИНКА



Выделенный сервер EX51-SSD-GPU

Intel® Core™ i7-6700 Quad-Core Skylake Processor
64 ГБ DDR4 RAM
2 x 500 ГБ SATA 6 Gb/s SSD
Видеокарта GeForce® GTX 10800
100 ГБ место для резервных копий
50 ТБ трафик*
Без минимального контракта
Установка 7200 рублей



7200 рублей в месяц

Идеальное решение для ресурсоёмких вычислений.

Новый выделенный сервер EX51-SSD-GPU содержит невероятно мощную графическую карту GeForce® GTX 1080 для ускорения графических приложений и быстрого 3D-рендеринга.

www.ru.hetzner.com

* Нет платы за превышение. При превышении 50 ТБ/месяц скорость соединения ограничивается (подсчёт ведётся по исходящему трафику, входящий и внутренний трафик не учитывается). Опционально можно снять ограничение, подтвердив оплату 85 руб. за каждый дополнительный ТБ.

Все цены вкл. НДС 18%. Цены могут измениться без уведомления. Все права защищены соответствующими производителями. Intel, логотип Intel, Core и Core Inside являются товарными знаками корпорации Intel в США и других странах.