DVD ДАРОМ! UBUNTU 15.04 + EMBY

LINUX FORMAT

Главное в мире Linux

72 страницы учебников и статей!

» Ищем лучший рабочий стол
» Печать — это просто
» Blender 3D в Голливуде

+ Сервисы синхронизации файлов Linux: копируйте и пересылайте

Январь 2015 № 1 (204)

ВЕЩАЕМ!

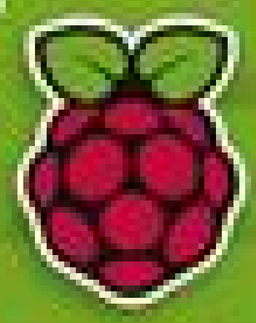
Соберем лучший медиа-центр на Ubuntu

» Простое пошаговое руководство
» Соорудим идеальную медиа-библиотеку
» Запись телепрограмм вживую
» На Android, Pi и не только!

Web: Жмем 'Сохранить'

« Правительственные документы и должны быть доступны »

Алексис Росси — об Internet Archive с. 36



Плюс: Для Pi!

8-страничный гид по Raspberry Pi

» Astro Pi: приступим уже!
» Создадим сервер печати Pi
» Мониторы для Raspberry Pi

ВОПРС № Linux

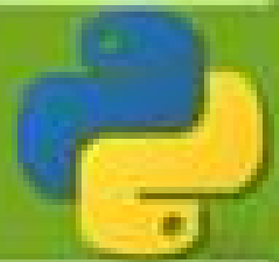
Сетевая обработка

» Дарим науке простаивающие циклы процессора своего ПК

Интерфейсы на GTK

Python Glade

» Создаем элегантные интерфейсы для приложений



iTeleRadio

ИНТЕРАКТИВНОЕ **РАДИО&ТЕЛЕВИДЕНИЕ**

ОГРН 1097847050021

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

Вот о чем мы спросили наших экспертов: что создает вам абсолютную отключку и отдых после тяжелого рабочего дня?

Джонни Бидвелл
Последнее время я много играю в *Transistor*. Это научно-фантастическая ролевая игра-экшн с отличным сюжетом и великолепной графикой, и играть в нее очень весело. И она работает на Linux. Она, плюс моя склонность к психоделической музыке, развлекает меня до глубокой ночи.

Нейл Ботвик
Я думаю, что приятнее всего в Linux быть в состоянии подправлять и настраивать всё, что душе угодно. А противнее всего — пытаться выяснить, что же я поломал своими правками и настройками и как это лечится.

Лес Паундер
HD-фильмы и коробки с DVD-сериалами — это замечательно, но самое технарское развлечение — просмотр *Звездных войн* через Telnet: эпизод 4 во всей красе в анимации ASCII-арт. Откройте терминал и введите Telnet towel.blinkenlights.nl, да не забудьте запастись попкорном!

Ник Пирс
Вас не удивит (если учесть, что я написал про это статью), что для меня самая развлекательная штука в Linux — обслуживание своей медиа-библиотеки по требованию через Raspberry Pi 2. Ну есть ли способ проводить время лучше, чем просматривая ТВ-шоу и художественные фильмы?

Крис Торнетт
После трудового дня по правке моих же опечаток меня занимают — во всех смыслах — мои дети. Умыв их и рассовав по постелям (стараясь не перепутать порядок действий), я гляжу Netflix на своем разблокированном хромбуке, скачу по 2D/3D-платформам в *Fez* или увиливаю от смерти в *Banner Saga*.

Валентин Синицын
Раньше на этот вопрос было принято отвечать: «Пишу программы для Linux». Но сейчас я пишу программы для Linux весь рабочий день. Значит, для отдыха надо написать что-то для Windows. Где там мой Visual Studio Community Edition?..

Памяти Основателя

» 28 декабря прошлого года ушел из жизни Ян Мёрдок [Ian Murdock]. За свои 42 года он успел сделать столько, что хватило бы на три жизни. Основаный им в 1993-м дистрибутив Debian стал одним из самых успешных проектов в мире СПО. По количеству производных он на порядок превзошел «классический» AT&T UNIX System V, а уж про число установок и говорить нечего. Этого уже хватило бы, чтобы войти в историю. А ведь были еще проект Indiana и очень важная работа по созданию LSB на посту технического директора Free Standards Group и Linux Foundation. А сколько еще было в планах...

Биография Яна удивительным образом похожа на биографию Линуса Торвальдса. Оба росли в среде европейских интеллектуалов, оба начали программировать еще в школьные годы, а уже на младших курсах университета смогли не просто писать хороший код, но и организовать активно работающее сообщество вокруг своего проекта. Согласитесь, это — отдельный талант. Не каждому дано. Оба не «ушли в бизнес» по пути наименьшего сопротивления «изобрети и продай», а остались Инженерами — людьми, которые делают мир лучше.

Настоящие инженеры не умирают: они живы, пока люди пользуются их изобретениями. Каждый раз, набирая apt-get, будем вспоминать Основателя...

PS Особенно грустно, что мы никогда не узнаем истинных причин и обстоятельств гибели Яна Мёрдока. Свидетелей нет. Официальные заявления департамента полиции Сан-Франциско очень напоминают неуклюжую попытку «замазать» эпизод грубейшего полицейского произвола. А судейский бюрократ всегда прикроет бюрократа полицейского. Даже в условиях «эталонной демократии» на «родине Линкольна»...

Кирилл Степанов
Главный редактор
» info@linuxformat.ru

Как с нами связаться

Письма для публикации: letters@linuxformat.ru
Подписка и предыдущие номера: subscribe@linuxformat.ru
Общие вопросы: info@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

LINUX FORMAT

Содержание

«Программирование — сочетание ремесла, любви и фантастики.» Чарльз Стросс

Обзоры

Tails 1.6 12
Свежий релиз дистрибутива для тех, кому есть что скрывать, или для тех, кто желает скрыть, что скрывать им совершенно нечего.

MySQL Fabric 1.6.2 14
Высокая доступность и масштабируемость — главные заботы современных вычислительных систем. Им поможет программный продукт от Oracle.

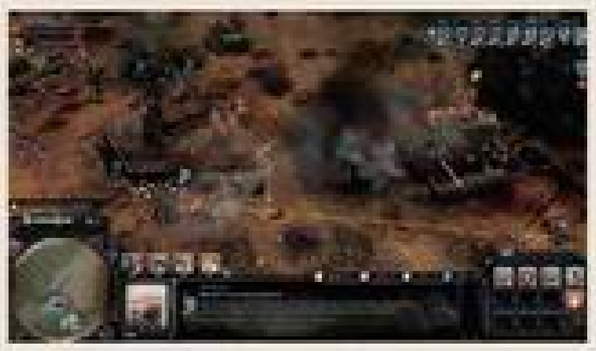
Korora 22 16
Новая инкарнация дистрибутива, несущего Fedora в массы, отлично отлажена и поставляется с целым гарнитуром рабочих столов.

Смартфон SM-Z300F 17
Samsung ставит на свои новые смартфоны открытую ОС Tizen. Эта ОС не только надежная и защищённая, но еще и феноменально экономит заряд батареи.

Dell Chromebook 13 18
Новый номер — новый хромбук. Последнее предложение от Dell покушается на корпоративный рынок, почему и снабжено ПО виртуализации, включая *Citrix* и *VMware*.

› Этот хромбук позволяет запускать разные ОС — виртуально.

Company of Heroes 2 19
Бросайте в бои Восточного фронта бесстрашных русских солдат и молитесь об осечках винтовок Вермахта, или его сдаче, или о чем там еще.

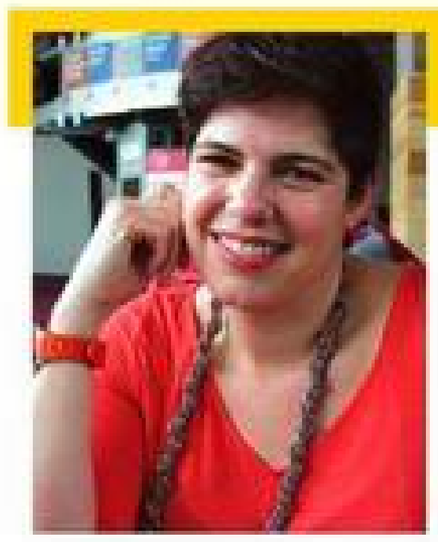


› *CoH2* приходит в Linux; переиграем операцию «Барбаросса».

Сделайте себе медиа-центр на Emby и повергните проприетарных соперников с. 28

Сравнение: Инструменты синхронизации с. 22

Интервью



«Наша миссия — обеспечить всеобщий доступ к знаниям.»

Алексис Росси — о высокой цели Интернет-архива с. 36

На вашем бесплатном DVD

Ubuntu 15.04 32/64-битный, Kodibuntu 14.0 64-битный, Kodi 32-битный, OpenELEC

» Каждый месяц — только лучшие дистрибутивы

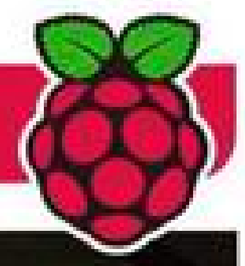
ПЛЮС: Emby и OSMC с. 106

Побалуйте себя и любимых подпиской на LXF!

Доступно в AppStore!

www.linuxformat.ru/subscribe

Пользователям Raspberry Pi



Новости Pi 88
Что творится в Pi Foundation и какие проекты жаждут поддержки на Kickstarter.

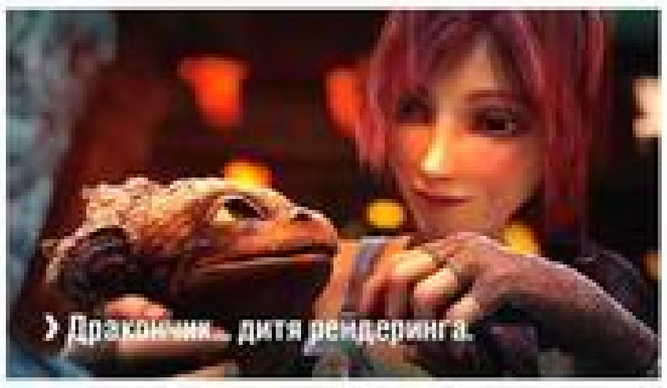
Официальный дисплей 7" 89
Не успел Pi Foundation выпустить чуткий сенсорный дисплей, как мы скогтили его своими грубыми лапами.

Astro Pi Sense HAT 90
Учитесь считывать данные об окружающей среде этим дополнением к Pi — самым небесным.

Сервер печати на Pi 92
Подключите к своему Raspberry Pi принтер и наслаждайтесь сетевой печатью, за мизерную цену.

Ищите в номере

Linux в кино 44
Blender избежал банкротства, увидел свет свободы ПО и намерен штурмовать Голливуд.

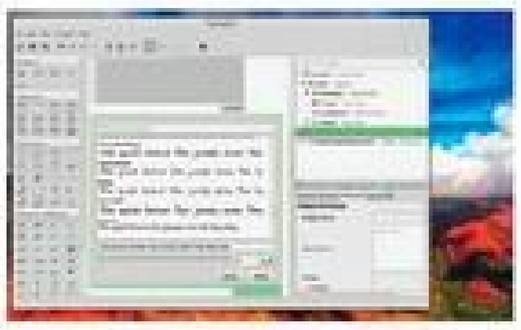

› Дракончик... дитя рендеринга.

Бледный лик Луны 49
Бывалые скитальцы по Интернету выбирают web-браузер *Palemoon*.

Академия кодинга

Системное программирование: UDP 80
Доктор поет лебединую песнь в форме оды UDP, грубоватому собрату TCP. Прощаемся с добрым Доктором — нам будет сильно не хватать его мудрости и эрудиции.

Glade: Календарь на Python 84
Спроектируем элегантный интерфейс пользователя на *Glade*, заставим Python общаться с ним через GObject и перестанем опаздывать на деловые встречи.



Учебники

Основы Linux
Идеал рабочего стола 58
Да, он достижим! За чем бы вы ни гнались, решение найдется. Здесь у нас их семеро.

Мониторинг системы
Conky 62
Выбрав рабочий стол, вытащим на него всевозможную информацию о своей системе.

Периферия в Linux
Печатаем и сканируем 64
Победим *CUPS*, подчиним *SANE* — будем всякие штуки то выводить на бумагу, то считывать с бумаги.

Ключевые утилиты
GNU Coreutils 68
Вникаем в операции над файлами, ссылками, каталогами и межпотоковыми каналами.

Распределенные вычисления
BOINC 72
Употребим простои процессора на службу науке, помогая достойным проектам.

Ubuntu на хромбуке
Crouton 74
Хромбуки накачали достаточно мускулатуры, чтобы вытянуть полноценный дистрибутив.

Автоматизация
ITSM-процессы 76
Замучены инцидентами у пользователей? Свалите нагрузку на агента *OCS Inventory NG*.

Постоянные рубрики

Новости 6
3D-печать проникла в медицину, Mozilla дружит с поисковиками, суперкомпьютеры КНР доминируют, ЕС объединяет документы, 4G плавно переходит в 5, Raspberry Pi дешевеет до $5, диски разменяли 10 ТБ, а Linux и Microsoft нашли общий язык.

Новости Android 20
Android эмулирован в Windows и проник в часы, обновилась IDE разработчиков, а продажи носимых устройств бешено растут.

Сравнение 22
Мы перешерстили инструменты синхронизации файлов в поисках лучшего: *BitTorrent Sync*, *OwnCloud*, *Rsync*, *Syncthing*, *Unison*.

Интервью LXF 36, 40
Алексис Росси мечтает о всемирном архиве Интернета, а **Сергей Бронников** — о живой миграции контейнеров.

Рубрика сисадмина 52
В мире контейнеров м-р **Джолион Браун** чувствует себя Человеком в футляре, и украшает себе жизнь посредством *Kubernetes* от Google.

Ответы 96
ВАШИ ПРОБЛЕМЫ РЕШЕНЫ!
Нейл Ботвик — про загрузку дистрибутива только в командную строку, понижение версии *Wine*, дурное поведение Clonezilla, ошибки при загрузке DVD, добавление обоев, переразбивку на разделы.

HotPicks 100
Отведайте горяченького! Лучшие в мире новинки свободного ПО: *Add-on converter*, *Briss*, *Drawpile*, *Mega Mario*, *Midori*, *Oilwar*, *Solaar*, *sView*, *Terminator*, *Trinity*, *ZSH*.

Диск Linux Format 106
Содержимое двустороннего DVD этого месяца.

Пропустили номер? ... 108
Откуда ж вы узнаете, какой дистрибутив был лучшим в 2015 г., если не из **LXF203**?

Через месяц 112
Автоматизируем свое жилище, налюбуемся видеоплейерами и озаримся свечением Plasma.

В ЭТОМ НОМЕРЕ: Медицина 3D » Mozilla и поисковики » Top-500 » Единая Европа » Три, четыре... пять! » Pi по пятерке » 10-ТБ диск » Кони и лани

НОВЫЕ ТЕХНОЛОГИИ

3D-печать для хирурга

Воссоздается сосудистая сеть человека для подготовки к операции.

» Рубрику готовил АНДРЕЙ ГОНДАРЕНКОВ

Разрыв аневризмы — заполненной кровью выпуклости в стенках кровеносного сосуда — причина почти 500 тыс. смертей в год в мире. Аневризмы, особенно мозговые, возникают среди лабиринта сосудов, где ориентироваться трудно, даже при самых современных технологиях. Врачи и ученые из Jacobs Institute при Университете Буффало совместно с производителем 3D-принтеров Stratasys создали способ всего за 24 часа распечатать полную модель сосудистой сети пациента, предоставив хирургам источник для практической тренировки до реальной операции.

47-летняя Тереза Флинт была среди первых пациентов, которым помогла 3D-модель сосудистой сети. Как пояснил нейрохирург д-р Аднан Сиддики [Adnan Siddiqui], главный врач Jacobs Institute, моделирование процесса на 3D-копии показало, что традиционный метод лечения, с помощью металлической клипсы, просто не будет работать. Печатная модель помогла определить возможные осложнения и разработать оптимальную стратегию лечения.

В бедренную артерию в паху пациента вводится катетер с красителем, который продвигается в артерии головного мозга, и при помощи MRI или компьютерной томографии делаются снимки. Затем на ПК специализированное ПО создает виртуальную модель сосудистой сети, а 3D-принтер Stratasys Objet Eden260VS печатает ее физическую модель в натуральную величину.

Ранее на создание по снимкам модели сосудистой системы из силикона уходило 7–10 дней. 3D-печать ускоряет процесс и снижает затраты с пары тысяч долларов до нескольких сотен. Материал — гибкий фотополимер TangoPlus — имитирует чувствительность человеческой ткани и ее сосудистой структуры, вплоть до поверхностного натяжения кровеносных сосудов и растяжимости васкулатуры.

› **Печатная 3D-модель сосудистой сети пациента. Черная линия — катетер, вводимый хирургом в ходе предоперационной подготовки**

ИТОГИ ГОДА

Новая стратегия Mozilla

Доходы Mozilla прирастают поисковыми запросами.

Некоммерческая организация Mozilla Foundation, создатель браузера *Firefox* и другого ПО, сообщила о своих финансовых результатах за 2014 г. Общая сумма доходов фонда составила $329,5 млн — почти на $15 млн больше, чем в 2013-м. После крупной сделки с Google (в 2012–13 гг. Google был поисковой системой по умолчанию в *Firefox*), бóльшую часть доходов Mozilla получает от договоров с Yahoo (основной поисковый движок *Firefox* в США), Yandex (в России) и Baidu (в Китае). Соглашение с Google обеспечило 88% доходов в 2013 г. (около $ 275 млн) и способствовало росту в 2014 г., когда Mozilla не продлила договор, а заключила сделки с другими поисковиками. В число партнеров вошли также Amazon, eBay и Bing. «Мы продолжим инвестиции в партнерства, расширяющие наше влияние на мир», отмечает Mozilla. Но финансовый итог новой стратегии выявится лишь после очередного отчета Mozilla в ноябре 2016 г.

Финансовый итог выяснится лишь после отчета в ноябре 2016 г.

После дебюта в 2002 г., когда альтернатив *Firefox* практически не было, браузер от *Mozilla* долго был любимцем web-серфистов, но сдал позиции браузеру *Chrome* от Google. С ростом популярности интернет-серфинга с мобильных устройств Mozilla вышла и на этот сегмент рынка, с браузерами для Android и iOS, а также мобильной Firefox OS на базе Linux для планшетов и смартфонов. Внедриться в этот сегмент рынка оказалось трудно — разработка Firefox OS прекращена (но Mozilla продолжит ее продвижение на устройства IoT, в частности, «умные телевизоры»).

Источники: www.computerworld.com

ГОНКА ЛИДЕРОВ

46-й суперкомпьютерный

Китайский кластер Tianhe-2 доминирует в списке Top-500 шестой год подряд.

Китай продолжает доминировать в списке 500 лучших суперкомпьютеров мира. В последнюю, 46-ю редакцию вошли несколько новых китайских систем. Tianhe-2 шестой год подряд занимает в Top 500 первое место, и крайне маловероятно, чтобы в ближайшее время конкуренты смогли потеснить этот кластер, размещенный в Национальном суперкомпьютерном центре в Гуанчжоу.

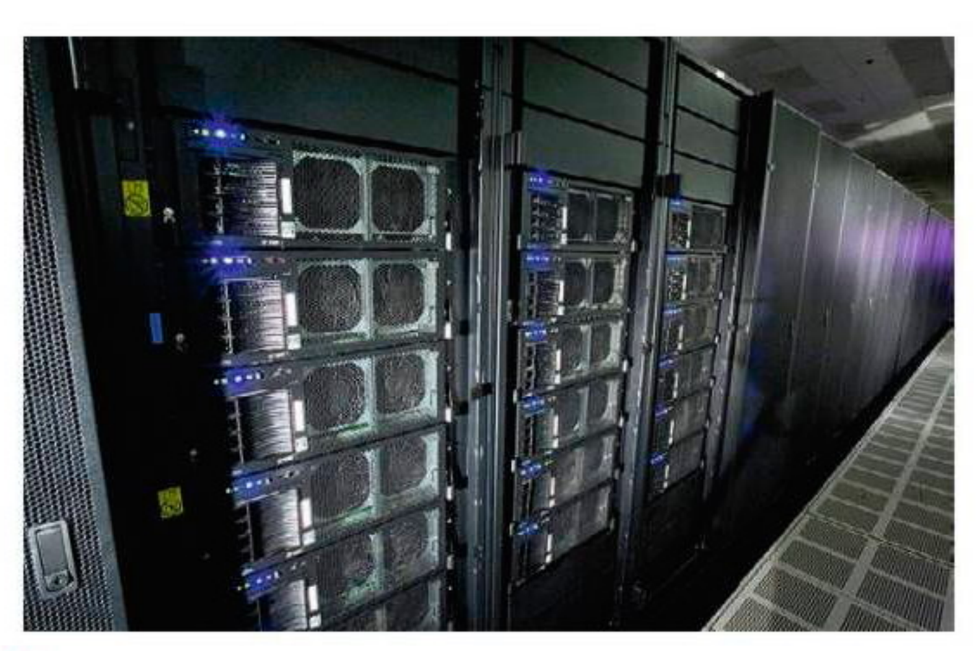
Производительность Tianhe-2 в тесте Linpack оценена как 33,86 петафлоп/с, а за ним следует принадлежащая Министерству энергетики США система Titan Cray XK7 (17,59 петафлоп/с).

Изменения в верхнем сегменте списка незначительны. Только две новые системы вошли в Top-10: суперкомпьютер Trinity, созданный Cray и развернутый Лос-Аламосской и Сандийскими национальными лабораториями Министерства энергетики США, и система Hazel-Hen компьютерного центра Hochstleistungsrechenzentrum (Штутгарт, Германия), также от Cray.

Число принадлежащих Китаю компьютерных суперсистем практически утроилось, а показатели США упали до самой низкой точки с момента создания Top-500 в 1993 г. Сейчас в списке наличествуют 199 систем из США, тогда как еще полгода назад их было 231.

Россия представлена в списке семью системами. Самый быстрый российский суперкомпьютер, входящий в Top-500 — установленный в МГУ «Ломоносов-2», с производительностью 1,849 петафлоп/с, занял 35-е место.

И еще несколько цифр из последнего Top'а. По странам количество суперсистем распределено следующим образом: США — 199, Китай — 109, Япония — 37, Германия — 32, Великобритания — 18, Франция — 18, Индия — 11, Южная Корея — 10, Россия — 7, Саудовская Аравия — 6, Швейцария — 6, Бразилия — 6, Канада — 6, Польша — 6, Австралия — 5. Суперкомпьютеры работают под управлением Linux — (494, 98,8 %) и Unix — (6, 1,2 %). Суперсистем с гибридными ОС, а также Windows и BSD в этой редакции нет. При этом для 66,2 % систем с Linux используемый дистрибутив не назван, на 10,8 % используется CentOS, 8,8 % — SUSE, 8,6 % — Cray Linux, 3,6 % — RHEL, 0,6 % — Scientific Linux, 0,4 % (в том числе Tianhe-2) — Ubuntu Kylin.



› Tianhe-2 содержит 16 тыс. узлов с двумя процессорами Intel Xeon IvyBridge и тремя процессорами Xeon Phi — в сумме это более трех млн процессорных ядер.

ОТКРЫТИЕ ЕВРОПЫ

Данные всех стран, объединяйтесь!

ЕС объединит общеевропейские открытые данные в новом портале.

В ходе прошедшего в Люксембурге European Data Forum 16 ноября Еврокомиссия запустила бета-версию European Data Portal — общеевропейского портала, который станет центральным шлюзом к данным, опубликованными правительствами стран Европы (как членов, так и не членов ЕС). Проект реализует консультационная фирма Capgemini, Open Data Institute и ряд других академических и корпоративных учреждений. Сейчас доступ выдан к более 240 тыс. пакетов данных из 34 стран в 13 различных категориях (сельское хозяйство, транспорт, наука, юстиция, здравоохранение и т. д.). Функция поиска на нескольких языках обеспечивает индексацию данных, а также и сбор метаданных. «Единый web-портал позволит избежать ненужного дублирования дорогостоящей инфраструктуры. Преимущество сбора чисто метаданных состоит в том, что ими легко манипулировать, создавать визуальные представления и наносить на карту. Вы не отягощаете себя огромными хранилищами, и в то же время обновление таких метаданных — непрерывный процесс», поясняет руководитель проекта Венди Каррара [Wendy Carrara], директор и главный консультант Capgemini. Запуск портала преследует две цели. Во-первых, выдается централизованный доступ к документам, сохраненным в национальных репозиториях открытых данных по всей Европе. Во-вторых, у правительства страны появляется стимул опубликовать документы определенной тематики, уже размещенных на портале другими странами.

Согласно экономическому исследованию, проведенному командой European Data Portal, масштабы рынка открытых данных за ближайшие четыре года возрастут на 36,9 %, и к 2020 г. достигнется объем €75,7 млрд. Повторное использование открытых данных — одно из направлений, способствующих дальнейшему росту экономики. Его преимущества разнообразны и варьируются от улучшения деятельности государственных учреждений и экономического роста в частном секторе до расширения социальной защищенности.

› European Data Portal поможет предприятиям, способным извлечь экономическую выгоду из публичных данных, применив их для создания новых продуктов и принятия решений

Источники: top500.org; joinup.ec.europa.eu

МОБИЛЬНЫЙ ИНТЕРНЕТ

Общедоступная сеть 5G

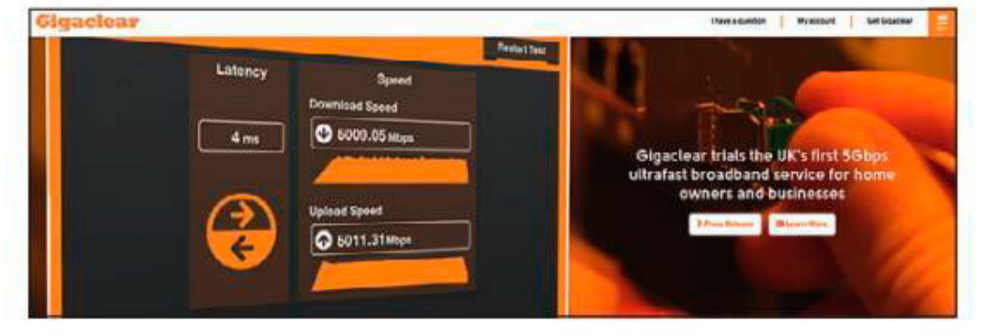
Британский провайдер Gigaclear предлагает широкополосную сеть со скоростью доступа 5 Гб/с.

В ноябре минувшего года британский провайдер широкополосных телекоммуникационных систем, компания Gigaclear, анонсировал планы предоставления пользователям малых и средних компаний (SMB), а также домашним пользователям интернет-соединения на скорости 5 Гб/с. Это первая в стране общедоступная сверхскоростная широкополосная сеть пятого поколения (5G). С января 2016 г. частным и бизнес-пользователям доступны услуги H5G и B5G соответственно.

В отдельных районах Великобритании Gigaclear уже предоставляет сервис 1 Гб/с; новая сеть 5G уверенно затмевает предлагаемые кабельным гигантом Virgin Media 200 Мб/с; а поскольку у компании Google Fibre (основной продукт которой — проект по созданию инфраструктуры сети широкополосного доступа к Интернету с помощью оптоволоконной связи) «нет серьезных планов» по организации подобного сервиса в UK, конкуренты у Gigaclear отсутствуют.

Возможности, возникающие у пользователей сети Gigaclear 5G, впечатляют: например, скачивание видеофайла размером 1,5 ГБ займет всего 4–6 секунд. Ложкой дегтя, однако, является цена: стоимость пакета business составит £399 в месяц, premium — £1500, т. е. для домашних пользователей ежемесячная абонентская плата сопоставима со стоимостью нового ноутбука.

Мэттью Хеэ [Matthew Hare], руководитель Gigaclear, пояснил: «Мы предлагаем нашим клиентам шанс получить доступ к совершенно феноменальным широкополосным скоростям. Это сервис премиум-класса, за свои деньги наши клиенты смогут получить самую высокую скорость и качество Интернет-соединения, которые сегодня предлагаются в стране. Недалек тот день, когда сервис 5 Гб/с станет широко востребованным, поскольку связанность становится всё более важной частью повседневной жизни. Запуск пробной версии этого сервиса показывает нашим клиентам, что уже сегодня мы предлагаем сеть завтрашнего дня».



› Подключение домашних абонентов и офисов SMB к оптико-волоконной сети Gigaclear 5G позволит значительно сократить время, требуемое на загрузку больших файлов и массивов данных.

ЯГОДКА-МАЛИНКА

Дешевле только даром!

Raspberry Pi Zero — компьютер стоимостью $5.

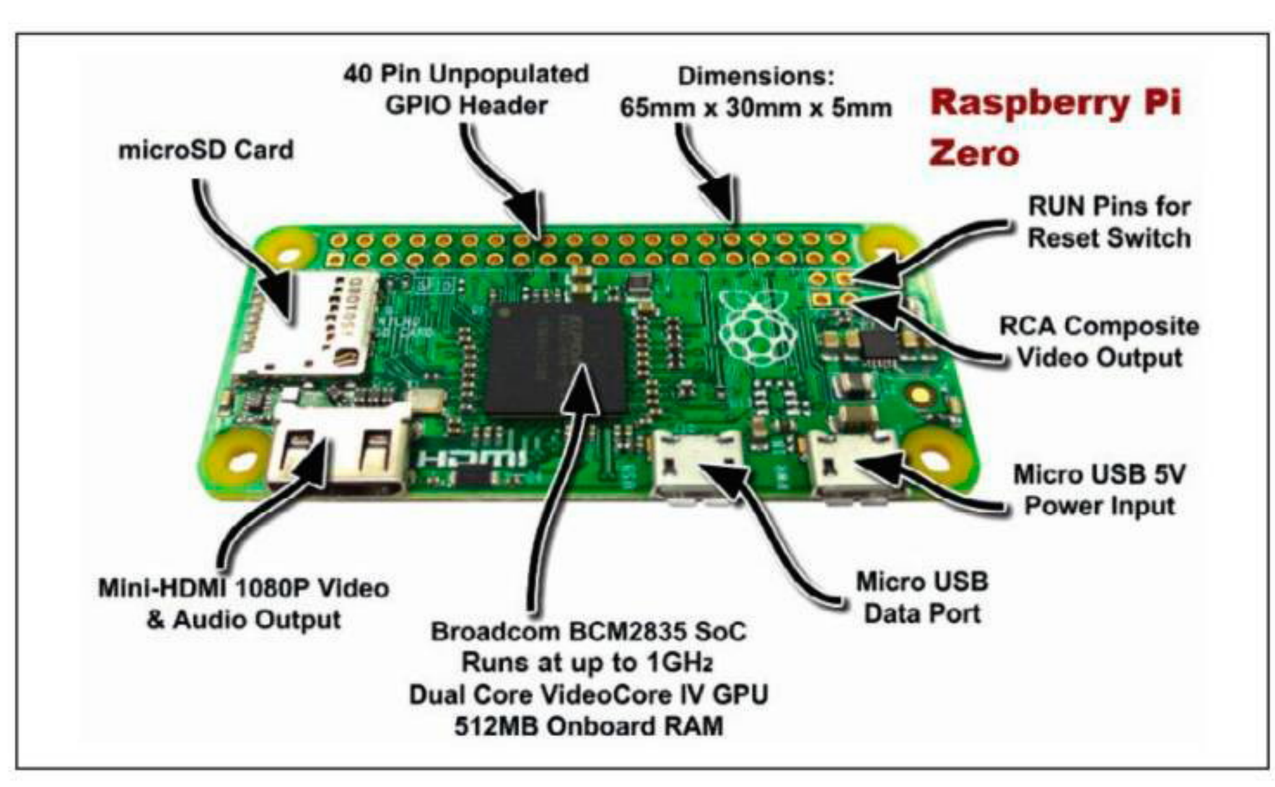
Несколько лет поработав над снижением стоимости, основатель Raspberry Pi Foundation Эбен Аптон [Eben Upton] пришел к заключению, что дальше тут двигаться некуда: цена его последнего творения всего $5, и вряд ли возможно создать компьютер за меньшую стоимость.

Несмотря на ошеломляюще низкую цену, Raspberry Pi Zero обладает многими из функций Raspberry Pi 1 и работает приблизительно на 40% быстрее своего прародителя; у него лишь отсутствуют некоторые порты.

В центре печатной платы, размеры которой 65×30 мм — процессор Broadcom BCM2835, такой же, как у Raspberry Pi 1, с 1-ГГц ядром ARM11 (у процессора предыдущей «малинки» — 700 МГц) и графическим ускорителем Broadcom VideoCore IV 250 МГц. Плата содержит 512 МБ ОЗУ, операционная система — традиционная для Raspberry Pi специализированная версия Debian GNU/Linux, Raspbian — загружается с карты памяти microSD. Имеются слот mini-HDMI для вывода видео 1080p и слоты micro-USB для данных и питания.

Как и у ее предшественников, на плате предусмотрена возможность расширения: ее 40-контактный слот GPIO имеет идентичные контакты, однако изначально не распаян — в случае необходимости, пользователю придется приобрести и распаять разъем самостоятельно. Аналогичная ситуация с композитным видеовыходом.

Для сравнения, Raspberry Pi 2 модели B располагает дополнительной памятью, четырехъядерным процессором, входом/выходом аудио, тремя дополнительными портами USB и возможностью прямого подключения камеры и дисплея.



› Ультра-малая и ультра-тонкая плата Raspberry Pi Zero с размерами 65×30×5 мм — самый маленький форм-фактор Raspberry Pi на современном рынке.

Источники: www.theinquirer.net; www.computerworld.com

ДИСКОТЕКА

10 ТБ на гелии

Создан первый жесткий диск емкостью 10 ТБ с гелиевым наполнением.

Western Digital представила первый HDD емкостью 10 ТБ, заполненный гелием для уменьшения трения. HDD 3.5-in Ultrastar He10 компания создала на III поколении технологии WD HelioSeal Platform, с меньшим, чем у любой предыдущей модели, потреблением мощности на 1 ТБ емкости. Новый HDD применяет типовую перпендикулярную магнитную запись (PMR); емкость 10 ТБ достигается за счет семи пластин с поверхностной плотностью записи 816 Гб на квадратный дюйм. Ultrastar He10 на 7200 об./мин. поставляется с интерфейсами SATA 6 Гб/с или SAS 12 Гб/с, с максимальными показателями на чтение/запись 249 Мб/с и 225 Мб/с соответственно.

Мощность Ultrastar He10 на 25% больше, чем у 8-ТБ HDD предыдущего поколения Ultrastar He8, а его показатель потребления Вт/ТБ на 56% меньше, чем у традиционных HDD, заполненных воздухом. При простое диск потребляет лишь 0,5 Вт/ТБ; при работе через интерфейс SATA — 6,8 Вт, а с SAS — 9,5 Вт. Показатель среднего времени наработки на отказ 2,5 млн MBTF (mean-time-between-failures), с пятилетней гарантией.

WD позиционирует свой новый HDD как "drop-in ready", т.е. для работы на сервере или в составе массива не требуется дополнительного ПО — в отличие от дисков большой емкости на технологии записи с перекрытием дорожек SMR (shingled magnetic recording). Диски емкостью 10 ТБ могут стать ключевыми компонентами видео-, фото-, бизнес- и других облачных приложений, публичных и приватных.

Цену нового диска, предназначенного для корпоративного использования, WD пока не называет.

ГИБРИДНЫЕ СИСТЕМЫ

Экзамен MCSA: Linux on Azure

Linux Foundation и Microsoft совместно сертифицируют специалистов.

Границы между открытым и проприетарным ПО размываются на глазах. Все больше организаций создают даже технологии для внутреннего потребления методами Open Source. Еще недавно в это никак не верилось, но сейчас в их число входит Microsoft. Участие в проекте Node.js, Core Infrastructure Initiative, других «совместных проектах [Collaborative Projects]» от Linux Foundation и недавнее партнерство с Red Hat и SUSE обнаруживает практичный подход Microsoft к созданию новых технологий и поддержке своей обширной клиентской базы: компания, например, открыла часть кода .NET и ключевых частей браузеров *Internet Explorer/Edge*, и использует Linux в платформе Azure Cloud Switch.

Декабрь прошлого года увидел очередной шаг в этом направлении: в рамках нового соглашения Microsoft предложила своим клиентам сертификационный экзамен "Microsoft Certified Solutions Associate (MCSA): Linux on Azure". Для получения сертификата надо сдать 2 экзамена: LFCS (Linux Foundation Certified System Administrator) от Linux Foundation и 70-533 (Implementing Microsoft Azure Infrastructure Solutions) от Microsoft. «Успешная сдача докажет способности специалиста проектировать, внедрять и обслуживать сложные облачные Linux-среды, целиком задействуя преимущества Microsoft Azure», говорится в пресс-релизе Microsoft. Технический директор Azure Марк Руссинович [Mark Russinovich] отметил, что если рынок требует использовать Linux, Microsoft выйдет за рамки одной ОС, не игнорируя столь выгодный бизнес. Популярность GNU/Linux в Azure растет. Пока в Azure поддерживается 8 дистрибутивов GNU/Linux: Ubuntu, CentOS, CoreOS, Debian, Oracle Linux, SUSE (Enterprise и open) и Red Hat (с RHEL). LXF

Источники: www.computerworld.com; www.linuxfoundation.org; www.theinquirer.net

Новости короткой строкой

» ThinkPad T400 Libreboot со свободным дистрибутивом Trisquel на борту и прошивкой Libreboot вместо штатной BIOS — уже четвертый ноутбук с сертификатом Фонда СПО по приватности и свободе пользователей.
Источник: www.fsf.org

» Программа Mozilla Open Source Support финансирует ПО сетевого мониторинга безопасности *Bro*, web-фреймворк *Django*, систему управления исходниками *Mercurial*, сервис подготовки и размещения документации *Read The Docs*, движок web-форумов Discourse и редактор кода *CodeMirror*.
Источник: blog.mozilla.org

» Microsoft опубликовала на GitHub под именем ChakraCore движок Chakra на JavaScript (из Edge, Azure DocumentDB, Windows 10 IoT Core и Cortana).
Источник: blogs.windows.com

» 7 декабря 2015 г. исполнилось 11 лет со дня выпуска *Thunderbird*. Число его активных пользователей — более 10 млн, а общая аудитория —25 млн.
Источник: blog.mozilla.org

» Начато публичное бета-тестирование Let's Encrypt — удостоверяющего центра под контролем сообщества («золотой» спонсор проекта — Facebook).
Источник: letsencrypt.org

» С марта 2016 г. Google прекращает выпуск обновлений браузера *Chrome* для 32-битных версий Linux, сохраняя возможность сборки *Chromium*.
Источник: groups.google.com

» *Enlightenment 0.20* ввел поддержку *Wayland*, инфраструктуры управления экранами и микшером звука и модуль Geolocation, улучшил работу с FreeBSD.
Источник: phab.enlightenment.org

» Аудит Федерального управления по ИБ (Германия) подтвердил надежность шифрования *TrueCrypt*. Выявленные проблемы легко устранимы.
Источник: sseblog.ec-spride.de

» Ядро Linux 4.1.15 показано в фильме *Терминатор* в эпизоде демонстрации интерфейса киборга T-800.
Источник: www.reddit.com

» В SUSE Enterprise 12 SP1 — интеграция с Docker, идентификация по Shibboleth, сжатие данных на Btrfs, поддержка процессоров Intel (Xeon v4 E3, E5 и E7, D-1500 и Core VI поколения.
Источник: plus.google.com

ВТР: Трансфузиология

«**ВТР: Трансфузиология**» — программный продукт, предназначенный для информатизации задач, связанных с забором, хранением и использованием донорской крови, ее компонентов и кровезаменителей, а также специальных лечебных процедур, выполняемых в отделении переливания крови.

Преимущества системы «ВТР: Трансфузиология»

- Разработка на базе свободного программного обеспечения (ПО с открытым кодом), что соответствует политике импортозамещения.
- Трехуровневая архитектура с тонким клиентом.
- Возможность интеграции с защищенным хранилищем персональных данных.
- Возможность проследить трансфузионную информацию от донора до реципиента и обратно.
- Предотвращение ошибок персонала на пути «от вены до вены» путем обеспечения информационного сопровождения и контроля.
- Эффективная организация планирования, распределения и подбора компонентов крови, анализа результатов и осложнений, проведения научных работ.
- Автоматическая загрузка результатов проведенных исследований из ЛИС (клинико-диагностической и бактериалогической лабораторий и лаборатории цитогенетики).
- Автоматическое получение заявок на кровь и лечебные процедуры из МИС и передача обратно данных по выданным компонентам крови с прикрепленной этикеткой и результатов проведения лечебных процедур.
- Возможность проведения лечебных процедур, фиксация данных о полученных биоматериалах в рамках процедуры, печать процедурных листов.
- Автоматическая выгрузка данных по пациентам и отводам в ЕДЦ.
- Возможность оповещения доноров о приходах на сдачу крови и рассылки другой информации через СМС.
- При выездах сохранение связи с локальной БД путем использования «веб-клиента».

ВТР

ООО «Высокотехнологичные решения»
Санкт-Петербург, Пионерская ул., 30, лит. В, офис 301
(812) 644-51-10

www.hitsl.ru

Обзоры

Новинки программного и аппаратного обеспечения в описании наших экспертов

АЛЕКСЕЙ ФЕДОРЧУК
Тэг <сарказм> по умолчанию, смайлики по вкусу.

Полет продолжается

DragonFly BSD, отпочковавшись от FreeBSD 4.4 летом 2003 года, внесла в мир UNIX'а принципиальные новшества: распараллеливание легковесных нитей ядра (LWKT), перенесение драйверов устройств в пространство памяти пользователя, механизм сообщений и их очередей вместо традиционных системных вызовов, файловую система HAMMER. Эти новшества могли бы заиграть яркими красками на настольных ПК с широким распространением многоядерных процессоров, что и задумывалось изначально. Однако доступ к прелестям ОС для простых применителей долго затруднялся неразвитостью пользовательской инфраструктуры, в частности, доступа к приложениям.

Ситуация стала меняться в 2012 г., когда в рамках проекта FreeBSD разработали систему управления пакетами *Pkg*. Она мгновенно инкорпорировалась в DFBSD и получила в ней официальный статус и собственные репозитории даже раньше, чем в материнской ОС. Сперва она требовала весьма кропотливой настройки, но ныне (в версиях 4.4.X) справно работает «из коробки». Притом чисто консольную систему DFBS, получаемую после установки через *BSD Installer* (кстати, очень простой), можно превратить в полноценный десктоп буквально двумя командами: pkg xorg и pkg <имя любимой рабочей среды>, например, cinnamon, mate или xfce.

Так что применителям, заскучавшим в стремительно стандартизируемом и заорганизованном мире Linux'а, есть куда двигаться.

alv@posix.ru

Сегодня мы рассматриваем:

Tails 1.6 12
Этот дистрибутив сохранит вашу конфиденциальность онлайн. Для некоторых — скажем, журналистов или активистов — анонимность может стать вопросом жизни или смерти.

MySQL Fabric 1.6.2 14
Расширяемая платформа с открытым кодом для работы с фермами серверов *MySQL* упрощает операции с базами данных, гарантируя высокую доступность и масштабируемость.

Korora 22 16
Korora — ремикс, который взял новейший дистрибутив Fedora 22 и привел его к готовности для повседневного пользователя. Идеальный дистрибутив для исследования.

Смартфон SM-Z300F ... 17
Tizen — не просто «еще одна» операционка, но перспективная ОС для мобильных устройств, носимой электроники и предметов «умного дома». Перед вами — смартфон от Samsung на этой ОС, поражающий долгожительством батареи.

Dell Chromebook 13 ... 18
Хромбук из высококачественных материалов претендует на звание корпоративного. И впрямь, поддержка целого ряда настольных программ виртуализации позволит запустить даже Windows.

Company of Heroes 2 ... 19
Игра предоставляет в ваше распоряжение почти безграничные человеческие ресурсы, которые можно бросить в сражение, и сильно напоминает события новейшей истории. Соблазн изменить весь ход мировой войны не поддается описанию...

› **Хромбук выглядит по-корпоративному солидно.**

MySQL Fabric 1.6.2

› **Решения *MySQL Fabric 1.6.2* способны обеспечить и доступность, и масштабируемость, и обе сразу.**

Company of Heroes 2

› **В огне и дыму сражений Второй мировой войны, Советская армия отчаянно рвется к Берлину.**

Сравнение: Синхронизация файлов с. 22

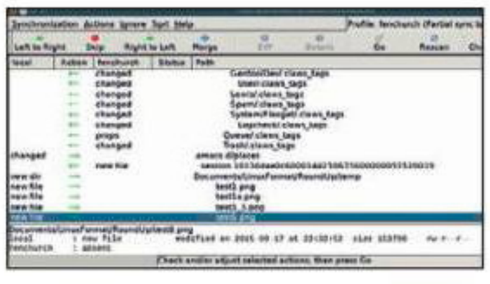
BitTorrent Sync

OwnCloud

Rsync

Syncthing

Unison



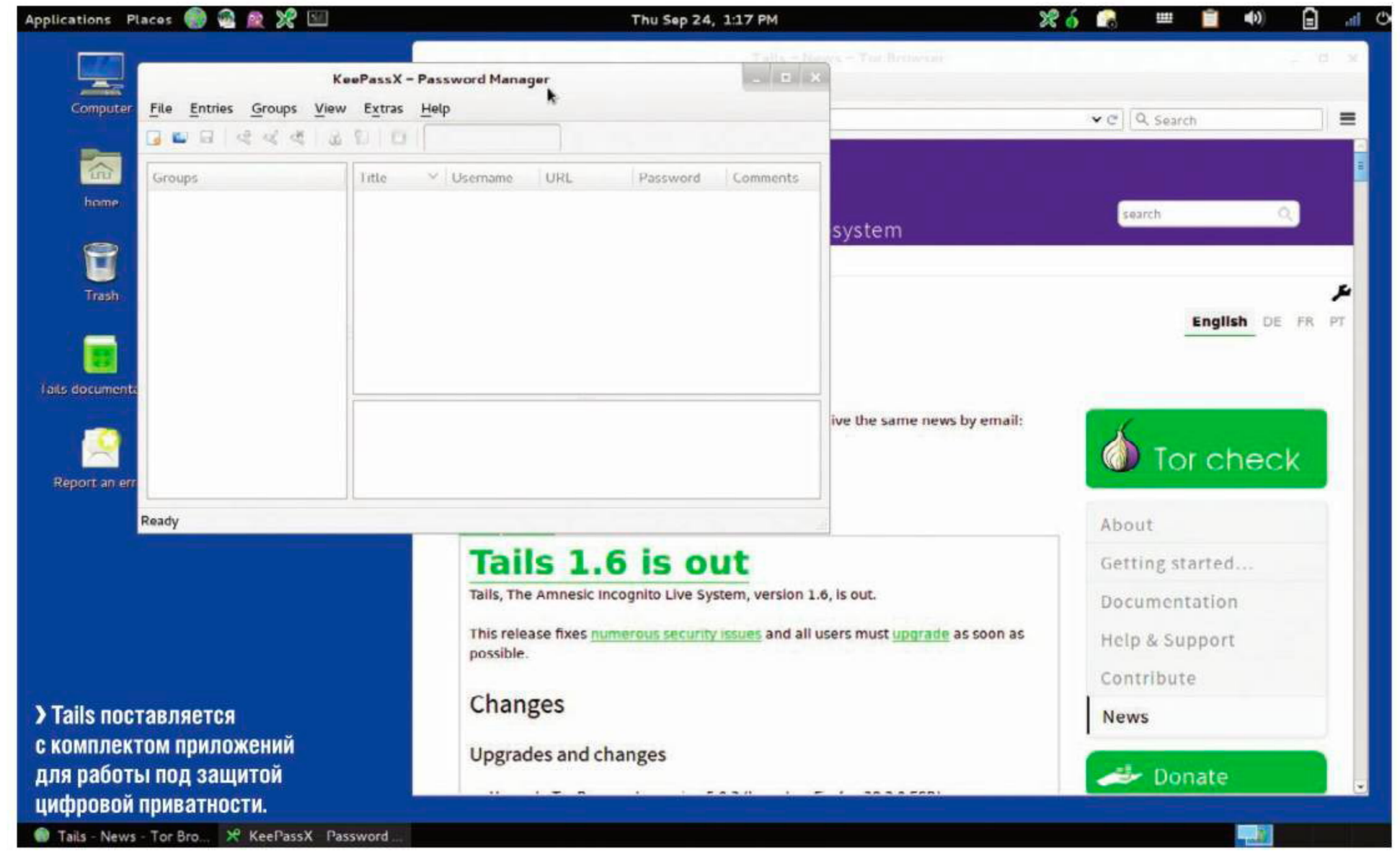
Конечно, есть всякие сервисы типа Dropbox; но что если вы хотите обладать полным контролем над своими файлами и хранить их локально, и тем не менее с доступом со всех своих устройств?.. Вот вам решения.

Tails 1.6

Прикинувшись секретным агентом, **Лес Паундер** улетает из-под радара в Париж, затем в Берлин. Спасет ли его дистрибутив с упором на приватность?

Вкратце

» Tails — дистрибутив Linux, который защищает вашу конфиденциальность онлайн с высокой целью разрушения барьеров цензуры. Работая как Live DVD или с флешки, Tails может использоваться на любом типе компьютера, шифрует ваши данные и сохраняет их на постоянный раздел для удобства транспортировки. Tails также поставляется с целым набором спецприложений безопасности.



› Tails поставляется с комплектом приложений для работы под защитой цифровой приватности.

Защита личности онлайн — насущная забота не для каждого, но для некоторых — скажем, журналистов или активистов — анонимность может стать вопросом жизни или смерти. В «арабской весне», в серии демократических восстаний в 2011 г. на Ближнем Востоке и Северной Африке сети по типу Tor применялись для доступа к социальным медиа-платформам с сообщениями в реальном времени. Tails (The Amnesiac Incognito Live System) — дистрибутив Linux на базе Debian — обеспечит безопасное подключение к Интернету и направлен на сохранение конфиденциальности. Tails доступен в виде ISO-образа и может быть записан на USB-флешку или SD-карту, или на DVD в качестве защищенной платформы. Tails может использовать постоянный раздел для хранения данных, также шифруемый для безопасности.

При загрузке Tails можно открыть расширенное меню, имеющее опцию создания пароля администратора, полезного для дальнейшей настройки, а также подмены MAC-адреса вашего устройства, чтобы избежать вашей однозначной идентификации. Рабочий стол Tails можно замаскировать под Windows 8, чтобы не выделяться, работая в кофейне. По умолчанию, впрочем, Tails 1.6 использует окружение рабочего стола Gnome и поставляется с обычными приложениями Gnome, такими как файловый менеджер *Nautilus* и *NetworkManager*.

Свойства навскидку

Управление идентификацией

Панель управления Vidalia изменит вашу «идентичность», исходя из точки выхода в сеть Tor, к которой Tails подключается автоматически.

Камуфляж

По умолчанию используется среда Gnome, но Tails умеет маскироваться под Windows 8, что удобно для ситуации тайного просмотра в кофейне.

Безопасное соединение

Главная особенность Tails 1.6 в том, что он обеспечивает автоматическое подключение к сети Tor через Ethernet или Wi-Fi. Это отличное новшество: когда мы тестировали Tails 1.2.3 [см. Сравнение **LXF196**, стр. 30], он для этого потребовал ручной настройки.

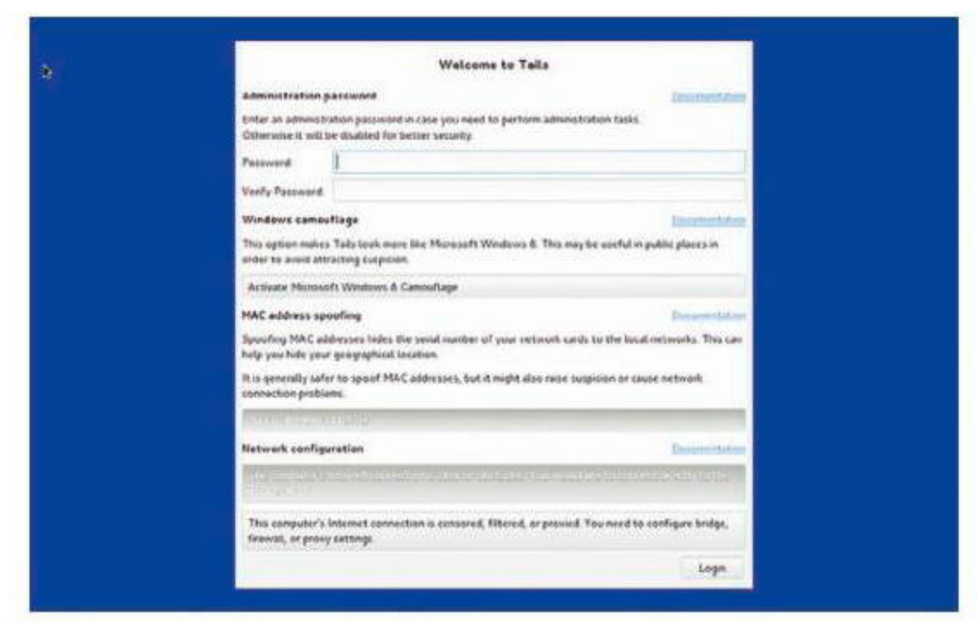
Tails использует *Tor Browser*, безопасную версию *Firefox* от Mozilla, с кнопкой Tor и расширениями HTTPS Everywhere, NoScript и Adblock Plus. Браузер не запустит JavaScript, Java, Flash или другие фоновые приложения, способные на утечку данных. На самом деле, браузер настолько блюдет безопасность, что помешал нам загрузить скриншоты на Flickr, отключив доступ к основной файловой системе.

Но Tails — это не просто Tor для чайников, а полноценный дистрибутив с набором инструментов. Кроме типовых приложений, имеющихся во многих дистрибутивах, он включает *MAT* (Metadata Anonymisation Toolkit), использующийся для сканирования файлов в поисках информации, способной идентифицировать или отследить по файлу владельца. Есть также *Electrum Bitcoin Wallet*, чтобы обезопасить вашу криптовалюту слоем шифрования; для его использования надо установить Tails или создать постоянный том для кошелька. *Traverso* — приложение для многодорожечной аудиозаписи. *Gobby* — коллективный редактор, в основном применяемый удаленными командами, желающими совместно работать над проектом. *GtkHash* создает уникальный хэш для файла — хэш вместе с файлом передается адресату и используется для идентификации при обнаружении вскрытия при передаче.

Если вам действительно понадобится «небезопасный» браузер, в Tails есть и такое приложение, но оно будет напоминать, что ваши действия уже не анонимны и ваша безопасность скомпрометирована. Все приложения работают с AppArmor, снижая риск применения эксплойта или взлома приложений. Скрытое в терминале популярное приложение *aircrack-NG* проводит аудит безопасности сети, пытаясь взломать ключи, которые ее защищают.

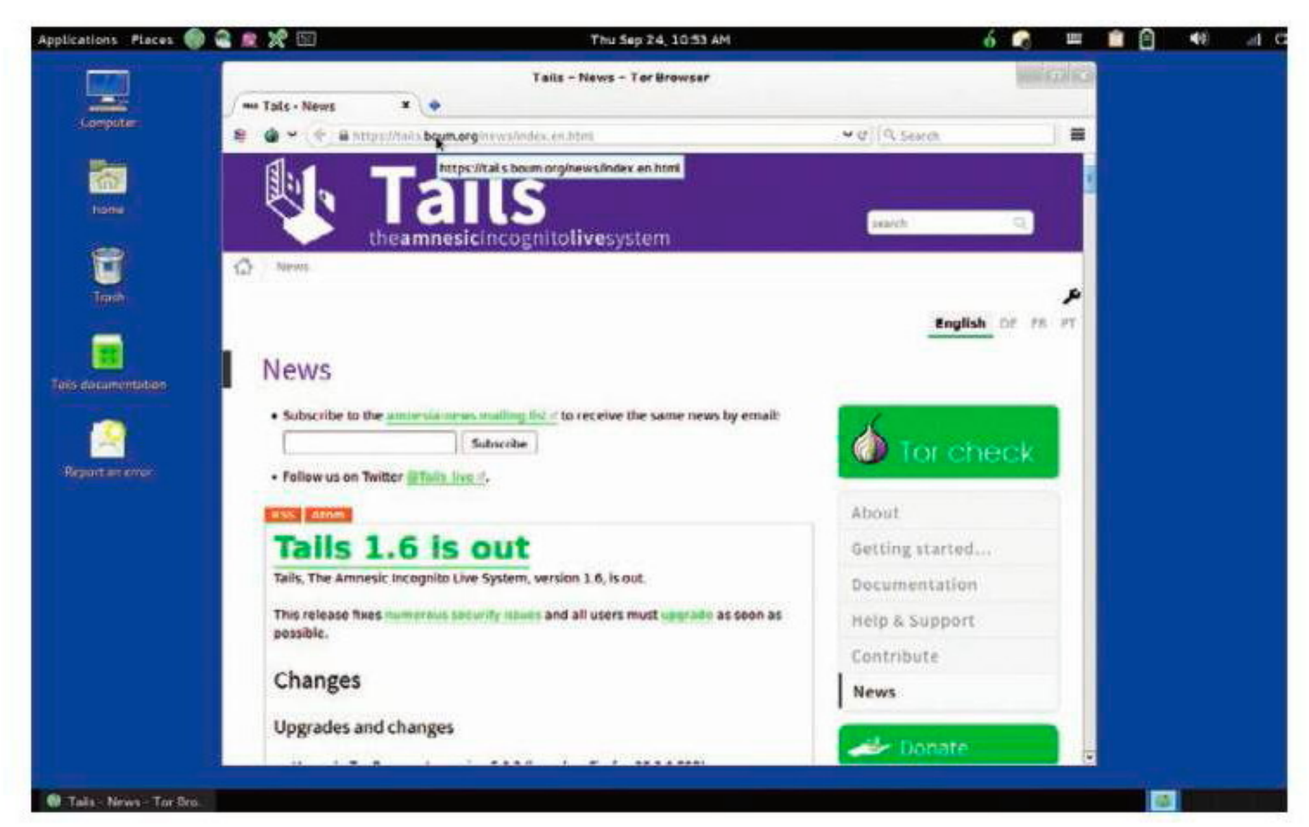
Автоматическое подключение к сети Tor через Ethernet или Wi-Fi.

На нашей тестовой машине, с четырехъядерным процессором Intel 2,4 ГГц и 4 ГБ ОЗУ, рабочий стол был отзывчив и имел очень малую задержку, ожидаемую при работе с USB-устройством. Задержку удается снизить установкой Tails на быструю SD-карту. Tails можно запустить на компьютере с 1 ГБ ОЗУ, что делает его идеальным дистрибутивом для старого нетбука на выезде,



› Расширенные параметры настройки безопасности хорошо продуманы.

› *Tor Browser* основан на *Firefox*, но в целях безопасности не запускает JavaScript, Java или Flash.



при общедоступном интернет-соединении, которое может попытаться проверить ваши данные.

Удобная функция Tails заключается в том, что можно стереть информацию из ОЗУ, которая при обычной работе сохраняется и специалист может ее восстановить.

Tails автоматически подключится к сети Tor и установит ваш IP-адрес в зависимости от того, где вы выходите в сеть. В наших тестах мы вышли через оконечные узлы Tor в Германии и Франции, но при необходимости можно переменить свою «идентичность» с помощью панели управления *Vidalia* через значок Tor в правом верхнем углу экрана. На этой панели управления можно также посмотреть состояние сети Tor и проверить текущую пропускную способность.

Поговорим о Tor

Можно счесть Tails идеальным дистрибутивом для обмена файлами, но, хотя и можно направлять торренты через сеть Tor, этот вид деятельности не одобряется, поскольку тогда тратится огромная часть полосы пропускания, предназначенная для других потребностей.

Tor, однако, не является всеохватывающим решением для цифровой анонимности. В принципе, ни один сервис не защитит вас полностью: были сообщения о компрометации властями выходных узлов и узнаванию пользователей по их характерной активности. Для истинной защиты требуется совокупность нескольких служб и смена своих привычек.

Известный хакер Freakyclown говорит: «Никогда не полагайтесь на что-то одно, чтобы сохранить анонимность. Луковица сделана логотипом Tor не просто так. Используйте Tor как луковую шелуху — слой за слоем. Шифрование — еще одна вещь, которую люди умеют плохо, но принимаемые вами меры зависят от того, что вы намерены сделать. Незачем лететь в Таиланд с краденым ноутбуком и тырить Wi-Fi из соседнего кафе, чтобы просмотреть секретную вики в подсети, но если вы ищете на сервере НАСА доказательства существования пришельцев, не делайте этого из дома ваших родителей!»

В Tails 1.6 видно несколько изменений, в основном обновления для *Tor Browser*, который теперь основан на *Firefox* версии 38. Обновлена анонимная оверлейная сеть I2P (Invisible Internet Project), применяемая приложениями для безопасной отправки информации. Есть обновления и исправления для SSL-сертификатов и bind9, используемых службами DNS.

В нашем сравнении конфиденциальных дистрибутивов [см. стр. 30 **LXF196**] Tails оказался на втором месте позади JonDo Live-DVD. С выходом Tails 1.6 это решение остается в силе. Да, Tor быстрее JonDo при подключении к сети, и Tor лучше при рандомизации выбора узла; но по сути, Tails — и в этой последней версии Tails.

Tails предлагает очень простой вариант анонимности онлайн и предлагает достаточно полный набор приложений для многих повседневных задач. Хороший выбор по данному прямому назначению: помочь свободе информации. LXF

LINUX FORMAT Вердикт

Tails 1.6

Разработчик: Tails
Сайт: tails.boum.org
Лицензия: GNU/GPLv3

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

» Его не следует использовать как единственное средство анонимности, но это отличное начало для защиты вашего онлайн-присутствия.

Рейтинг 8/10

MySQL Fabric 1.6.2

Лада Шерышова представляет инновационный продукт от Oracle — «управляющего» для серверных ферм.

Вкратце

› ***MySQL Fabric* — инновационный продукт, который позволит упростить работу с группами баз данных *MySQL*, повысить готовность бизнес-приложений и автоматически масштабировать систему.**

Деятельность предприятий и организаций все больше требует непрерывности выполнения бизнес-задач: их остановка бывает чревата тяжелыми финансовыми и имиджевыми последствиями, а то и катастрофой. Высокая доступность (ВД) [High Avaliability] — это способность системы обеспечить бесперебойное обслуживание. В ВД особенно нуждаются те, чьи вычислительные системы (ВС) должны работать круглосуточно, без перерывов свыше нескольких минут. Доступность сервиса обычно измеряют количеством девяток. Если сервер работает 90% времени, его доступность — одна девятка. Золотой стандарт "High" — «пять девяток»: 99,999%. Высокодоступная ВС имеет право на перебои примерно пять минут в год! И это с учетом плановых простоев. Другое важное требование к ВС — масштабируемость [scaling]. Любая современная крупная система — будь то сайт или облачное приложение — ежедневно обрабатывает до десятков миллионов запросов, обычно имеет сложную структуру и хранит немало данных. В таких ВС доминируют запросы на получение информации, а узкое их место — операция чтения; именно ее надо масштабировать, т.е. наращивать ресурсы и увеличивать производительность системы — естественно, при минимуме затрат.

Один из инструментов, удовлетворяющих эти нужды современных бизнес-приложений — продукт *MySQL Fabric* от корпорации Oracle. Эта расширяемая платформа с открытым кодом создана для работы с фермами серверов *MySQL*, упрощает операции с базами данных (БД) и обеспвчивает ВД, автоматически выявляя сбои и перераспределяя нагрузку, и масштабируемость, посредством автоматизированного шардинга (сегментации) данных. Эти возможности реализуются и по отдельности, и совместно. *MySQL Fabric* управляет репликацией и при отказе главного сервера замещает его одним из подчиненных. Для масштабирования *MySQL Fabric* автоматизирует шардинг данных на основе маршрутизации запросов к серверу (или серверам, если *MySQL Fabric* используется совместно, с ВД) и ключа шардинга, задаваемого приложением: ключ позволяет узнать, на каком сервере лежат те или иные данные.

Работа MySQL Fabric

MySQL Fabric реализован в виде административного процесса mysqlfabric и коннекторов, позволяющих перенаправлять запросы к конкретному *MySQL*-серверу на основе полученной маршрутной информации. Процесс mysqlfabric обрабатывает все управляющие запросы, получаемые из командной строки интерфейса *MySQL Fabric* и от других процессов через интерфейс XML/RPC, а также отвечает за мониторинг главного сервера и обнаружение отказов. Всё состояние фермы серверов *MySQL* записывается в хранилище *MySQL Fabric*, которое реализовано на СУБД *MySQL*. Приложение обращается к БД через *MySQL*-коннекторы, которые хранят кэш маршрутной информации, получаемой от процесса mysqlfabric, и затем использует эту информацию для отправки транзакций или запросов на нужный *MySQL*-сервер, минуя прокси, что позволяет избежать задержек. Сейчас *Fabric* поддерживает коннекторы для PHP (pre-GA+Doctrine), Python, Java+Hibernate, .Net и C (pre-GA).

Концепция ВД

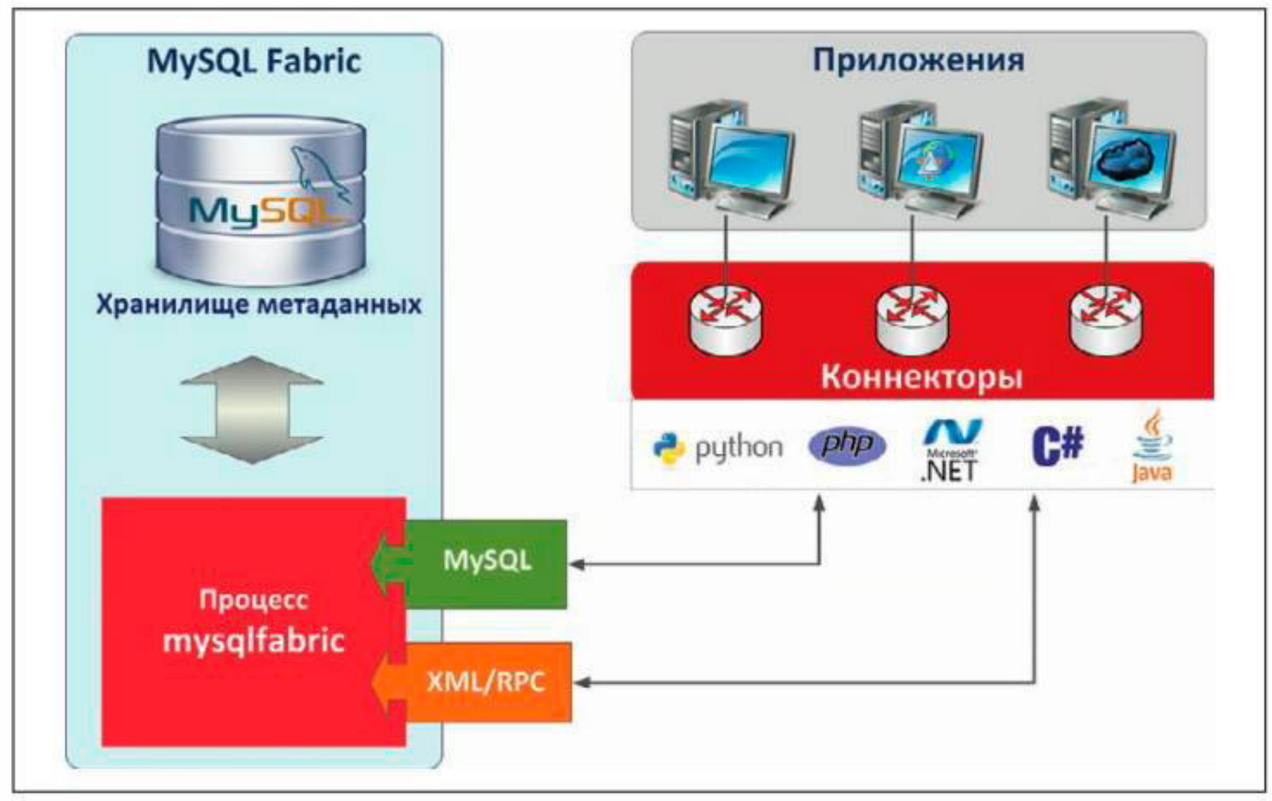
В основе *MySQL Fabric* — понятие высокодоступной группы (High Avaliability или HA-группа): это пул из двух и более *MySQL*-серверов. В любой момент времени один из этих серверов является главным (*MySQL Replication Master*), а все остальные — подчиненными (*MySQL Replication Slaves*). Каждой HA-группе присваивается уникальный идентификатор; именование производится по схеме регулярного выражения (shard_1, shard_2 и т.п.). Задача HA-группы — гарантировать бесперебойный доступ к данным, хранящимся в этой группе.

MySQL Fabric отслеживает состояние *MySQL*-серверов и гарантирует доставку запросов и транзакций на жизнеспособный сервер. Заметив отказ главного сервера, *MySQL Fabric* выбирает один из подчиненных серверов и назначает его главным, а остальным подчиненным серверам в HA-группе автоматически рассылаются обновления от нового главного сервера. Это позволяет выполнять операции чтения/записи в обычном режиме и без потерь. Еще одна ключевая возможность *MySQL Fabric* — маршрутизация запросов к базе данных, автоматическое перенаправление транзакций на текущую главную БД и балансировка нагрузок по выполнению запросов для подчиненных БД. При отказе главного сервера и изменении топологии сети приложению не надо вносить изменений для отправки транзакций на новый главный сервер. Логика обработки инкапсулирована внутри *MySQL Fabric* и прозрачна для приложения.

Другая возможность продукта — реализация географического резервирования, путем формирования HA-групп в различных дата-центрах. *MySQL Fabric* хранит свое состояние в БД *MySQL*, и его можно скопировать на удаленный сервер.

Путь к масштабированию

MySQL Fabric легко позволяет автоматически выполнять линейное масштабирование операций и чтения, и записи, благодаря автоматическому разнесению данных на шарды (shard [*англ.* «обломок»] — сегмент данных в таблице). Еще одна концепция *MySQL Fabric* — глобальная группа [global group]: она хранит обновления, которые следует

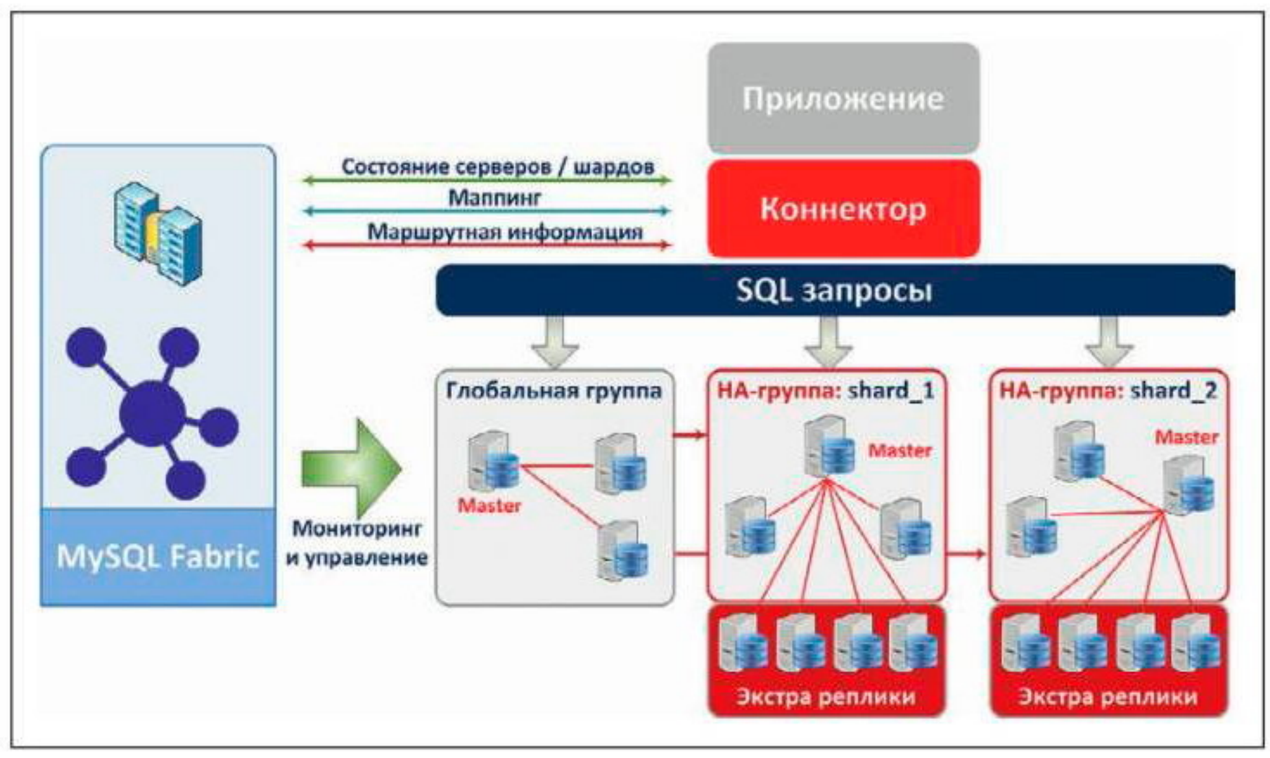


› **Взаимодействие *MySQL Fabric* с приложениями.**

распространить на все шарды из схемы шардинга. Репликацию данных выполняет утилита *MySQL Replication*. Каждый шард может храниться либо на одном *MySQL*-сервере, либо в HA-группе. Администратор может выбирать, какой таблице нужен шардинг, какие столбцы таблицы взять в качестве ключей шардинга, а также задавать функции (HASH или RANGE) для маппинга данных. *MySQL Fabric* поддерживает операции разделения и перемещения шардов. Если сервер, где хранится шард, надо вывести из группы, или ему не хватает ресурсов для хранения данного шарда, *MySQL Fabric* может переместить шард на новый сервер или их группу. Если на сервере или группе объем данных в шарде или скорость транзакций растет слишком быстро, *MySQL Fabric* делит этот шард на два или более и шлет на хранение на другой сервер или группу. Операции с шардами для приложения прозрачны. *MySQL Fabric* выполняет автобалансировку запросов между доступными подчиненными серверами в пределах шарда. Пользователь может назначить серверам свои веса. ВД можно обеспечить и без шардинга, создав одиночную HA-группу с главным сервером и одним подчиненным. Шардинг легко добавить по требованию.

Архитектура MySQL Fabric

MySQL Fabric обращается к серверам напрямую, без агентов. Метаданные хранятся в БД



› Архитектура *MySQL Fabric* — простая и эффективная.

MySQL; для администрирования и резервирования служат стандартные инструменты *MySQL*. Поддерживаются ACID-транзакции, что гарантирует непротиворечивость хранимых данных. Расширяемый фреймворк позволяет легко дорабатывать сервисы и дополнять их новыми. И, наконец, *MySQL Fabric* разработан той же командой, что и *MySQL Server* и *MySQL Replication*, поэтому сопровождается экспертной и квалифицированной поддержкой Oracle Premier Support.

Исходный код *MySQL Fabric* (на Python) открыт под GPLv2. Открыты и API-интерфейсы к метаданным. Данные о топологии, маршрутной информации и статусе *MySQL Fabric* можно получить или модифицировать из командной строки через протокол API XML/RPC, либо напрямую из хранилища по SQL-запросам. Поддерживается сервер *MySQL* версии 5.6.17 и выше. Зависимости — Python 2 (2.6 и выше) и Connector/Python 1.2.1 и выше. *MySQL Fabric 1.6.2* стал самостоятельным продуктом, но пока только для тестирования. Самая свежая промышленная версия — *1.6.1*, она входит в пакет *MySQL Utilities* и скачивается с http://dev.mysql.com/downloads/utilities/.

Не без ограничений

Процесс mysqlfabric — одиночный узел без резервирования. Он не защищен от сбоев и должен перезапускаться в случае отказа. Но это не означает падения всей системы: во время перезапуска маршрутизация операций идет в штатном режиме, используется локальный кэш. Шардинг не вполне прозрачен для приложения — приложению для доступа к БД нужен ключ. В запросах не поддерживаются связи между шардами, автоинкрементные столбцы нельзя назначить ключами шардинга. Маршрутизация основана на коннекторах, но их список ограничен. *MySQL Fabric* не поддерживает GUI и управляется через CLI, *MySQL*-протокол или API на XML/RPC. LXF

Oracle о своем продукте

«Многие крупнейшие в мире сайты и облачные приложения используют репликацию и шардинг *MySQL* для обеспечения высокой готовности и масштабируемости. Выпуская эту интегрированную систему с открытым исходным кодом, Oracle упрощает безопасное горизонтальное масштабирование *MySQL*-приложений, развернутых в корпоративной или облачной среде».

› Томас Улин [Tomas Ulin], вице-президент Oracle по *MySQL*.

LINUX FORMAT Вердикт

MySQL Fabric 1.6.2

Разработчик: Oracle
Сайт: http://labs.mysql.com
Лицензия: GPLv2
Цена: Бесплатно под GPL

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

» На удивление, Oracle сдерживает свое обещание развивать свободное промышленное ПО.

Рейтинг 9/10

MySQL — одна из самых популярных систем управления базами данных с открытым кодом

Оформите подписку на глобальную техническую поддержку Oracle для продуктов линейки MySQL, и вам будут доступны:

- 7500 специалистов в режиме 24×7
- Неограниченное количество запросов на обслуживание через Интернет или по телефону
- Патчи и обновления на портале MyOracleSupport.com
- Опыт использования продукта более чем у тысяч

+7 812 309 0686
WWW.LINUXCENTER.RU/SHOP/MY_SQL • INFO@LINUXCENTER.RU

Korora 22

Лес Паундер исследует ремикс дистрибутива Fedora со множеством удобных приложений, который мигом ускорит разработку вашего следующего проекта.

Вкратце

» Korora 22 основан на Fedora 22 и является последней версией дистрибутива, появившегося в 2005 г. как сборка Gentoo Linux. Korora — это отточенный опыт и прагматичный подход к выбору программ, который понравится далеко не каждому, но обеспечит хорошо сбалансированный подход для мейнстрима.

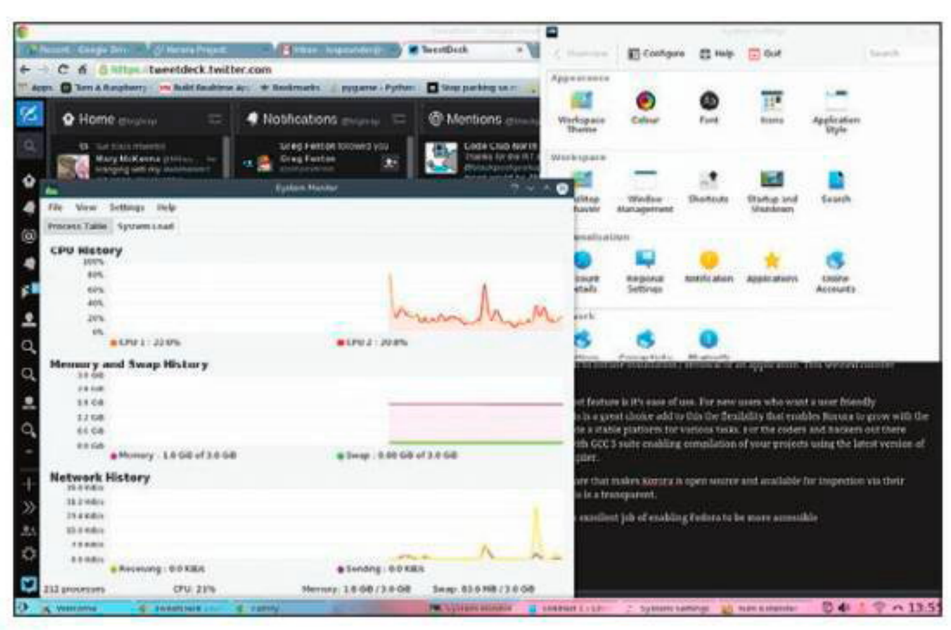
В сообществе Linux присутствуют два очень известных дистрибутива: Fedora и Ubuntu. Из них Fedora — это передовой дистрибутив, скорее для опытных пользователей, способных справиться с необычной ошибкой. А если Fedora приручить? Korora — это ремикс Fedora 22, который взял новейший дистрибутив и привел его к боеготовности для повседневного пользователя.

Korora поставляется с пятью различными окружениями рабочего стола. Для нашего обзора мы выбрали KDE, а в KDE есть все, что ни пожелаешь — приложения для любых задач; по умолчанию предпочитаются *Firefox* для просмотра web-страниц и *VLC* для воспроизведения мультимедиа. Естественно, эти предпочтения пользователь в состоянии изменить. Рабочее пространство KDE Plasma 5 обеспечивает четкий и ясный интерфейс, чрезвычайно отзывчивый благодаря аппаратному ускорению графического стека.

Мы протестировали Korora на четырехъядерном Core 2 Duo с 4 ГБ оперативной памяти и рады сказать: работает исключительно хорошо, даже при использовании довольно тяжеловесной среды KDE. Для слабых машин есть альтернативные окружения рабочего стола; очевидный выбор — *Xfce* и Mate. Для окружения Gnome поставляется версия 3.16 и предлагает отличный интерфейс пользователя, не слишком отличающийся от KDE.

Korora поставляется с рядом готовых дополнительных репозиториев, и они предоставляют медиа-кодеки для просмотра большинства типов медиа-контента. Также включено все необходимое для производства медиа, в том числе очень мощные приложения: пакет для обработки фото *Darktable* плюс *Kdenlive* — отличный инструмент для редактирования видео. В меню Мультимедиа есть дополнительные инструменты для редактирования и записи аудио, записи скринкастов и перекодирования видео в разные форматы — действительно приятный штрих, поскольку эти приложения нередко трудно установить из-за сложных зависимостей. Хотя и увеличивается размер скачиваемого ISO-образа, тогда как многие другие дистрибутивы стремятся его уменьшить, это очень хорошо, поскольку дает пользователю возможность попробовать такие приложения до их установки.

› Могучий Korora поставляется с разными окружениями рабочего стола, но элегантный интерфейс ему обеспечивает чистая красота KDE Plasma 5.



› Korora старается предоставить дистрибутив, работающий «из коробки» с широким спектром полезных приложений.

Еще установщики

Если вам надо установить дополнительное проприетарное программное обеспечение, например, драйверы для видеокарт, это осуществляется через приложение *pharlap*, а дополнительные приложения, такие как *Chrome*, могут быть установлены с помощью установщика программного обеспечения *Yum Extender*. Это графический вариант *DNF*, который заменяет *Yum* в качестве менеджера пакетов. Его интерфейс очень элегантен и прост в использовании, но раздражает значок шестеренки для запуска установки или удаления приложения. Это кажется нелогичным.

Главная черта Korora — простота использования. Это превращает его в отличный выбор для начинающих пользователей, которым нужен удобный дистрибутив: он способен расти вместе с пользователем и обеспечивать стабильную платформу для различных задач. Для настоящих кодеров и хакеров Korora поставляется с пакетом *GCC 5*, что позволит компилировать ваши проекты при помощи последней версии классического компилятора.

Korora делает Fedora гораздо доступнее и предлагает, если требуется, для желающих простой переход на Linux в профессиональной сфере.

Если вы приверженец СПО, Korora — это не совсем ваш дистрибутив, поскольку он многое делает за вас. Но если вы новичок в Linux или любитель менять дистрибутивы, желающий быстро их устанавливать и сразу работать, то это идеальный дистрибутив для исследования. Он эффективен и по-настоящему впечатляет. LXF

LINUX FORMAT Вердикт

Korora 22

Разработчик: Korora Project
Сайт: kororaproject.org
Лицензия: Разные

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

» Не для пуристов, но цельный прагматичный дистрибутив, предлагающий куда больше, чем стандартная Fedora.

Рейтинг 8/10

Samsung Z3 (SM-Z300F)

Александр Толстой две недели не мог израсходовать заряд этой тонкой «плитки». Вскрытие показало: внутри не ядерный реактор, а открытые исходники...

Вкратце

» **Смартфон на Tizen OS 2.4 для крупных заказчиков в России.**

Спецификации

» **ОС** Tizen 2.4
» **Платформа** Spreadtrum (SC7730S)
» **CPU** 4-ядерный, 1,3 ГГц, с ускорителем графики Mali 400
» **ОЗУ** 1 ГБ
» **Хранилище** 8 ГБ NAND
» **Экран** 5" AMOLED, 720×1280, 294 ppi
» **Связь** GPS/Glonass, LTE
» **Батарея** 2600 мА·ч

В тестовую лабораторию **LXF** попал смартфон Z3 от Samsung на ОС Tizen 2.4. Tizen — перспективная ОС для мобильных устройств, носимой электроники и предметов «умного дома». В ней нет Java-подобной ВМ (снижающей производительность) или проприетарных сервисов Google, а также разумнее, чем в Android, организован доступ к исходному коду: влиться в сообщество и сделать вклад в целом проще. Одно из ключевых преимуществ Tizen — сверхнизкое энергопотребление, во многом обязанное тому, что в системе мало сервисов, постоянно лезущих в Сеть.

Наш подопытный смартфон Z3 является развитием модели Z1 (первого смартфона на Tizen, вышедшего год назад). Z3 впервые попал к разработчикам в июле 2015 г. во время Tizen Developer Summit в Бангалоре — Samsung раздала несколько тысяч устройств бесплатно, надеясь привлечь энтузиастов открытого кода к развитию Tizen.

Z1 был бюджетным (до $100), без особых наворотов, а Z3 — лучше и дороже: его официально продают в Индии за $160.

Выглядит аппарат «не хуже других», а благодаря сочному дисплею с большими углами обзора — даже дороже своей цены; задняя крышка из приятного материала Soft Plastic ощущается очень убедительной. Под съемной крышкой спрятан аккумулятор. В наших тестах Z3 прожил на первой зарядке 2 недели — его зарядили еще в Корее, потом привезли в Россию и (не сразу) передали в редакцию **LXF**, где он несколько дней пролежал в столе. При стандартном сценарии — пара-тройка звонков в день, пара часов на web-серфинг и просто блуждание по меню — Z3 уверенно работал 3 дня, и это при самой обычной батарее.

Аппарат для разработчиков TM1 (Tizen Mobile 1), версия для Индии SM-Z300H и версия для России SM-Z300F идентичны лишь с виду: в одной модели нет LTE, в другой не тот размер корневого раздела, третьей нужен особым образом подписанный модуль доверенной загрузки (sboot.bin). И прошивка от одного аппарата не годится для другого. Программная начинка нашего SM-Z300F минималистична: нет ни магазина, ни добавочных приложений, ни средств разработки. Все разделы меню легко умещаются на одном экране! Расположение пунктов внутри разделов похоже на примененное в Android, и пользователь, далекий от разработки, разницы не заметит. Файловая система области пользователя тоже подозрений не вызовет, и изучается без проблем. Отсутствие излишеств и потенциальных источников угроз улучшают безопасность и надежность устройства.

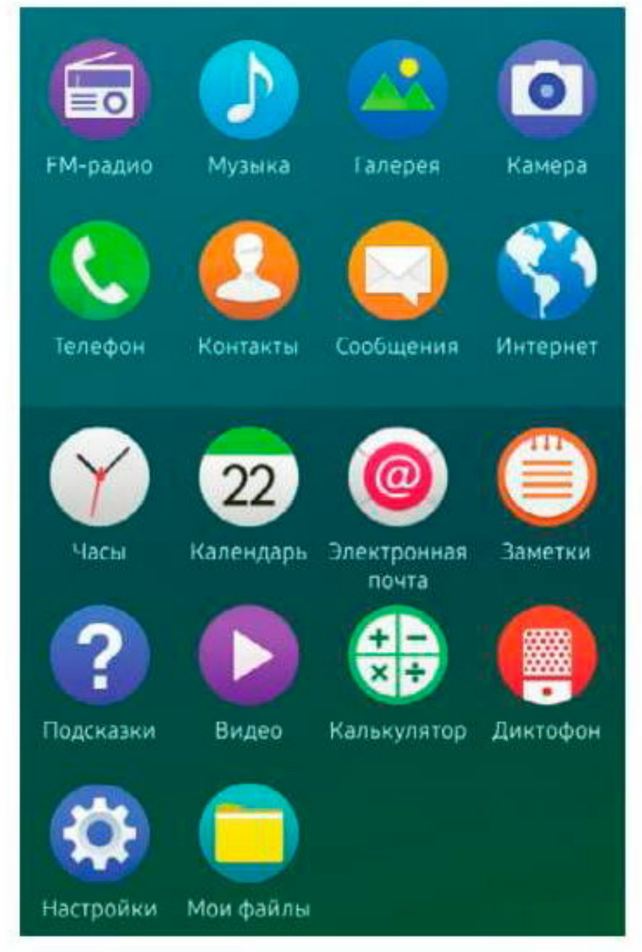
Работа с кодом начинается с регистрации на https://www.tizen.org/user/register и копирования себе исходного кода через *Git*. Готовые наработки затем следует заслать в web-службу Gerrit для проверки кода на соответствие требованиям Tizen TSG (Technical Steering Group). Сборка кода и формирование образов (в терминологии Tizen это «цели») происходят на стороне сервера при посредстве сборочной фермы OBS. На выходе — готовые архивы в формате **tar.gz**. Внутри архива находится набор образов (***.img**, ***.bin**), которые и прошиваются в корневой раздел устройства.

Сейчас приняты два подхода к прошивке смартфонов на ОС Tizen. Для владельцев TM1 имеется утилита *Ithor* (Linux THOR), она заливает прошивку в телефон, загодя переведенный в «Режим загрузки». Телефон распознается в Linux как устройство **/dev/ttyACM0**. Образы для прошивки берутся с официального сайта Tizen — http://download.tizen.org/releases/2.4/2.4-mobile/.

Второй подход относится ко всем устройствам Samsung, выпущенным на рынок. Для них есть Windows-утилита *Odin*, которая в принципе делает то же, что и *Ithor*, но имеет довольно неряшливый вид и множество неиспользуемых кнопок с непонятными названиями. К счастью, у бесплатного (однако проприетарного) *Odin* имеется замечательный аналог с открытым исходным кодом *Heimdall* (https://github.com/Benjamin-Dobell/Heimdall/downloads), который превосходно работает со всеми смартфонами Samsung, включая модель Z300H.

Прошивки для Z3 Samsung публикует на официальном сайте сообщества Tizen (http://www.tizenexperts.com), но выкладывает их почему-то на бесплатные хостинги с рекламой и необходимостью ждать 60 секунд. На момент написания этих строк самая свежая версия Tizen для Z3 — прошивка 2.4.0.2 (Z300HDDU0BOL4), но разработка продвигается быстро: на новые сборки можно рассчитывать не реже раза в месяц.

Выше говорилось про модель Z300H, тогда как в наших руках оказалась Z300F. Увы, пока официальной прошивки именно для этой модели нигде нет, поэтому «поиграть» с Tizen в полную силу пока нельзя.



› **Не проходите мимо, когда увидите это устройство на прилавке...**

Однако если вы интересуетесь данной ОС, ничто не мешает использовать эмулятор Tizen, входящий в состав общедоступного SDK для Tizen (https://developer.tizen.org/development/tools).

У SM-Z300F отличное соотношение цены и качества: как-никак, это двухсимочный телефон с LTE и качественным дисплеем, на свободной ОС. Правда, пока приложений в Магазине (http://www.tizenstore.com) немного. Но там имеется замечательная программа Open Mobile Application Compatibility Layer, через которую можно запустить ряд Android-приложений (где взять ***.apk**, наверное, объяснять не надо). LXF

LINUX FORMAT Вердикт

Смартфон SMZ300F

Разработчик: Samsung
Сайт: www.tizen.org
Лицензия: GPLv2
Цена: ~$160

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

» Производит приятное впечатление, в т.ч. благодаря перспективной ОС.

Рейтинг 9/10

Dell Chromebook 13

Хромбук от Dell реально для бизнеса... или **Чуонгу Нгуену** хочется в это верить.

Спецификации

- **ОС** Chrome OS
- **CPU** Intel 1,5 ГГц Celeron 3205U (опции: Core i3 или i5)
- **ОЗУ** 2 ГБ (опции: 4 ГБ, 8 ГБ)
- **Диск** 16 ГБ SSD (опция: 32 ГБ)
- **Дисплей** 13,3" 1920×1080 IPS (опция: сенсорный)
- **Связь** 802.11ac, Bluetooth 4.0 LE
- **Порты** 1×USB 3.0, 1×USB 2.0, HDMI 1.4, MicroSD, спаренное гнездо микрофон/наушники
- **Габариты** 323,4×225,8×18,4 мм
- **Вес** 1,47 кг

В отличие от конкурентов, компания Dell ориентирует свои хромбуки на вполне конкретные ниши: например, Chromebook 11 предназначался для рынка образовательных услуг, а на сей раз Dell возвращается с большим, ближе к премиум-классу Chromebook 13 для корпоративных пользователей.

Сделанный из высококачественных материалов, Chromebook 13 бросает вызов таким устройствам, как металлизированный Chromebook Pixel от Google и великолепный Chromebook Flip от ASUS. Но, в отличие от этих предложений потребительского класса, мощный Dell Chromebook 13 имеет программное обеспечение VPN корпоративного уровня, средства мобильного менеджмента и развертывания и ПО виртуальных рабочих столов.

У Chromebook 13 корпоративное не только программное обеспечение, но и внешний вид. На самом деле, он похож на гибрид Dell XPS 13 и Latitude 12 7000 с покрытием из ткани из углеродного волокна на крышке, как в XPS 13 и Latitude, для укрепления и защиты экрана. Откинув крышку, вы обнаружите металлическую панель островковой клавиатуры из магниевого сплава с уменьшенными клавишами с подсветкой. (Клавиатура полноразмерная, но сами клавиши выглядят чуть поменьше, чем клавиши на MacBook Pro. На самом деле, они очень напоминают клавиатуру на Latitude 12 серии 7000.) Большой стеклянный трекпад центрирован по отношению к клавиатуре.

Продолжая тему металла: каркас ноутбука «обернут» листом алюминия, внутри крепятся направленные вниз динамики. Петли, встроенные в корпус, по конструкции очень похожи на петли MacBook Pro; позади них размещаются вентиляционные отверстия.

Покупатели могут выбрать между процессорами Intel Celeron или Intel Broadwell Core i3 или i5. Chromebook 13 может быть оснащен 2, 4 или 8 ГБ оперативной памяти. Также можно выбрать твердотельный накопитель на 16 ГБ или 32 ГБ и дополнить его картой памяти SD. В топовой комплектации с процессором Core i5, накопителем 32 ГБ и 8 ГБ оперативной памяти, Chromebook 13 стоит всего на £100 меньше, чем Chromebook Pixel. Ой...

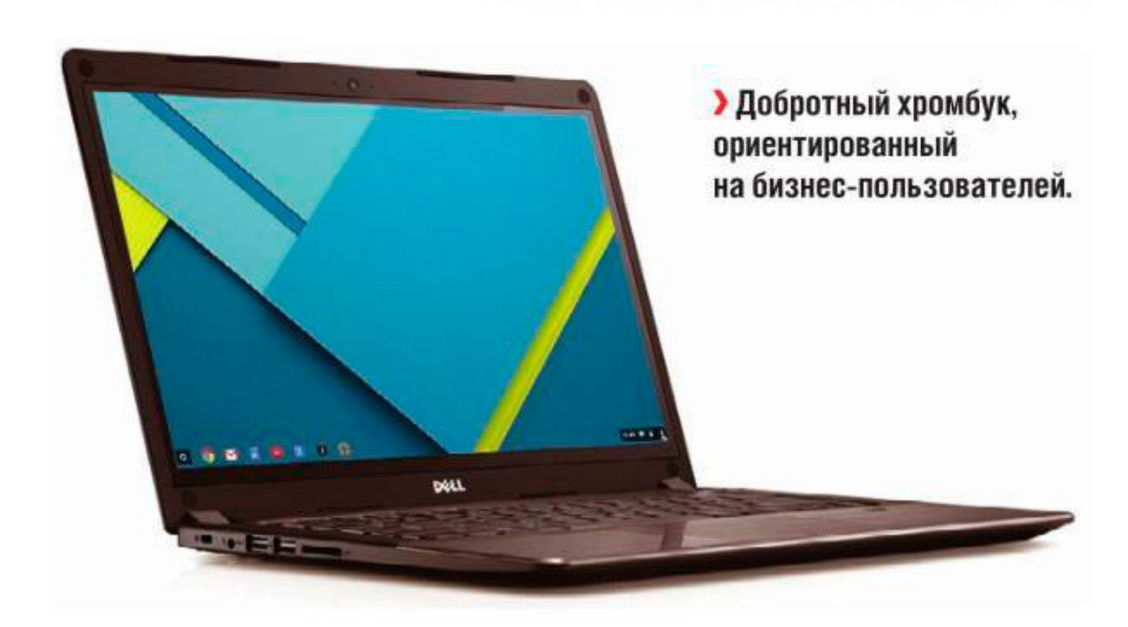
Chromebook 13 имеет версии с разными дисплеями. Мы видели версию с 13,3-дюймовым антибликовым не сенсорным экраном с разрешением Full HD 1080p, но Dell также предлагает вариант сенсорного экрана с тем же разрешением 1080p. Сенсорный экран будет использовать Gorilla Glass для защиты от царапин. Dell на обеих моделях использует экран IPS, дающий широкие углы обзора.

Корпоративные опции

Ключевым моментом является поддержка ряда настольных программ виртуализации, позволяющая запускать приложения Windows на Chrome OS. Клиенты VDI [Virtual Desktop Infrastructure] Citrix, VMware и Dell Wyse на хромбуке позволят вам виртуализовать Windows 7, 8 или 10. В наших тестах Citrix потребовалось некоторое время, чтобы запустить Windows 7 Enterprise через офисный Wi-Fi. А после запуска программы открывались моментально.

Руководствуясь своим пониманием корпоративного, Dell предлагает то, что некоторые видят как легкую потребительскую операционную систему в корпоративной среде. Она сталкивается с жесткой конкуренцией со стороны более дешевых устройств потребительского класса, включая Acer Cloudbook с Windows и другие недорогие доступные хромбуки.

В начальном и среднем уровнях конфигурации Dell Chromebook 13 обеспечивает привлекательную альтернативу для Google Chromebook Pixel. Взяв пакет в целом, Dell может выйти на выигрышную формулу. Его решения управления мобильностью упрощают ИТ-менеджерам развертывание и управление хромбуком, а проверенная поддержка VPN-соединения и виртуальных рабочих станций позволяет корпоративным пользователям управиться и со своими Windows. LXF



Добротный хромбук, ориентированный на бизнес-пользователей.

LINUX FORMAT Вердикт

Dell Chromebook 13 7310

Производитель: Dell
Сайт: www.dell.com/uk
Цена: £407

Функциональность	9/10
Производительность	8/10
Удобство в работе	9/10
Оправданность цены	5/10

» Функции виртуализации и управления могут вывести его в лидеры, несмотря на дороговизну.

Рейтинг 8/10

Company of Heroes 2

Как настоящий бравый русский солдат, **Рич Мак-Кормик** вступает в бой с винтовкой, но без патронов, пытаясь уцелеть на Восточном фронте.

Спецификации

- **ОС** Ubuntu 14.04 64-бит, SteamOS
- **Процессор** Intel Core i5 3,4+ ГГц
- **ОЗУ** 8 ГБ
- **Диск** 35 ГБ
- **GPU** Nvidia серии 760, 4 ГБ VRAM

Игра называется *Company of Heroes 2* (*CoH2*), но чтобы добраться до героизма в этой стратегии реального времени от Relic, понадобится часов 15 кампании. Действие разворачивается на суровом Восточном фронте Второй мировой войны и связано с самыми кровавыми и жестокими эпизодами конфликта. Восточный фронт принял главный удар войны: немецкий Вермахт понес 80% своих жертв к востоку от Берлина, Советы потеряли здесь около 26 млн человек, из которых солдатами были 8,6 млн.

Играть в такое настолько же жутко, как и знать об этом. Военная мощь Советского Союза обуславливалась способностью страны щедро жертвовать молодыми мужчинами и женщинами, отправляя их сражаться и умирать за Родину. Отсюда идет и механика игры: как у советского главнокомандующего, в вашем распоряжении почти безграничные человеческие ресурсы, которые можно послать на смерть.

Отряды лучше подготовлены, более опытны и профессиональны, и включают такие разные группы, как орудийные расчеты, снайперы и ударные войска, вооруженные дымовыми и осколочными гранатами. Контингент подбрасывается со всех концов каждые 30 секунд. «Нижние чины» не могут противостоять практически ничему, но зато их можно раздобыть быстро и практически даром. Ограничены только два из четырех ресурсов *CoH2*: численность состава бойцов — восполнять которую на средней сложности до смешного легко — и численность населения.

Чмсло бойцов со времен первой *Company of Heroes* возросло, и во многих миссиях вы теперь управляете 135 крошечными пешками. По ощущениям, это оказывается много: отслеживать всю армию будет тяжеленько, особенно учитывая, что многие миссии разворачиваются сразу на двух фронтах, и надо защищать свою базу от мобильных групп пехоты. Поскольку число призывников не ограничено, такие операции можно особо не продумывать.

Худшие миссии *CoH2* ощущаются так, будто все в них перевернуто с ног на голову. Вы не играете отважных, умных солдат, а воплощаете саму безжалостную машину войны, бросая бесконечные ресурсы на разрушение кирпичной стены до тех пор, пока в ней от выветривания не появится дыра; и игра страдает от тяжести собственного сюжета. Бремя истории особенно тяжко давит на ранних этапах кампании. Фашисты продвигаются в глубь России, советские же войска целиком отступают, а значит, многие миссии нацелены не столько на победу, сколько на эвакуацию.

Как у настоящих советских генералов, в вашем распоряжении огромные войска, и можно одним щелчком выдвинуть их на Восточный фронт.

Здесь есть место и славе, и героям. Самые памятные — более тайные миссии *Company of Heroes 2*. На одном из уровней действие происходит на польской территории; вы управляете супер-снайпером и несколькими отрядами Сопротивления. В отличие от большинства отрядов, снайперы стреляют только по приказу, и рыскание по польским чащам станет очень эффективным боевым действием: послав часть снайперов к кромке леса прикрывать другую подкрадывающуюся группу, вы затем уничтожите вражеский отряд прицельным залпом из шести пуль.

Провал разведки

Действия искуственного интеллекта (ИИ) в *Company of Heroes 2* несколько сомнительны, что проявляется в целом ряде путаных сценариев. Мы не такие уж эксперты в механизированной войне, но навряд ли основной тактикой Вермахта при обнаружении танков противника было бросаться к ним со всей мочи, как обрадованный появлением хозяина щенок. У части войск наблюдались проблемы с ориентированием на местности.

Но с этим слабым вражеским ИИ справиться легче, чем вы думали. После отступления в первой трети кампании Красная армии сумела воссоединиться и нанести ответный удар. Почти весь остаток сюжета мы, как паровой каток, неумолимо двигались к Берлину.

Гораздо лучше в *CoH2* миссии *Theatre of War*. Эти бои делятся на три категории: одиночные, совместные сценарии и сражения с ИИ. В то время как сама кампания кажется раздутой, эти миссии, напротив, радуют четкой прицельностью и вариативностью. Совместные миссии являются более или менее свободными по форме. Оба союзных генерала могут объединять свои силы в «кулак» либо разделять их по родам войск и координировать.

Игра местами нескладная, не способная поддерживать все свои системы; уж слишком много разрозненных подвижных частей, мешающих ощутить ее целостность. Но конечным результатом, несмотря на небольшие потери по пути, все-таки будет победа. LXF

LINUX FORMAT Вердикт

Company of Heroes 2

Разработчик: Feral Interactive
Сайт: https://store.feralinteractive.com
Цена: £30

Игровой процесс	7/10
Графика	8/10
Увлекательность	8/10
Оправданность цены	7/10

» Среди стратегий в реальном времени, Company of Heroes 2 похож на СССР: огромная, мощная и лишь немного трещит по швам.

Рейтинг 8/10

ЭМУЛЯТОРЫ

AMIDuOS: гипервизор виртуального режима

«Двуосная» система таит угрозу Microsoft и Google.

Пользователям ПК хорошо знакомо название American Megatrends Inc. — со времен машин с DOS и до настоящего времени компания AMI остается крупнейшим поставщиком BIOS. В 2015 г. AMI представила продукт DuOS (или AmiDuOS) — эмулятор Android для ОС семейства Windows. AmiDuOS загружает на хостовую систему Windows (вплоть до Windows 10) полный образ гостевой системы Android Lollipop. В приложениях Android поддерживаются ускорение графики и «родные» расширения ARM. О проекте рассказали его создатель, вице-президент AMI Говинд Котхандапани [Govind Kothandapani], и руководитель по маркетингу Терри Оцубо [Terry Otsubo]. Первоначально планировали при помощи Android объединить планшеты и ноутбуки, но, по словам Оцубо, «дела шли не слишком хорошо». Проект не одобрили в Google, что сказалось на совместимости AMI с Android. Тогда AMI решила выйти на рынок B2C («бизнес для бизнеса») и заняться онлайн-продажами DuOS. «Мы разместили в Windows полный стек Android, сохранив его структурированным. Особых изменений базовая структура Android не претерпела, поэтому за выходом новой версии следуют быстрые обновления. Мы только эмулируем код, который должен быть эмулирован».

Это заставлят Microsoft задуматься о стратегии — есть риск растерять клиентскую базу разработчиков, 20 лет воспринимаемую как должное: в Windows те ощущают нехватку приложений, особо явную в секторе мобильных устройств. Идеально для Microsoft, чтобы разработчики создали единый API для мобильных устройств Windows PC и ARM. Нужные приложения уже написаны — на Java, для Android. Сейчас DuOS есть только для архитектуры x86, и планов портирования своего продукта на ARM у AMI нет. Но Intel продолжает снижать энергопотребление x86, и ходят слухи о выпуске Microsoft телефона на Intel x86... в итоге Android придет на мобильные устройства с Windows.

› В рамках концепции «двуосной» системы пользователь на одном устройстве совмещает всё лучшее от планшетов и традиционных ноутбуков.

РАЗРАБОТЧИКАМ

Android Studio 2.0 preview

Google сосредоточилась на стабильности и скорости своей среды разработки.

На состоявшемся в ноябре 2015 г. в Маунтин-Вью, Калифорния (в этом городе расположена штаб-квартира Google) Android Developer Summit, Google представил Android Studio 2.0 Preview — значительное обновление своей интегрированной среды разработки (IDE) приложений для Android. Новая версия инструментария предоставляет разработчикам возможность практически мгновенного обновления приложения после внесения изменений в код; добавлены профилировщик средствами GPU и нативный отладчик Java.

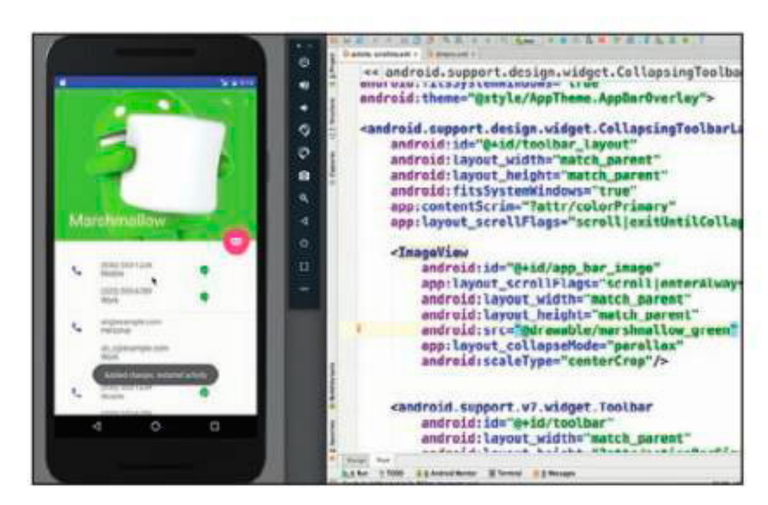
В основу Android Studio положена IntelliJ IDEA, популярная IDE Java. Google не рекомендует дальнейшего использования IDE Eclipse.

В новой версии пользователь может внести изменения в код в то время, когда приложение запущено и работает в эмуляторе Android. Щелчок по Run мгновенно применит новый код, после чего приложение будет перезапущено уже с внесенными изменениями.

По данным Google, сборки, выполненные в новом релизе, работают в 2–2.5 раза быстрее. Эмулятор Android также работает быстрее — учтены претензии разработчиков.

Другая новая функция — профилирование средствами GPU для оптимизации кода OpenGL ES (Open Graphics Library for Embedded Systems). Пользователь добавляет к своему проекту библиотеку трассировки вместе с небольшой частью кода для загрузки профилировщика. Профилировщик был продемонстрирован на саммите, сопровождаемый предупреждением, что пока он находится в экспериментальной фазе.

И еще одна экспериментальная функция, показанная на саммите: возможность переходить от кода на Java к коду на C++ во время сеанса отладки. Под Android существует как Java SDK, так и «родной» SDK для разработчиков на C и C++, и такая функция будет востребована в проектах, использующих код от обеих SDK.



› Новая функция instant run позволяет почти мгновенно обновлять запущенное под отладкой мобильное приложение при изменении исходного кода.

Google позиционирует Android Studio как основную среду разработки для Android. «Раньше мы предлагали отдельные автономные инструменты, теперь же пытаемся разместить все в IDE», заявил на саммите лидер Studio Tech Lead Top Норбю [Tor Norbye].

Источники: www.theregister.co.uk

www.linuxformat.ru/subscribe

«УМНЫЕ» ЧАСЫ

Сверяем время по Android

Fossil Q Founder с Android Wear поступили в продажу.

Устройство, управляемое процессором Intel Atom, выглядит как традиционные часы, причем ценник у него отнюдь не запредельный. Продажи Fossil Q Founder начались 24 ноября, начальная цена $275.

В Google уверены: авторитет бренда Fossil будет способствовать росту популярности платформы Android Wear.

Fossil рекламирует кроссплатформенную поддержку Q Founder: устройство работает со смартфонами под управлением Android 4.4 или выше, или же с Apple iPhone 5 и более новыми моделями, минимально с iOS версии 8.2. Но, хотя часы с Android Wear способны работать с iPhone, функциональность при этом будет ограничена по сравнению с возможностями взаимодействия со смартфонами на Android.

Для взаимодействия с пользователем Fossil Q Founder предлагает типичный для часов на Android Wear круглый сенсорный экран, помещенный в стальной водонепроницаемый корпус. Аккумулятор емкостью 400 мА·ч обеспечивает устройству сутки работы в автономном режиме. Часы укомплектованы 4 ГБ встроенной памяти и модулем Bluetooth 4.1. В комплект входит браслет из нержавеющей стали.

«Умными» часами компания Fossil начала заниматься задолго до того, как нынешнее поколение носимых устройств превратилась в хит. Еще в 2004 г. в партнерстве с Microsoft компания создала серию подключаемых часов, использующих платформу Microsoft SPOT и FM-радиоволны для получения информации.

Конечно, часы Android Wear создают многие производители мобильных устройств — можно назвать Samsung, Huawei, LG, Motorola и ряд других. Отличием бренда Fossil является значительный вес на рынке традиционных часов, и его предложение модели с Android Wear, даже при такой низкой цене, несет дополнительные выгоды для Google.

› Для поклонников традиционных наручных часов Fossil предлагает Q Founder на Android Wear, работающими как с Android-устройствами, так и устройствами на iOS.

ПРОГНОЗЫ

Квартальный отчет WQWDT от IDC

Поставки носимых устройств в 2019 г. превзойдут 200 млн.

Мировой рынок носимых устройств (в англоязычном сегменте — wearables) ждет рост после того, как поступят устройства второго и третьего поколений. Новинки будут основываться на программном и аппаратном обеспечении своих предшественников, но свободны от некоторых недостатков и проблем, которые потенциальные клиенты имеют сегодня. Согласно прогнозу IDC из очередного ежеквартального отчета Worldwide Quarterly Wearable Device Tracker, в 2016 г. на рынок будет поставлено 111,1 млн устройств, что на 44,4% больше 80 млн в 2015 г. В 2019, последнем году, охватываемом прогнозом, общий объем поставок достигнет 214,6 млн единиц, в результате чего величина совокупных темпов годового роста за 5 лет составит 28%.

Наиболее распространенные сегодня типы носимых устройств, такие как фитнес-трекеры, достаточно просты, но в течение следующих нескольких лет ожидается увеличение как форм-факторов, так и типов. «Разумная» одежда, очки и даже ушные гарнитуры сегодня находятся на ранних стадиях массового признания, и в настоящее время еще не могут считаться значительно «умнее» своих аналоговых коллег, но уже следующее поколение wearables предложит значительно улучшенную функциональность и, не исключено, даже расширит возможности человека.

Одним из самых популярных видов носимых устройств будут «умные» часы: 34,3 млн проданных единиц в 2016 г. (21,3 млн в 2015 г.). К 2019 г. общий объем поставок составит 88,3 млн единиц, а среднегодовой темп прироста — 42,8%. Ведущая роль в этом сегменте останется за WatchOS от Apple: 45,2 млн устройств и доля платформы на рынке 51% к 2019 г. Android и Android Wear — на втором месте, но далеко позади, несмотря на расширяющийся список поставщиков, включающий как технологические компании (ASUS, Huawei, LG, Motorola, Sony), так и традиционных производителей часов (Fossil и Tag Heuer). Функциональности различных устройств на Android Wear в значительной степени схожи, и для пользователей остается выбор только по цене и дизайну.

Pebble, пионер создания «умных» часов», продолжит уступать свою долю рынка AndroidWear и watchOS, но не исчезнет совсем. Простой интерфейс устройств и невысокая, относительно других платформ, цена делают Pebble одними из самых доступных «умных» часов на рынке.

Tizen от Samsung остается «темной лошадкой» и представляет угрозу для Android Wear, в том числе за счет совместимости с большинством ведущих смартфонов на Android и выбора приложений, конкурирующего с Android Wear. Преимуществом Tizen являются технологические наработки Samsung, в том числе QWERTY-клавиатура на экране «умных» часов, сотовая связь и новый пользовательский интерфейс. Такое сочетание позволяет Tizen выделиться, но его недостаточно, чтобы догнать Android Wear и watchOS.

> Преимущество Tizen — технологические наработки Samsung.

Маленький, но, тем не менее, значительный сегмент носимых на запястье «умных» часов останется за устройствами под управлением ОС реального времени (RTOS). Такие устройства с, как правило, проприетарными ОС от OEM-производителей, могут быть востребованы в определенных ситуациях, но значительную долю рынка занять вряд ли смогут. LXF

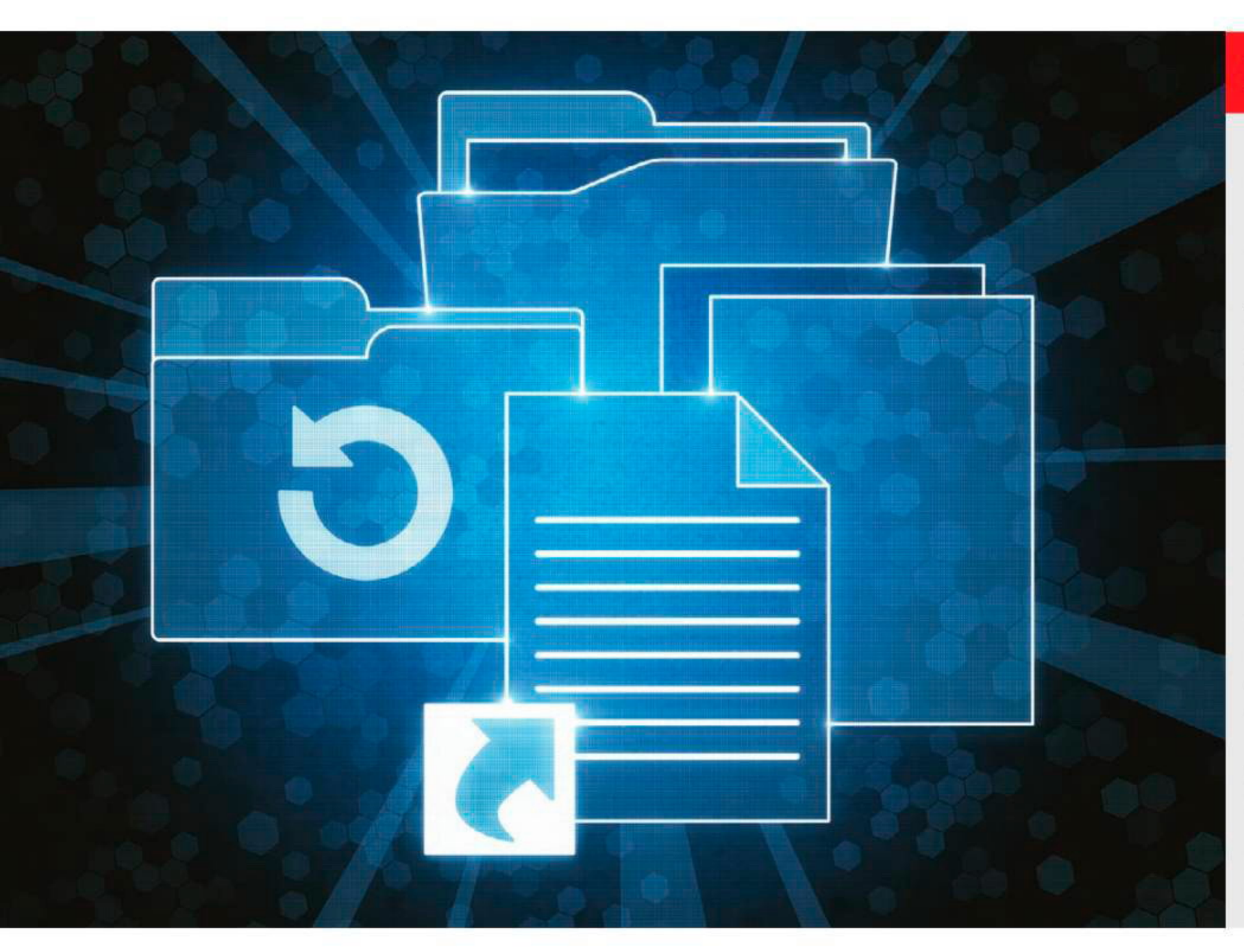
Источники: www.zdnet.com; www.idc.com

Сравнение

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

Синхронизация файлов

Поскольку у большинства из нас по нескольку компьютерных устройств, **Нейл Ботвик** рассматривает программы для синхронизации данных на всех.



Про наш тест...

Большая часть тестов выполнялась на нескольких компьютерах Linux (настольный ПК и ноутбук с тестовым Gentoo, хотя дистрибутив здесь роли не играет). Мы использовали директории, содержащие файлы, которые вы, вероятнее всего, решите синхронизировать: фотографии, фонотека солидного размера и кое-какие документы общего назначения. Кросс-платформенное тестирование проводилось на Nexus 6 с Android и Windows 7, запущенной на ВМ. Мы отмечали наличие программ для других платформ, но не имели возможности их тестировать, ведь это территория без iPhone! Мощность здесь не проблема, однако мы приглядывали за скоростью передачи. Важнее всего получить файлы, которые вам нужны, когда они вам нужны. Поскольку синхронизация обычно проходит в фоновом режиме, потребление CPU может иметь значение; вовсе ни к чему, чтобы сканирование директории поминутно тормозило работу вашего компьютера.

Наша подборка

- » BitTorrent Sync
- » OwnCloud
- » Rsync
- » Syncthing
- » Unison

В большинстве домов теперь имеется не по одному компьютерному устройству. И хотя очень удобно, когда можно проверить свою электронную почту дома — с компьютера, а вне дома — со смартфона, тут требуется доступность ваших данных на всех ваших устройствах. Облачные сервисы облегчают эту задачу, храня вашу почту, фото или музыку на центральном сервере, и доступ к облакам можно получить откуда угодно, но как насчет остальных данных? Или — если вы не хотите размещать свою информацию на облачном сервисе? Мы рассматривали разные онлайн-сервисы (см. Сравнение, стр. 22 **LXF180**), но здесь нас интересуют опции локального хранения. Не говоря уже о проблемах с конфиденциальностью — если ваши устройства подсоединены к одной сети, синхронизировать файлы, не скидывая их на облако, будет намного быстрее.

Существует множество опций, и некоторые из них требуют сервера наподобие онлайн-сервиса, хотя и работают локально, а другие предпочитают использовать для синхронизации ваших устройств скорее одноранговые [peer-to-peer] методы. Мы увидим, как наши пять кандидатов справляются с этой задачей и легко ли они могут делиться определенными типами файлов с другими. Не менее важный вопрос — кросс-платформенная совместимость, если только у вас в доме не используются исключительно устройства Linux, даже без смартфона на Android!

Установка и первые впечатления

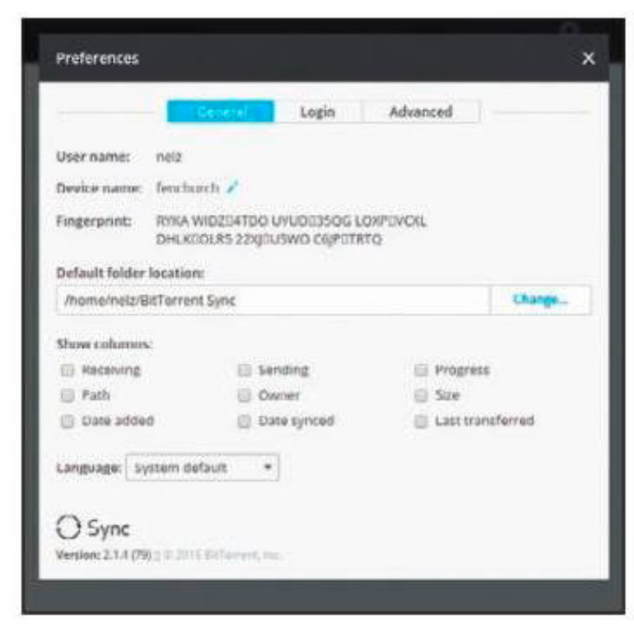
Просто ли будет установить и использовать каждую опцию впервые?

Есть смысл при любой возможности использовать для установки программ менеджер пакетов вашего дистрибутива; но некоторые участники нашего Сравнения являются пакетами, не ориентированными ни на какой дистрибутив. Проще всего обзавестись *Rsync*, *BitTorrent Sync* и *Syncthing* — первым потому, что в большинстве дистрибутивов он уже есть по умолчанию, а два остальных идут в виде tar-архива с исполняемой программой, готовой к запуску сразу после распаковки, причем *Syncthing* также предоставляет пакеты Debian/Ubuntu.

Unison поставляется в виде исходного кода для традиционного в Linux метода установки; или используйте менеджер пакетов. *OwnCloud* просто устанавливается в виде tar-архива, который вы распаковываете в директорию, доступную вашему web-серверу, однако здесь-то и кроется загвоздка: ему нужен web-сервер, и популярным выбором является *Apache*. Все остальные соперники общаются напрямую между компьютерами, а вот *OwnCloud* подавай этот сервер, и потом вы легко установите клиенты на всем остальном, хотя просматривать файлы можно и через браузер. Все это превращает начальную настройку в каторгу, хоть и хорошо документированную; правда, это каторга однократная. Настольный клиент *OwnCloud* по умолчанию синхронизирует весь сервер с директорией *OwnCloud* на вашем жестком диске. Можно синхронизировать директории и по отдельности, но сперва придется удалить настройки по умолчанию.

И *BitTorrent Sync*, и *Syncthing* используют web-интерфейс со своим собственным встроенным web-сервером. Когда вы запустите программу в терминале, она сообщит вам порт для соединения с localhost. Затем просто следуйте подсказкам для синхронизации папок. Естественно, вам придется сделать это как минимум на двух устройствах.

OwnCloud тоже настраивается через web-браузер, а *Unison* и инструмент *Rsync* используют в качестве протокола SSH. Если у вас уже настроен SSH — желательно с аутентификацией ключом, а не паролем — между устройствами, они должны просто работать: достаточно один раз сообщить им, что именно надо синхронизировать.



› *BitTorrent Sync* не нужно устанавливать: просто распакуйте tar-архив и запустите программу изнутри директории.

По умолчанию для *Unison* используется директория **home** на каждом компьютере, но это можно перенастроить созданием файлов профиля. *Unison* — единственная из всех наших программ, которая обладает традиционным настольным интерфейсом. Отсюда вы можете выбирать и настраивать профили синхронизации (это позволит осуществлять синхронизацию разных устройств или разных наборов файлов), запускать синхронизацию вручную, разрешать конфликты и читать руководство.

Вердикт

Rsync ★★★★★
Syncthing ★★★★☆
BitTorrent Sync ★★★☆☆
Unison ★★★☆☆
OwnCloud ★★☆☆☆

» За исключением OwnCloud, все программы просты в установке.

Приватность и безопасность

Конфиденциальны ли ваши данные, если не использовать облако?

Вы хотите сохранить конфиденциальность своих данных, и при этом хотите разместить их на разных устройствах, где бы они ни находились — как разные решения справятся с этой задачей? *Unison* и *Rsync* используют SSH для соединения, а значит, безопасны. Основной недостаток в том, что надо открыть порты SSH Интернету. Это безопасно, но вы сможете поместить за своим роутером только один компьютер — разве что вы любите мороку с лишними портами SSH (или у вас исключительно умный роутер). *OwnCloud* — web-приложение, поэтому может использовать HTTPS. Это усложняет настройку сервера, зато вы просто выставляете сервер в Интернет, а устройства соединяются с ним.

Syncthing и *BitTorrent Sync* используют одноранговые протоколы: они соединяются с центральным сервером обнаружения, чтобы там определиться, но сами файлы передаются между хостами. То есть вам незачем беспокоиться насчет IP-адресов или местоположений в сети: устройства просто находят друг друга. Они оба используют шифрованную передачу, но только *Syncthing* показывает исходный код, чтобы вы были уверены в том, что ваши данные не пойдут туда, куда не следует. Для максимальной безопасности можно запустить свой «узел обнаружения», что даст вам абсолютный контроль над своими данными. *BitTorrent Sync* умеет синхронизироваться с локальными устройствами и без центрального сервера, но для доступа к нему через Интернет придется включить UPnP в своем роутере, иначе он не будет работать.

› Можно велеть *Syncthing* не использовать центральный сервер обнаружения, для большей безопасности.

Вердикт

Rsync ★★★★★
Syncthing ★★★★★
Unison ★★★★★
OwnCloud ★★★★☆
BitTorrent Sync ★★★☆☆

» Три опции используют надежное шифрование по умолчанию.

Функции и простота в работе

Выберите то, чем хотите поделиться.

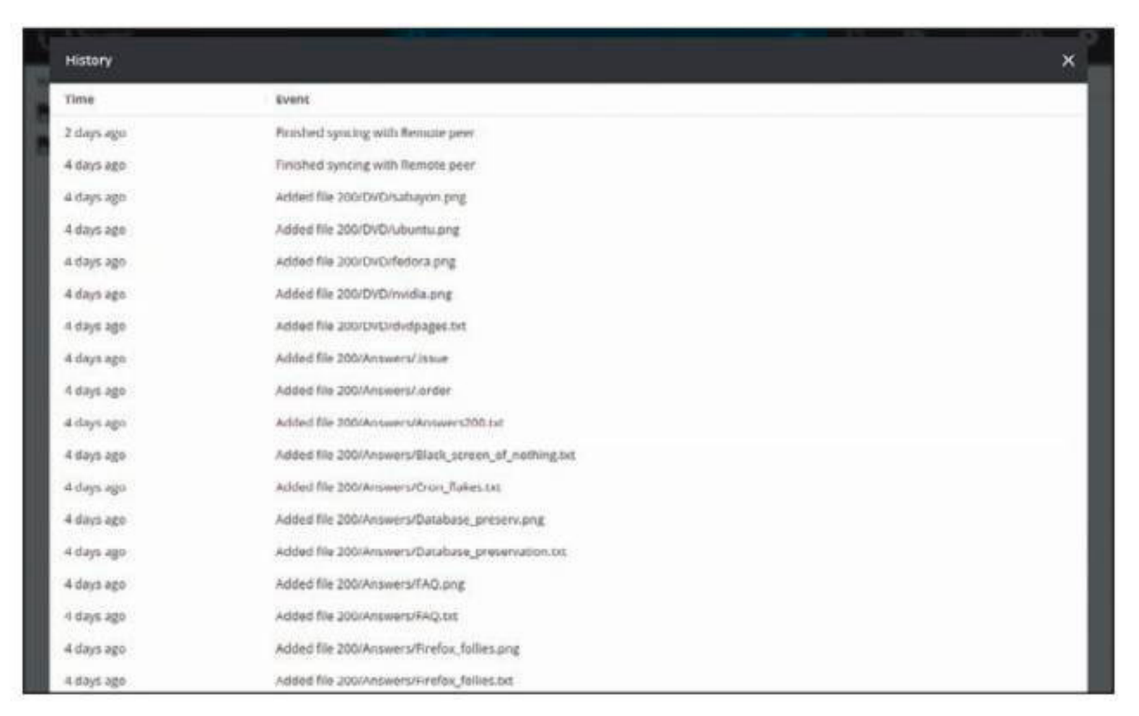
Как правило, синхронизировать всю вашу директорию **home** с разными устройствами незачем, особенно учитывая то, что копирование файлов настройки с одного ПК на другой может вызвать проблемы. От синхронизатора файлов вам требуется возможность выбрать определенные директории для синхронизации — желательно в неограниченном количестве, и быть в состоянии точно выбрать, какие именно файлы в этих директориях скопировать. Мы не жадные, но хотим, чтобы настройка тоже была простой, чтобы вы получили именно то, что вам надо. Одни наши программы позволяют создать список директорий и файлов для включения или не включения в web-интерфейс, другие требуют, редактирования текстового файла, а у третьих существуют обе эти опции.

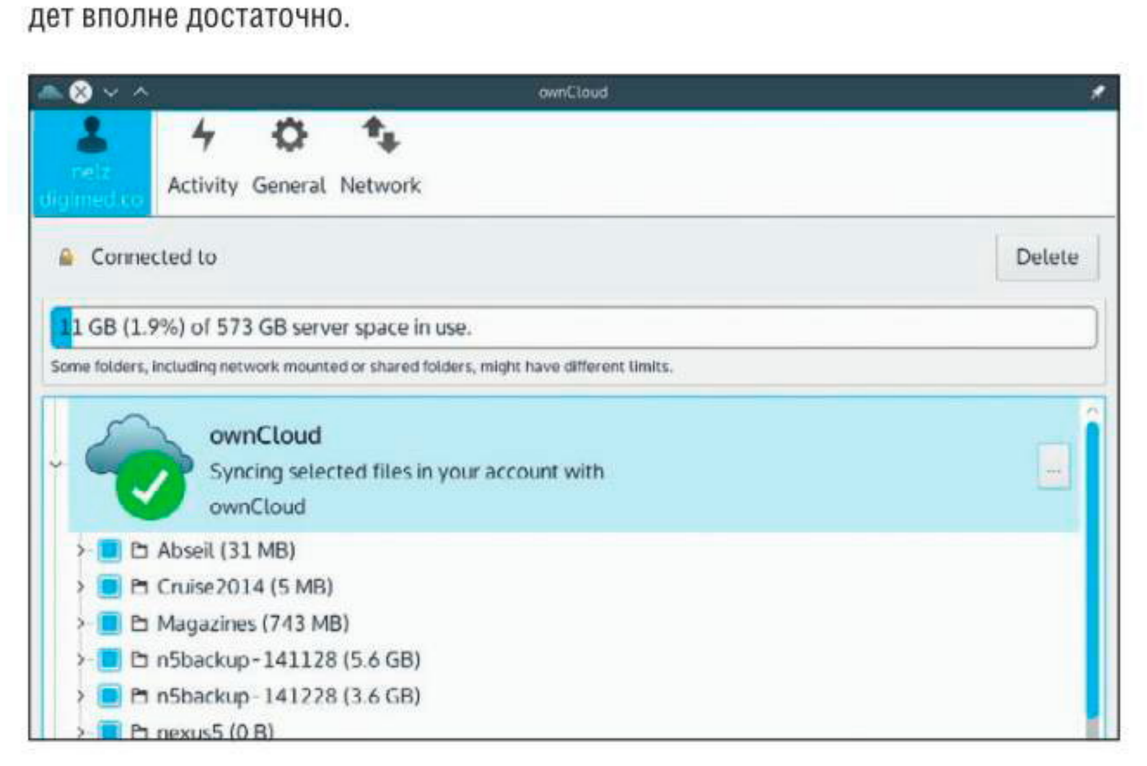
Ни один из подходов не является лучшим; это старый компромисс между удобством и контролем, и только вы сами в состоянии решить, как будет удобнее именно вам.

BitTorrent Sync ★★★★☆

BitTorrent Sync (он же *btsync*) во многом похож на *Syncthing*. Однако это коммерческий продукт с бесплатной пробной версией, имеющей ограничения. Настройка соединений в *btsync* даже проще, чем в *Syncthing*, хотя степень контроля над ним ниже.

Интерфейс *btsync* уж точно менее загружен, и вы можете настроить то, какая именно информация о соединениях вам будет показываться. Есть ссылки на онлайн-справку, но та, как часто бывает с коммерческими продуктами, злоупотребляет предложениями «Обновиться до уровня Pro». Здесь есть удобная функция History, которая перечисляет недавние дополнения и изменения. Тем, кто не хочет платить за функции уровня предприятия, *BitTorrent Sync* предлагает базовый набор функций, весьма простых в настройке. Если только вам не требуется синхронизировать более 10 папок, бесплатной версии будет вполне достаточно.





OwnCloud ★★★★☆

Самая сложная из программ в этом Сравнении — *OwnCloud*, и, как следствие, на привыкание к ней уходит несколько больше времени. Учитывая количество предлагаемых ею функций, она на удивление проста в ознакомлении, и вам не нужно использовать все и сразу. В отличие от навороченных серверов, клиенты рабочего стола просты и недвусмысленны в использовании.

Просто откройте клиент, чтобы добавить папку для синхронизации или изменить настройки, и львиную долю времени он будет тихо сидеть в системном лотке, обновляя ваши файлы и не путаясь под ногами. Поскольку *OwnCloud* создан для работы в качестве центрального сервера, можно создать несколько учетных записей, и у каждый из них будут свои директории с распределенным доступом и разрешениями на загрузку и скачивание — это особенно удобно для коллективной работы или иных задач, требующих предоставить другим возможность читать или делать записи в тех же файлах, что используете вы.

Разрешение конфликтов

Что происходит, если вы изменяете слишком много файлов?

Меняя файл на одном устройстве, вы ожидаете, что эти изменения перейдут и на другие устройства; ну, а если между сеансами синхронизации изменить тот же файл не на одном устройстве? Как справятся с этим программы — ведь в них нет модуля чтения мыслей? *Rsync* и *Unison* умеют синхронизировать только два устройства. *Rsync* строго однонаправленный: он обновляет пункт назначения, приводя его в соответствие с источником. *Unison*, использующий протоколы *Rsync*, сравнивает файлы с обеих сторон и копирует те, что менялись с последней синхронизации. Он ведет журнал действий, записывая, что было изменено, а в интерактивном режиме показывает все конфликты и все файлы, измененные с обеих сторон, чтобы вы указали, какой именно использовать. При пакетной обработкн он может пропустить их либо учесть ваши указания по приоритету устройств.

OwnCloud тоже однонаправленный: каждый клиент синхронизируется с сервером, хотя можно употребить таким образом несколько клиентов. Он обрабатывает конфликты, приписывая к имени файла **-conflict** и присваивая файлу временну́ю отметку. В *Syncthing* обычно нет концепции назначения и источника, хотя можно настроить его и на это — для конкретных директорий. Файлы синхронизируются на всех устройствах; если какое-то из них будет оффлайн (например, вашему телефону велено синхронизироваться только через Wi-Fi), оно догонит другие устройства после соединения. Конфликтные файлы переименовываются. *Btsync* тоже их переименовывает, но без временно́й отметки.

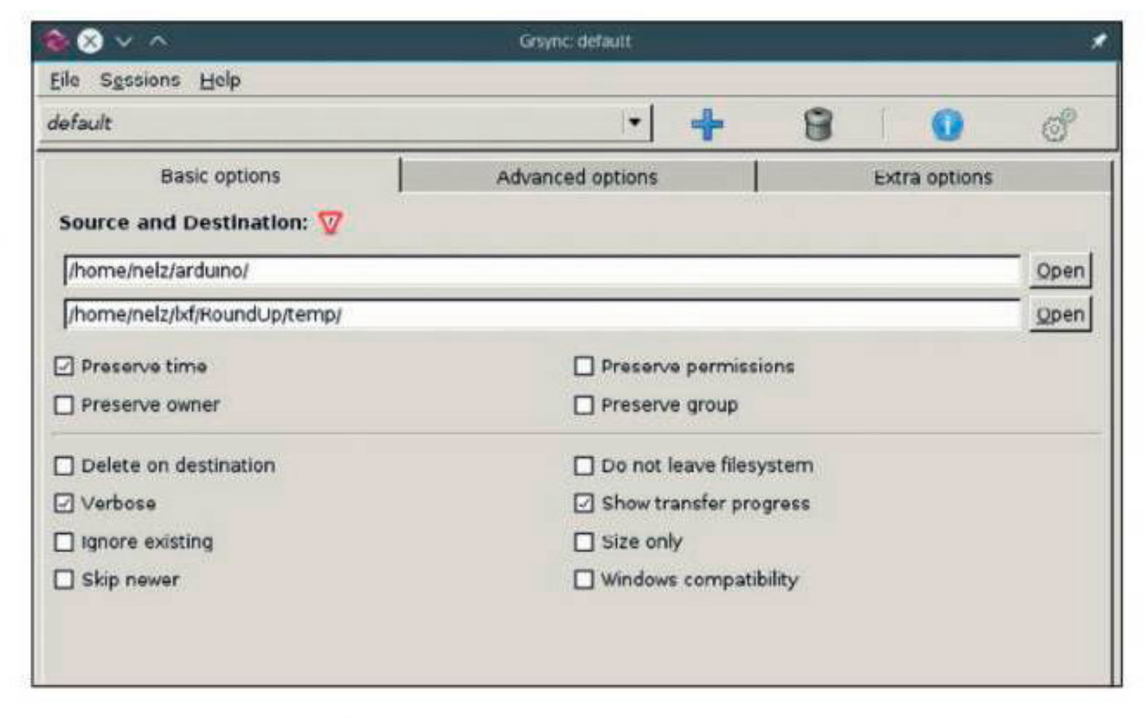
Вердикт

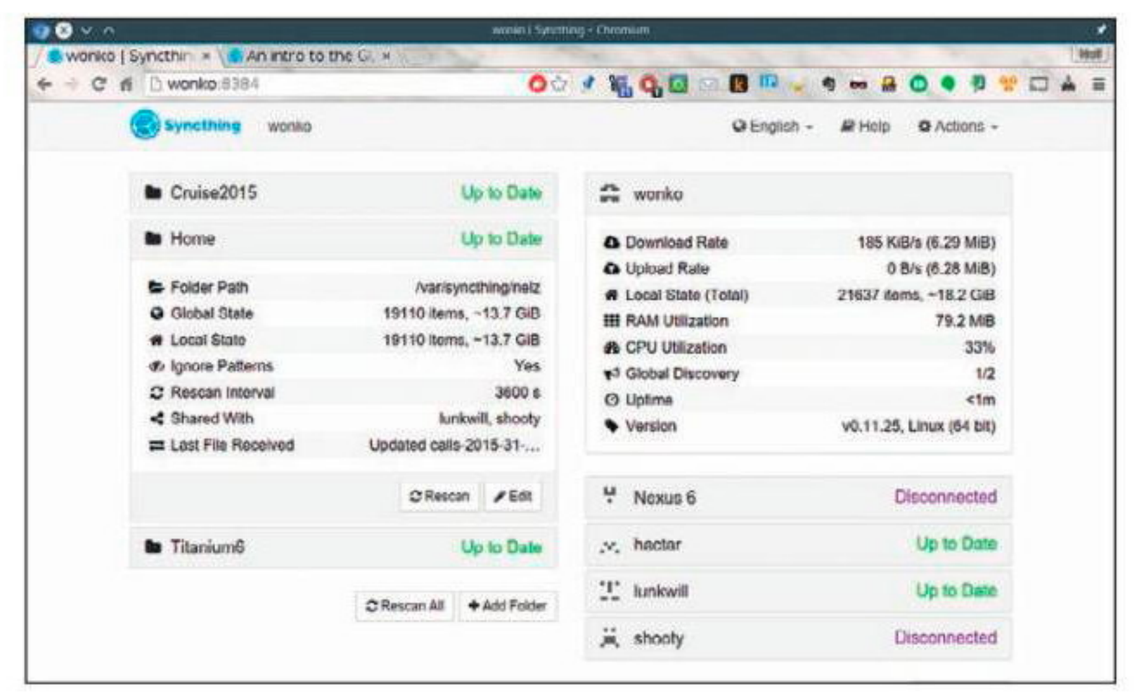
Unison ★★★★★
Syncthing ★★★★☆
BitTorrent Sync ★★★☆☆
OwnCloud ★★★☆☆
Rsync ★☆☆☆☆

» Ни один не идеален, но у Unison больше опций.

Rsync ★★★☆☆

Rsync имеет куда меньше функций, чем прочие программы Сравнения, но очень хорош именно в том, для чего и создан, то есть для односторонней синхронизации. Он очень эффективно использует полосу пропускания, и когда в большой файл вносится мелкое изменение, он отправит только это изменение. *Syncthing* умеет делать такое при добавлении к файлу, а вот *Rsync* справляется и с изменениями внутри файла. В нем нет собственного GUI, но есть богатый набор опций для управления и контроля за процессом синхронизации. GUI присутствует в виде *Grsync*, позволяя вам графически настроить пути и опции. Он может сохранять каждый набор настроек в виде сессии, что облегчает повтор каждого действия (немного похоже на профили *Unison*). Эти сессии можно запускать с помощью команды grsync-batch из скрипта или задачи *Cron*, и это значит, что вы наслаждаетесь удобством GUI для настройки и простотой синхронизации благодаря *Cron*.



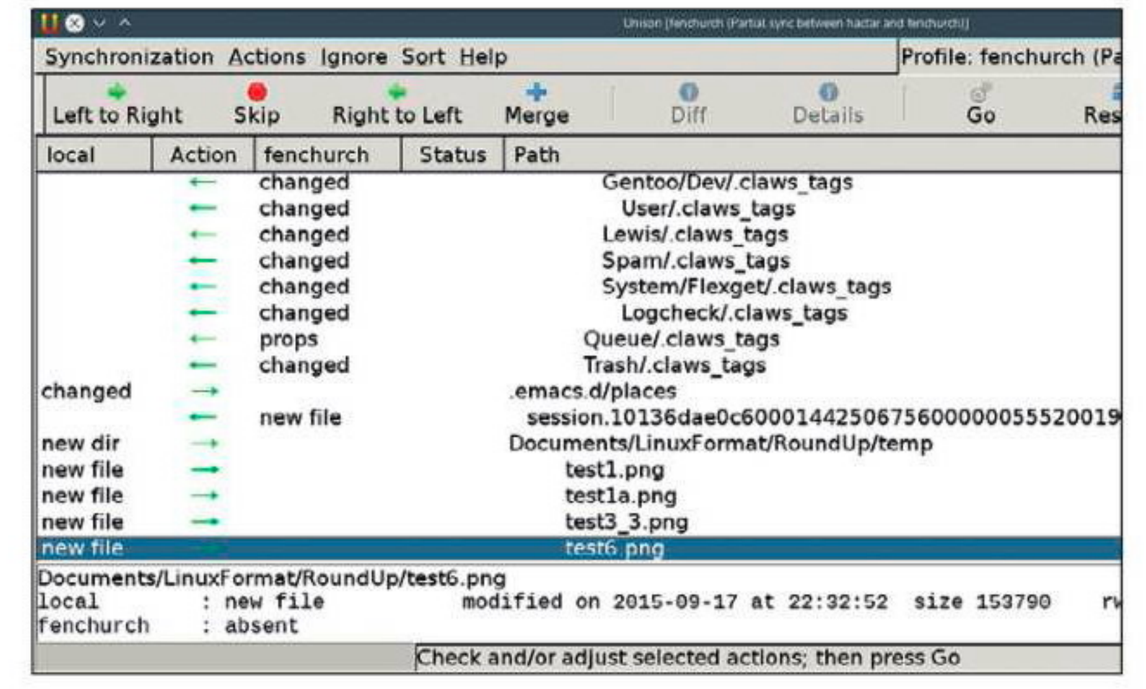


Syncthing ★★★★☆

Syncthing полностью управляется из web-браузера. Здесь вы добавляете устройства для соединения и директории для синхронизации. По умолчанию синхронизируется все, что есть в директории, но есть опция создания списка исключений, который делается в браузере или посредством редактирования простого файла. Настройка нескольких устройств с разными условиями синхронизации не представляет сложности, и вся синхронизация происходит автоматически. Каждая директория имеет собственный промежуток между сканированиями — по умолчанию это несколько минут, но обычно лучше настроить его на подольше. Контент синхронизируется на всех устройствах в списке для директории, однако можно настроить одно устройство в качестве мастера для этой директории. При выборе опции версий файлов можно сохранять более старые или удаленные копии файлов — например, во избежание всяких неприятностей очень удобно сохранять последние три копии каждого файла.

Unison ★★★☆☆

Unison настраивается через редактирование текстового файла, т.н. профиля. В простейшем виде он содержит локальную директорию и адрес SSH для синхронизации, и есть мастер GUI для создания профиля. Он также может содержать особый файл или части директории для включения или исключения; всё это снабжено хорошей документацией. Разрешается редактировать этот файл из GUI; можно выбирать пункты, отмеченные для синхронизации, и добавлять их в список исключений из GUI, но редактирование вручную обеспечивает максимум контроля. Unison разрешает наличие только одной пары директорий в профиле, но может быть несколько профилей, загружаемых по очереди, из GUI или из командной строки. GUI в основном применяется для синхронизации вручную, и с этой задачей справляется превосходно. Опции командной строки включают пакетный режим, пригодный для автоматической синхронизации директорий из планировщика задач, такого, как *Cron*.



Кросс-платформенное применение

Да, надо подумать и о пользователях Windows или iBling...

Линус пока что не достиг мирового господства, так что нам по-прежнему надо быть в состоянии делить ресурсы с другими операционными системами. Для начала, хотя *Rsync* доступен для Linux и Mac OS и есть порты в Windows и Android с разными уровнями совместимости, но это не чистый *Rsync* (если только не считать возможности работы с ним через Cygwin). *Unison* доступен для всех трех настольных ОС, и это отличный выбор для синхронизации файлов во внутренней сети. *Syncthing* доступен для трех настольных ОС плюс Android (в Play Store есть два клиента Android с очень похожими наборами функций), и недавно появилась бета-версия клиента iOS.

Сервер *OwnCloud* является web-приложением PHP, хотя его работа в Windows не поддерживается, а клиенты есть для Linux, Mac OS, Windows, Android, iOS и даже Blackberry (помните такие?). У *BitTorrent Sync* столь же впечатляющий охват, за исключением того, что он поддерживает Windows Phone, а не Blackberry. У всех мобильных клиентов предусмотрен базовый интерфейс с такими опциями, как выбор синхронизации только по Wi-Fi для сохранения изменений данных. *Syncthing* и *BitTorrent Sync* могут использовать камеру вашего телефона для добавления данных из QR-кода, отображаемого на рабочем столе; *Syncthing* к тому же позволяет вам выдавать распределенный доступ к папкам на мобильных устройствах с помощью QR-кода или ссылки. Чтобы добавить клиент в *OwnCloud*, надо ввести URL, имя пользователя и пароль, однако это одноразовая задача, так что особо жизнь не осложняет.

Вердикт

BitTorrent Sync ★★★★★
OwnCloud ★★★★☆
Syncthing ★★★★☆
Unison ★★★☆☆
Rsync ★★☆☆☆

» BitTorrent Sync охватывает все популярные системы.

Дополнительные функции

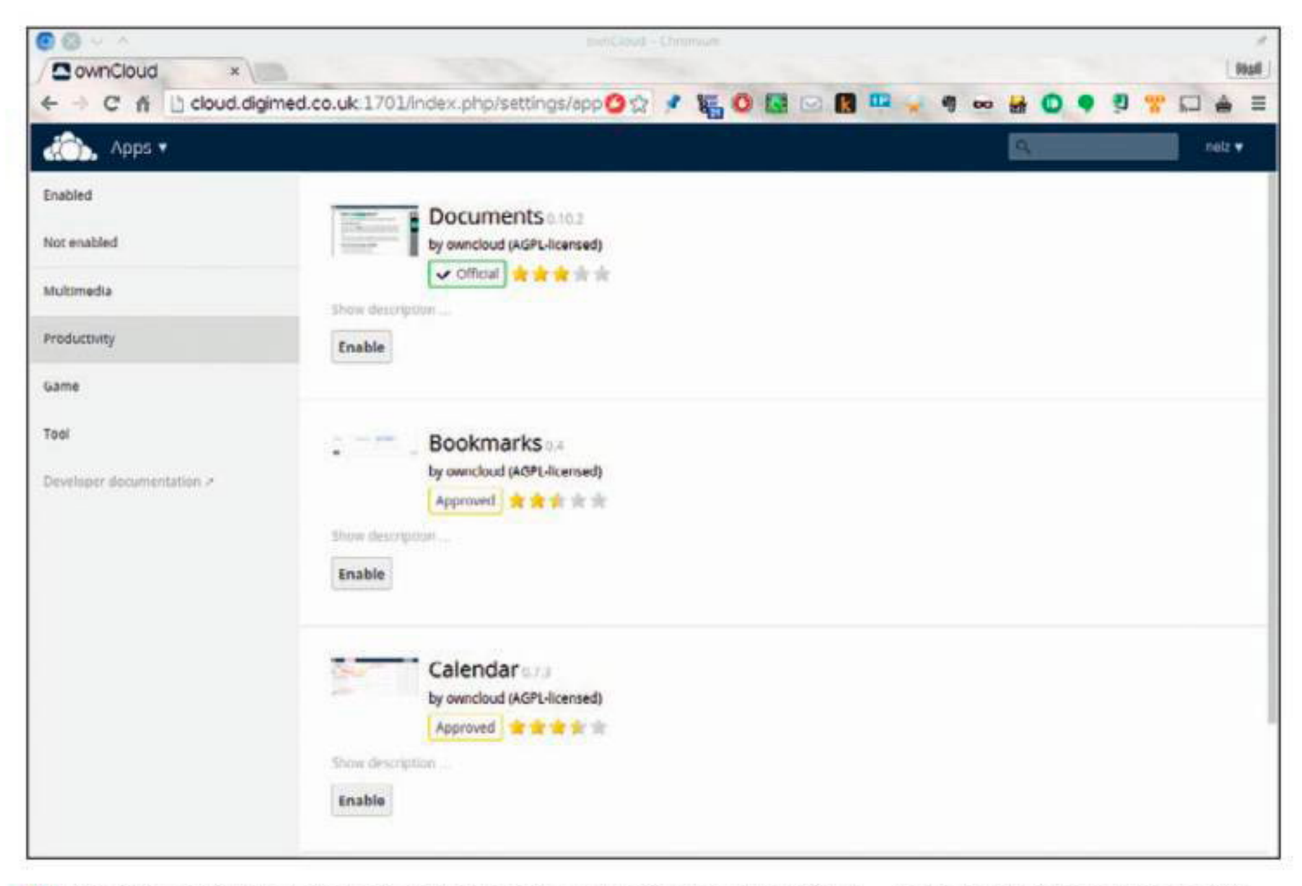
Все они справляются с базовыми задачами; а что еще умеют делать?

Все эти программы позволяют синхронизировать одно устройство с другим, что мы здесь и рассматриваем; но некоторые предлагают добавочные функции, способные сделать их более подходящими для вас.

BitTorrent Sync (*btsync*) облегчит вам кооперативное пользование отдельными директориями, и только на чтение, если вам нужно именно это, посредством предоставления ссылки, которую можно отправить по электронной почте другому пользователю или отобразить в виде QR-кода, что облегчает совместное использование с телефона. Отправка ссылки электронной почтой не вполне безопасна, поэтому вам придется подтвердить соединение при использовании ссылки. Бесплатная версия *btsync* позволяет синхронизировать только 10 директорий, и в ней отсутствуют некоторые функции версии Pro стоимостью $40 в год.

Syncthing также имеет опцию отправки информации по соединению через QR-код, и папки имеют отметки, так что для настройки совместного использования нет необходимости в URL, просто надо создать отметку при установке соединения; но сначала следует соединить два устройства.

OwnCloud по части дополнительных функций неподражаем. Это полноценный облачный файловый сервер, который вы запускаете сами, поэтому можете делиться доступом к файлам и директориям с кем угодно, позволяя другим загружать и запускать другие приложения на сервере, например, календарь. Если вам надо всего лишь синхронизировать несколько директорий, то он будет перебором, но если нужны дополнительные возможности и при этом вы хотите сами всё контролировать, то именно на него стоит взглянуть в первую очередь. *Unison* и *Rsync* — в чистом виде синхронизаторы файлов, простые и незамысловатые.



› В *OwnCloud* функций наберется на отдельную статью [*Ред.*: — Это тонкий такой намек?], но вот часть его приложений.

Вердикт

BitTorrent Sync ★★★★★
OwnCloud ★★★★☆
Rsync ★★★☆☆
Syncthing ★★☆☆☆
Unison ★★☆☆☆

» OwnCloud делает намного больше, это полнофункциональный облачный сервер.

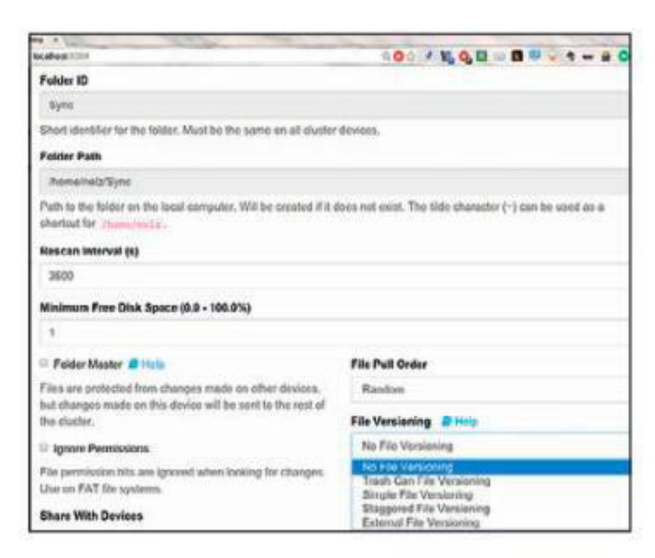
Автоматизация и скорость

Быстрая работа в фоновом режиме.

Чтобы ваши файлы всегда были под рукой, нужна автоматическая работа программ синхронизации; крайне неприятно сесть в поезд и вспомнить, что вы не скопировали файлы со своего ПК.

Syncthing работает в фоновом режиме, сканируя директории с интервалами, которые вы установили, своими для каждой директории, и передавая изменения. *BitTorrent Sync* использует для запуска синхронизации уведомления файлов, что может привести к лишней нагрузке на систему, если нечто затрагивает слишком много файлов в директории, даже и не меняя их содержимого. Есть и регулярное сканирование директории, но интервал его нельзя настроить. Клиент *OwnCloud* тоже использует уведомления файлов, чтобы решить, когда проводить обновление. В *Unison* встроенная автоматизация отсутствует; вы имеете возможность запускать его GUI как интерактивный инструмент или вызывать в режиме пакетной обработки из *Cron*. *Rsync* работает из командной строки, хотя имеется *Grsync*, сторонний GUI, его тоже надо запускать из *Cron* для автоматической работы. Это означает исследование имеющихся опций в man-странице, хотя `rsync -a SRC DEST` часто бывает достаточно. В порядке альтернативы, можно создать сессию в *Grsync* и вызвать `grsync-batch` из *Cron*, чтобы запускать то же самое обновление через определенные промежутки времени.

Передача файлов ограничена скоростью вашей сети; поэтому уменьшение количества передаваемых данных оказывает колоссальное воздействие на производительность. *BitTorrent Sync* делит файлы на части. Это способно повысить эффективность работы с изменяемыми файлами. *Syncthing* использует собственный открытый протокол Block Exchange Protocol, который тоже делит файлы для большей эффективности работы. *Unison* использует протокол rsync, который демонстрирует колоссальную эффективность при работе с измененными файлами. Очевидно, что это применимо и к самому *Rsync*. А вот *OwnCloud* использует простые HTTP или HTTPS, вынуждая пересылать измененный файл целиком и полностью. И если вы, например, изменяете большие файлы, это сильно замедлит работу.



› В *Syncthing* можно контролировать, как архивировать старые или удаленные файлы.

Вердикт

BitTorrent Sync ★★★★★
Syncthing ★★★★★
OwnCloud ★★★★☆
Rsync ★★☆☆☆
Unison ★★☆☆☆

» Syncthing и BitTorrent Sync просто работают автоматически.

Синхронизация файлов

Вердикт

Мы легко можем составить рейтинг программ по их общим функциям и простоте использования, однако у каждой есть свои сильные стороны. Нет даже никакой необходимости выбирать только одну из опций: для синхронизации своих ПК и ноутбука вы можете захотеть использовать одни методы, а для телефона — другие.

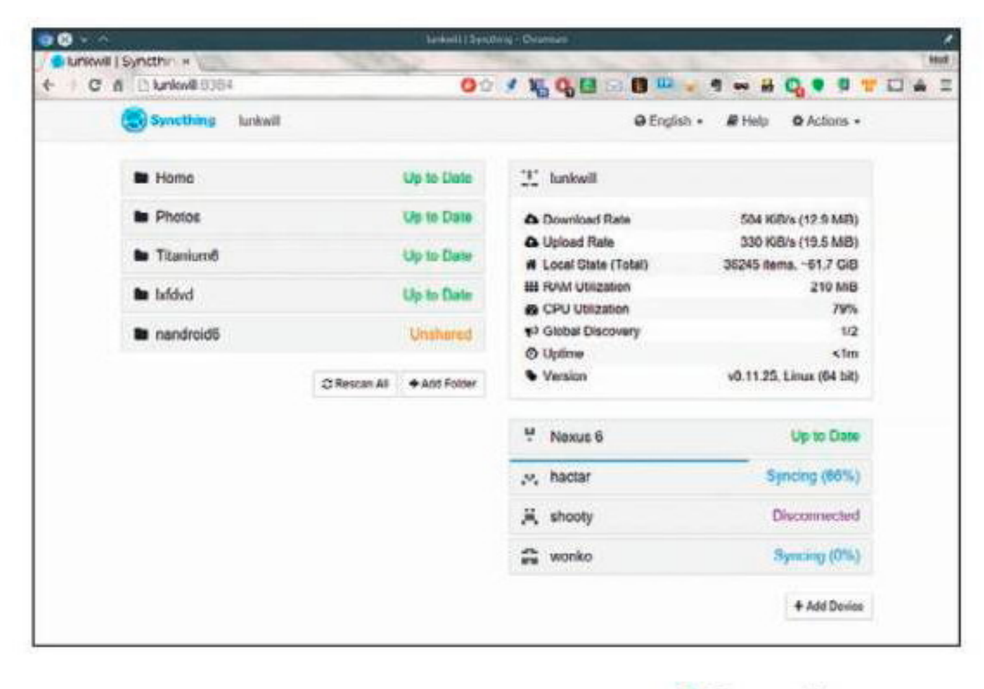
Если вам нужно всего лишь копировать файлы с одного компьютера на другой, то хорошим выбором будет *Rsync*, с или без *Grsync* в качестве GUI; однако в плане автоматизации он отстает, а с двусторонней синхронизацией не работает вообще.

Unison отлично справляется с задачей синхронизации двух компьютеров, отслеживая изменения на обеих системах и информируя вас о них. Для интерактивного использования он подходит лучше всех прочих. Некоторым отсутствие в нем автоматизации может показаться недостатком, а отсутствие мобильных клиентов сразу его вычеркивает, если вы хотите хранить контент еще и на смартфоне.

BitTorrent Sync работает очень хорошо, он прост в установке и настройке и имеет достойный набор функций и опций. Но — и это довольно солидное «но» — среди наших кандидатов это единственная программа с закрытым кодом. Для вас это может быть важно или неважно с этической точки зрения, но об этом определенно стоит помнить, если на ваших устройствах имеются важные данные и вам нужна заведомая безопасность.

OwnCloud обладает массой функций и нуждается в web-сервере, а значит, вам, вероятно, постоянно будет нужен компьютер онлайн. Настольный и мобильный клиенты легковесны и ненавязчивы, однако вся экосистема в целом немного чрезмерна, если все, что вам нужно — это простая синхронизация файлов. Однако если вам требуется больше, то в *OwnCloud* всего полно, и он может стать отличным выбором.

Syncthing как проект относительно юн, но уже завел солидный набор функций, работающих очень хорошо. Обновления выходят часто, и разработка ведется быстрыми темпами. Он тихо работает в фоновом режиме, и, однажды настроив его, вы можете о нем не беспокоиться: просто предоставьте ему делать свое дело, а это он отлично умеет.



› При всей своей молодости, *Syncthing* — впечатляющая программа и сильно облегчает задачу синхронизации устройств.

Syncthing относительно юн, но уже завел солидный набор функций.

I Syncthing ★★★★★

Сайт: https://syncthing.net **Лицензия:** MPL 2 **Версия:** 0.11.25

» Быстр, прост и удобен в использовании.

II Bit Torrent Sync ★★★★☆

Сайт: www.getsync.com **Лицензия:** Проприетарная **Версия:** 2.1.4

» Почти так же хорош, как Syncthing, но, увы, с закрытым кодом.

III OwnCloud ★★★★☆

Сайт: https://owncloud.org **Лицензия:** AGPL 3 **Версия:** 8.1.3

» Для обычной синхронизации файлов — перебор, а вот если нужны функции — штука стоящая.

IV Unison ★★★☆☆

Сайт: http://bit.ly/UnisonFileSynch **Лицензия:** GPL 2 **Версия:** 2.48.3

» Отлично для передачи с компьютера на компьютер, но не для мобильного использования.

V Rsync ★★★☆☆

Сайт: https://rsync.samba.org **Лицензия:** GPL 3 **Версия:** 3.1.1

» Хорошо справляется со своей работой, но не идеален для синхронизации между устройствами.

Обратная связь

А какая ваша любимая программа синхронизации файлов? Мы бы хотели узнать о вашем опыте. Пишите нам на lxf.letters@futurenet.com.

Рассмотрите также...

Выбор программ для синхронизации файлов не так велик, как выбор, скажем, медиаплейеров или текстовых редакторов, где каждый пользователь найдет именно то, что соответствует всем его нуждам. Тем не менее, есть еще несколько программ синхронизации для рабочего стола, таких, как *csync* (http://csync.org), которая не вошла в наш список, но определенно достойна внимания. Это двусторонний client-only синхронизатор файлов, обеспечивающий «перемещаемые директории» для синхронизации всего, что может потребоваться «обычному пользователю», например, коллекций мультимедиа и резервных копий.

Немало также мобильных утилит, которые соединяются с вашим домашним компьютером, если ваша основная забота — синхронизировать свой телефон с компьютером: например, для Android мы использовали *FolderSync* и *Titanium Media Sync*. Конечно, при решении подобных задач многих вполне устроят облачные сервисы, но для тех, кого заботят скорость и безопасность, есть достаточно опций для удовлетворения их потребностей, и они смогут найти то, что будет работать быстро и отличаться простотой в использовании. LXF

ДОМАШНИЙ МЕДИА-РАЙ

Выбросьте все свои DVD и CD — пришла пора разместить вашу медиа- библиотеку в таком месте, чтобы она была всегда под рукой. Ник Пирс создает центральный медиа-сервер.

Медиа-сервер вам нужен, если вы фанат своих мультимедиа — кино, ТВ, музыки, фото или чего бы то ни было еще. Это центральный файловый сервер, который дает вам полный контроль над всеми вашими мультимедиа, облегчая просмотр и прослушивание с любого устройства, имеющего доступ к Интернету — вы больше не привязаны к своей гостиной и к своему дому в целом. И, что еще лучше, не требуется ни малейших телодвижений по поводу обмена DVD и CD либо обязанности просматривать ненужные трейлеры.

Все, что нужно для вашего медиа-сервера — это подходящий компьютер с Linux. И хотя вполне возможно поселить медиа-сервер на вашем основном ПК, все же намного лучше отвести для этого отдельный компьютер. Главной вашей проблемой будет выбор медиа-сервера — но не переживайте, мы уже сделали за вас всю тяжелую работу и нашли решение с открытым кодом, которое отличается простотой в использовании и при этом достойно выглядит и несложно в навигации.

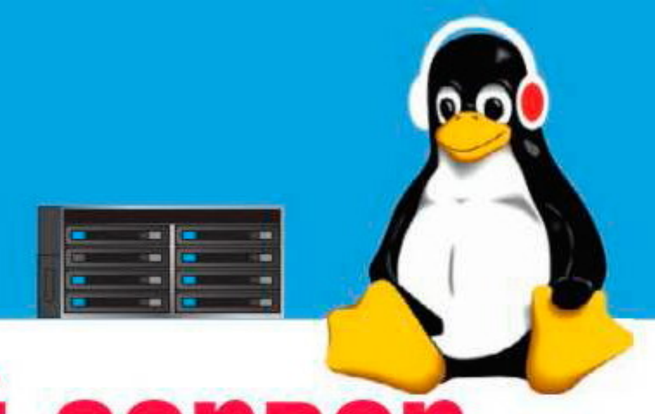
Мы покажем, как настроить его для работы, разместив на нем все ваши мультимедиа, а потом объясним, как расширить его функции — например, вы сможете добавить live TV, вывести потоковую музыку на беспроводные колонки у себя дома, получить доступ к своим мультимедиа за пределами своей локальной сети и удаленно поделиться ими с другими. Вы даже можете настроить несколько пользователей, разрешив детям доступ к их любимым программам, но не давая им добраться до неподходящих для юношества.

> **Медиа-сервер дает контроль. Наслаждайтесь на любом устройстве и везде.**

Как только ваш сервер будет настроен и примется работать и делать все, чего вы от него ждете, мы сосредоточимся на доступе к нему с других устройств. Вы с радостью узнаете, что ваш сервер поддерживает не только другие компьютеры, но также мобильные устройства, планшеты, smart TV, приставки и прочее. В большинстве случаев у вас будет точно такое же визуально богатое и информативное изображение.

Одно можно утверждать наверняка: стоит вам начать управлять своими мультимедиа через сервер на Linux, вы вряд ли когда-нибудь захотите вернуться к старому способу работы с ними. Читайте дальше, чтобы узнать, как запустить революцию мультимедиа.

www.linuxformat.ru/subscribe



Выберите свой сервер

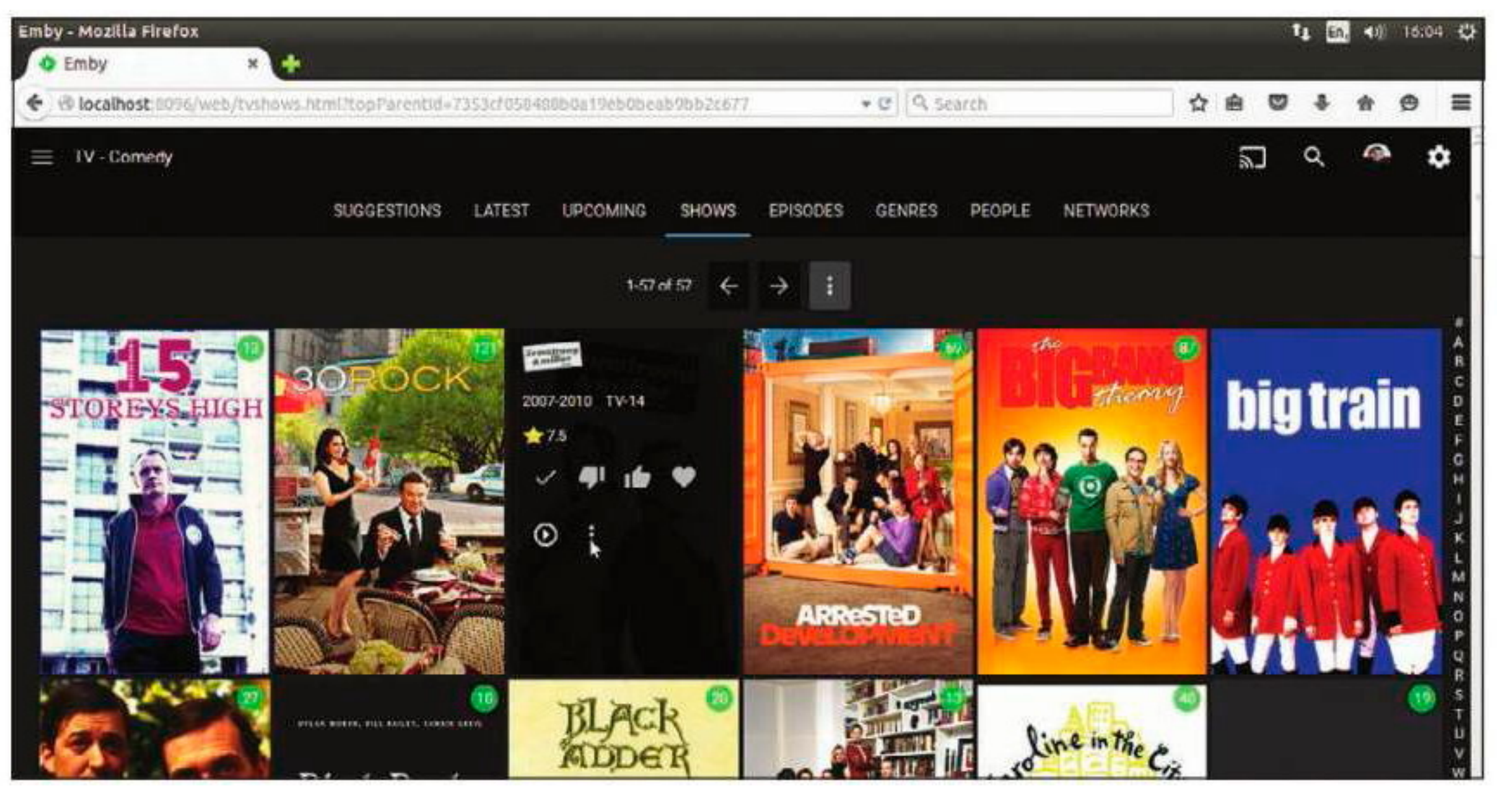
Во-первых, какое взять оборудование и ПО, чтобы делиться своими мультимедиа?

Хотя вполне возможно просто поставить медиа-сервер на вашем настольном ПК, его запросы могут помешать вашей обычной работе. Кроме того, вы можете захотеть, чтобы он работал круглосуточно, давая доступ к себе вам (и другим) без необходимости постоянно и специально его загружать. Так что имеет смысл выделить под эту задачу отдельную машину.

Требования медиа-сервера бывают довольно обременительны, в зависимости от того, надо ли будет перекодировать медиа-файлы для других устройств. Перекодирование — это конвертирование файла из его родного формата в формат, совместимый с устройством, где вы собираетесь его воспроизводить. Как показывает опыт, если медиа-файлы не в универсальном формате типа MP4 (с видеокодеком H.264 и аудиокодеком AAC или MP3), тогда, вероятно, для некоторых устройств понадобится перекодирование.

Если способности к перекодированию для вас не требуются, то подойдет довольно скромная машина (см. врезку про медиа-сервер на Raspberry Pi внизу). Но если нужна максимальная гибкость при минимальных проблемах доступа к вашим мультимедиа откуда бы то ни было, тогда ищите компьютер (возможно, старый, которым вы уже не пользуетесь) с разумными спецификациями: 2-ГГц двуядерный процессор будет идеальным выбором и позволит вам отделаться более дешевой видеокартой. Если процессор слабее, тогда ваш графический чипсет должен поддерживать декодирование видео на разогнанном оборудовании. Все карты Nvidia, начиная с GeForce 8, справятся с этой задачей, как и AMD Radeon R700 (HD4000) и Intel GMA X4500HD.

Вы также должны по возможности обеспечить свой сервер 4-ГБ ОЗУ, и если вы планируете выключать свой сервер, а не заставлять его работать круглосуточно, то подумайте о том, что надо бы вложить средства в загрузочное устройство SSD — или в быструю SD-карту, или USB-диск; 32 ГБ будет вполне достаточно для ваших целей. Для хранения ваших мультимедиа столько, конечно, не хватит, поэтому храните их отдельно: либо внутри — на выделенном жестком диске, либо на внешнем устройстве USB или NAS.



› Разместив свои мультимедиа, вы сможете напрямую просматривать их из web-интерфейса *Emby*.

Если вы не можете напрямую соединить свой сервер с роутером, используйте HomePlugs 500 Мб/с или более быстрый, чтобы обеспечить максимальную скорость передачи данных между ними, особенно для HD-контента.

Выберите свой медиа-сервер

Выбор медиа-серверов очень широк — в прошлом мы были зациклены на *Kodi*, однако его серверные возможности лишь зачаточные, и это его главная слабость. Очевиднейшей альтернативой для тех, кто хочет обеспечить себе быструю и приятную работу с широким диапазоном устройств, является *Plex* (http://plex.tv) — он хорошо работает, но беда в том, что большая его часть проприетарна. К счастью, у *Plex* есть соперник, который предлагает те же функции, но при этом относится к открытому коду. Этот соперник — *Emby* (http://emby.media), который ранее именовался *Media Browser* (вы поняли, в чем прикол?).

Emby предлагает те же функции, что и *Plex*, и доступен на ряде платформ. У него даже есть ряд преимуществ перед своим соперником: поддержка live TV плюс возможность настройки нескольких пользователей, которая позволяет настроить членам вашей семьи и друзьям собственный доступ к вашей библиотеке. Как и *Plex*, он предлагает опцию платной регистрации с дополнительными функциями, однако его цена более конкурентоспособна. Здесь мы сосредоточимся на бесплатных функциях, но позднее коснемся и преимуществ, предлагаемых регистрацией.

В отличие от *XBMC*, специальных дистрибутивов *Emby* пока не существует, и вам придется установить его поверх своего любимого дистрибутива Linux. В данной статье мы будем использовать Ubuntu 14.04.3 LTS — это самый стабильный и поддерживаемый дистрибутив для пользователей *Emby*.

»

Медиа-сервер на Raspberry Pi

Четырехъядерный Raspberry Pi 2 вполне способен стать достойным медиа-сервером — но не без ограничений. Основное из них — нехватка мощности для перекодирования из одного формата в другой. Так что лучше всего Pi будет работать, если форматы всех ваших мультимедиа совместимы с устройствами, на которых вы планируете их воспроизводить. В большинстве случаев идеальным будет использование видеофайлов MP4 с кодеками H.264/ MP3 плюс музыкальных файлов MP3.

Pi 2 может выполнять ограниченное перекодирование при подготовке файла к стримингу, а значит, у вас должен быть доступ к вашим медиа даже извне вашей локальной сети; но во время перекодирования готовьтесь к солидным задержкам — более минуты. Задержки будут наблюдаться и при доступе к серверу, но это не фатально; мы полгода успешно работали на Pi 2 с медиа-сервером *Plex* без особых проблем. Кто решил пойти этим путем, изучите руководство от TechRadar на http://bit.ly/pimediaserver: там объясняется, что надо сделать для настройки сервера с помощью индивидуального образа Minibian от HTPC Guides (www.htpcguides.com). Это, видимо, простейший способ добыть полноценный медиа-сервер на Pi 2, и поддержка *Plex* лучше, чем *Emby*; а если вам все-таки нужен *Emby* — возможно и это, хоть и будет посложнее. Руководство HowTo по его использованию с дистрибутивом OSMC см. на http://bit.ly/lxf-emby-install, и там же — бонус (получение медиа-центра вместе с медиа-сервером); но автор признает, что многие из функций — такие, как стриминг и интеграция с *Kodi* — не протестированы, поэтому не исключены проблемы. Некоторые пользователи сообщали об успешной работе *Emby* с Arch Linux — одно из руководств находится здесь: http://bit.ly/lxf-emby-arch.

Установка и настройка

Установим *Emby* и настроим свои библиотеки мультимедиа.

До начала работы позаботьтесь выделить своему серверу статический IP-адрес. Тогда он будет одним и тем же при каждом обращении к вашему серверу через web-браузер или SSH. В Ubuntu для этого откройте System Settings и выберите Network. При соединении через кабель Ethernet выберите Wired, нажмите Options и задайте статический адрес во вкладке IPv4 Settings.

Далее зайдите на http://emby.media/downloads/linux-server, где есть инструкции по установке для конкретных дистрибутивов. Во время установки вас попросят подтвердить создание пользователя с именем emby — при необходимости предоставьте этому пользователю доступ read/write к вашим папкам с мультимедиа. Затем *Emby* начнет работу.

После установки *Emby* пора рассмотреть, как организованы ваши мультимедиа-файлы. Вы можете хранить свои мультимедиа, где хотите — если только диск доступен для вашего медиа-сервера и монтируется автоматически при запуске. Перейдите на http://bit.ly/pimediaserver (*и см. стр. 29, если вам понадобится помощь на этой стадии*).

Emby отыщет в Интернете изображения и информацию о фильмах, телепередачах и музыке. Для более точного соответствия надо свои файлы правильно называть и структурировать. Начните с создания отдельных папок для ваших мультимедиа: **Музыка** [Music], **Фильмы** [Movies], **ТВ** [TV], **Фото** [Photos], **Клипы** [Music Video], и т.п. Далее, используйте соглашение об именах в таблице (хотя если вы задавали имена файлов для других медиа-серверов, таких, как *Kodi* или *Plex*, они тоже отлично будут работать). Если вы хотите ускорить переименование файлов, загляните в Filebot. Пользователи Ubuntu могут установить «родную» версию за $9,95 через *Software Centre*; в противном случае вам придется установить Java 8 (используйте репозиторий ppa:webupd8team/java, если JRE 8 нет в *Ubuntu Software Centre*), и затем скачайте и установите соответствующий Deb-пакет с www.filebot.net. Чтобы в первый раз запустить его, откройте окно *Terminal*, введите filebot и нажмите Enter (а дальнейшие запуски будут выполняться через посредство ярлыка программы).

Начните с установки формата имени, который должен соответствовать требованиям *Emby*: нажмите Match и выберите Edit Format. Проверьте, что Episode Format для телепрограмм установлен в {n} - {s00e00} - {t}, затем нажмите кнопку в нижнем левом углу, чтобы переключиться на Movies. Измените всё это на {n} ({y}) и нажмите Use Format. Выберите Match > Edit Format снова, и опять нажмите на кнопку, чтобы переключиться на Music. Проверьте, чтобы установки были {artist} - {t}, или измените их на {t}, если вы предпочитаете запись файла только по названию: для *Emby* годится и то, и другое.

Нажмите Use Format для завершения.

Тип	Структура папки	Синтаксис	Пример
Music	**Music\Artist\Album**	00. track name *or* track name	Music\a-ha\Cast In Steel\01. cast in steel.mp3
Movies	**Movies**	title (year)	Movies\star trek (2009).mkv
TV shows	**TV\Show Title\Season**	tvshow — s01e01	TV\Silicon Valley\Season 2\silicon valley s02e10.mkv

Теперь опять нажмите Match, но на сей раз выберите одну из предлагаемых баз данных согласно мультимедиа, которые вы переименовываете — рекомендуем TheTVDB для ТВ и TheMovieDB для фильмов, и попробуйте AcoustID для неизвестных музыкальных файлов. Предлагаемые для каждого файла названия будут отображаться в правой панели; если они вас устраивают, нажмите Rename, чтобы изменить имя файла.

Настройте свой медиа-сервер

Дав должные названия своим медиа и установив и настроив *Emby*, приступайте к стадии их знакомства друг с другом. Откройте браузер и введите localhost:8096 для доступа к web-интерфейсу *Emby*. Начните с мастера настройки — он достаточно прост: выберите страну, потом настройте свою персональную учетную запись. *Emby* поддерживает несколько пользователей, позволяя настроить родительский контроль при индивидуальном доступе в соответствии с личными предпочтениями детей. Вам также будет предложено настроить учетную запись Emby Connect — это обязательное условие, если вы планируете доступ к своим мультимедиа извне своей локальной сети.

Теперь настройте свои медиа-библиотеки. Каждая библиотека содержит контент определенного вида: музыку, фильмы, ТВ, другие типы видео или фото. Мастер настройки пока что позволяет вам добавить одну библиотеку: нажмите Add media folder, когда вам предложат это сделать, и выберите тип контента — музыку, фильмы, ТВ, книги, игры, домашнее видео, клипы, фото — вам предлагается всё это и еще особая опция смешанного контента. Дайте своей библиотеке подходящее описательное имя и нажмите OK. Далее выберите папку, где будет храниться ваш контент — *Emby* позволяет добавлять в библиотеку несколько папок, но если вы последовали нашему совету, все они будут сосредоточены в одном месте. Нажмите +1, чтобы просмотреть и выбрать его.

На следующем этапе вам надо заняться разными настройками метаданных — это информация и изображения, которые *Emby* использует, чтобы помочь вам идентифицировать свои мультимедиа. Вы можете выбрать хранение данной информации непосредственно в ваших медиа-папках или централизованно на вашем сервере — выберите опцию, оптимальную для вас и доступную вашему серверу по объему хранения данных.

После этого вам предложат настроить live TV — мы расскажем об этом на следующей странице — поэтому сейчас нажмите Skip. Примите условия [T&Cs], и ваша начальная настройка закончена. Теперь надо настроить и запустить ваш сервер *Emby*, следуя пошаговому руководству (*см. справа*).

Вернитесь во вкладку Server панели управления, чтобы отследить добавление файлов в вашу медиа-библиотеку через раздел Running Tasks. Когда оно завершится — что может произойти через несколько часов, если ваша медиа-библиотека велика — ваши мультимедиа будут готовы к просмотру. Нажмите ссылку Home в верху экрана, чтобы вернуться в главный интерфейс.

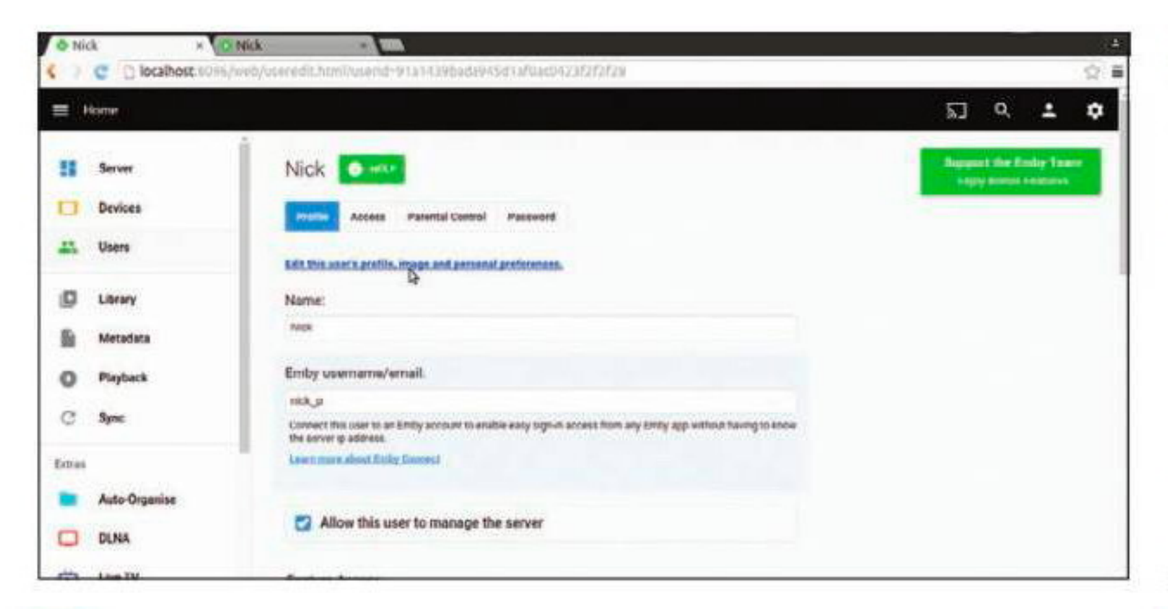
Раздел My Media перечисляет все ваши библиотеки и опции live TV, если вы их настроили (см. стр. 32). Щелкните по разделу, чтобы просмотреть его содержимое. Нажимайте на заголовки наверху страницы, чтобы изменить отображение, включая Collections, Genres и т.д. Если вы хотите осуществлять просмотр по названиям, есть стандартное отображение Show или Movie. Когда вы начнете просмотр программ, *Emby* примется выстраивать серии по порядку (см. в разделе Suggestions). Выберите серию, фильм или трек, и *Emby* предложит вам изображения и исчерпывающую информацию по ним. Это не только отлично выглядит, но и помогает вам понять, что вы сейчас увидите.

Уделите время тщательному ознакомлению с web-интерфейсом *Emby* — вы найдете удобные ярлыки и множество опций: например, возможность редактировать информацию и изображения для каждой программы. Переверните страницу — и вы найдете более подробную информацию о том, что еще можно сделать с вашим медиа-сервером.

› Нужно переименовать много файлов в формат, распознаваемый *Emby*? Совершенно незачем делать это тяжелым и трудоемким ручным способом — тут поможет *Filebot*.

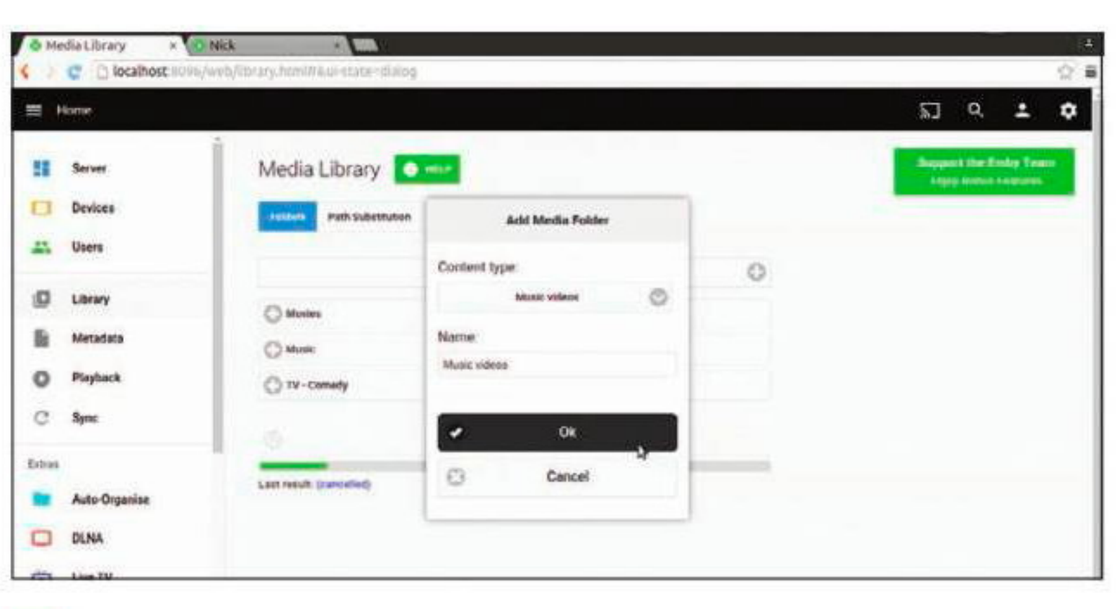
www.linuxformat.ru/subscribe

Управляйте своей медиа-библиотекой с помощью Emby



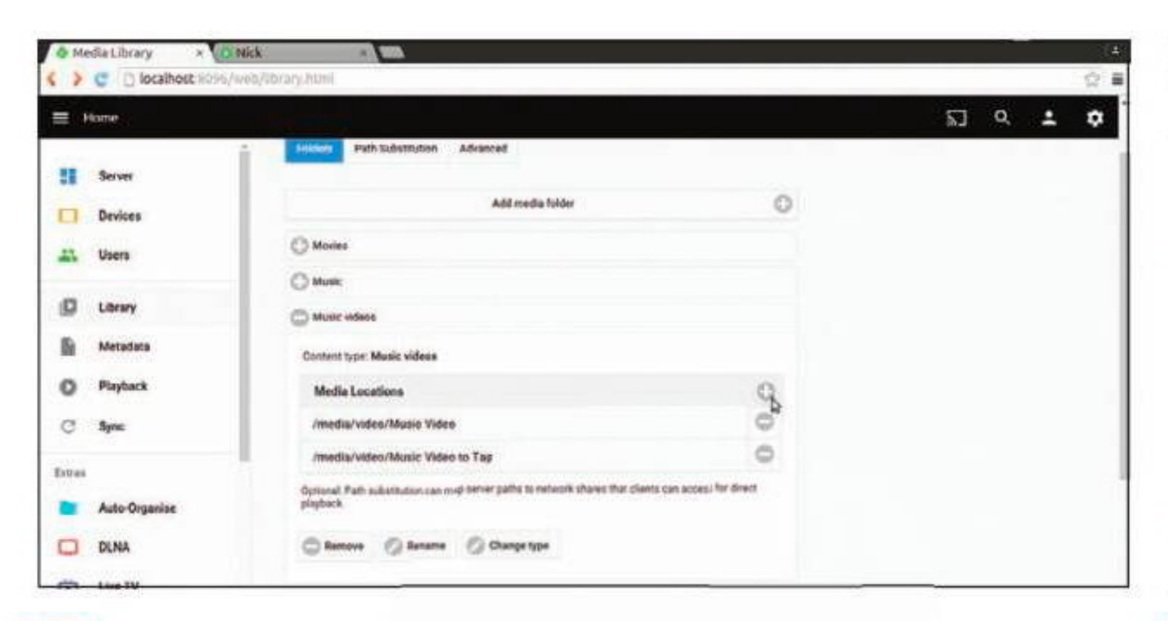
1 Настройки пользователя

Раздел Users панели управления *Emby* — это место, где вы управляете профилями пользователей, включая свой собственный. Если вы планируете делить свой сервер с другими пользователями (более подробно см. на стр. 33), начните с защиты своей учетной записи пользователя паролем здесь: для этого нажмите на свой пользовательский профиль и щелкните по Edit this user's profile — далее последует Profile (при желании можете загрузить свое фото).



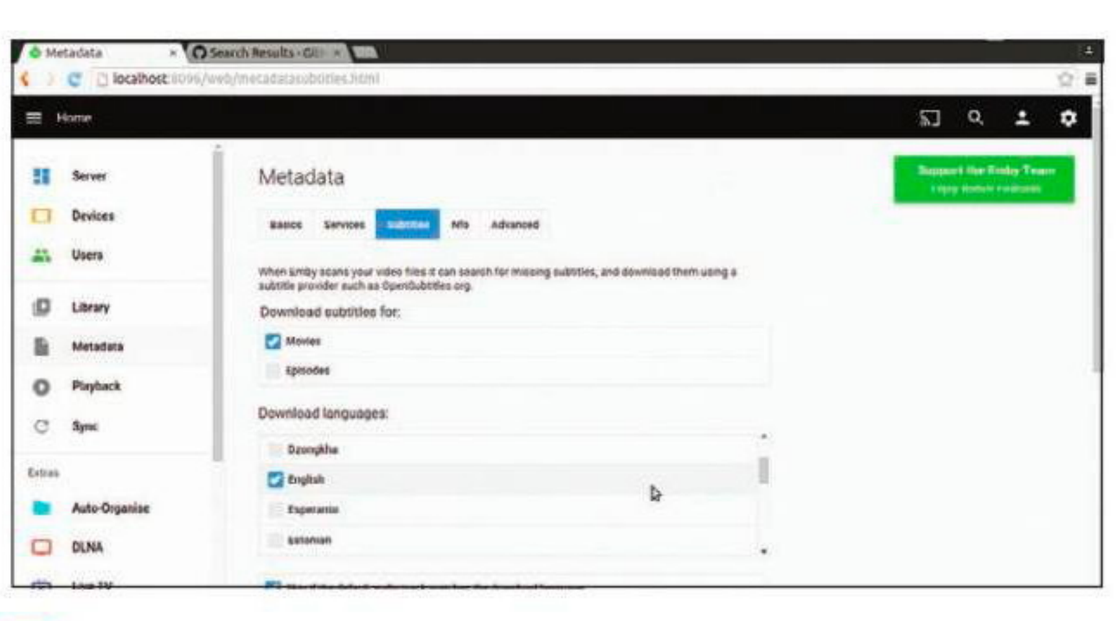
2 Добавьте библиотек

Перейдите в Libraries в панели управления и нажмите на Add media folder, чтобы добавить мультимедиа к вашей коллекции. Процесс похож на тот, с которым вы встречались в мастере настройки: выберите тип и имя, выберите папку (или папки), затем сохраните библиотеку и создайте еще, если нужно. *Emby* начнет сканирование в поисках контента на основе заданных вами критериев — если библиотеки большие, это потребует времени.



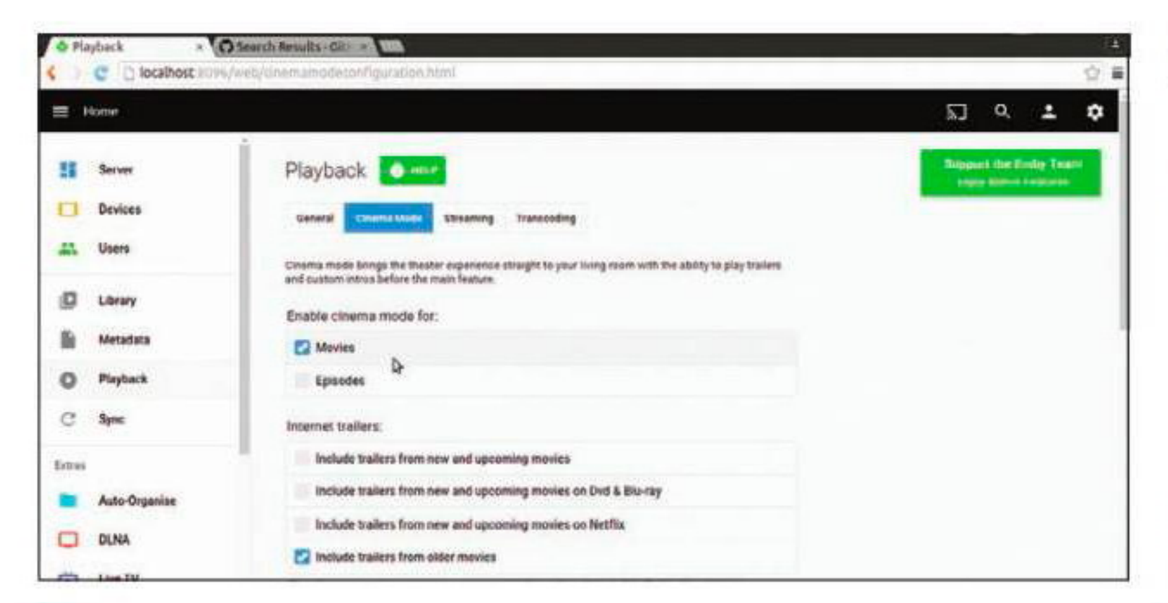
3 Редактируйте библиотеки

Нажмите на значок + рядом с существующей библиотекой, чтобы внести в нее изменения: добавить папки, изменить тип мультимедиа, переименовать или даже полностью удалить. Перейдите во вкладку Path Substitution, если ваши мультимедиа хранятся на автомонтируемых сетевых дисках и вы хотите, чтобы клиент воспроизводил их напрямую, а не заставлять сервер перекодировать их; настройте прямое указание на файл в вашей сети.



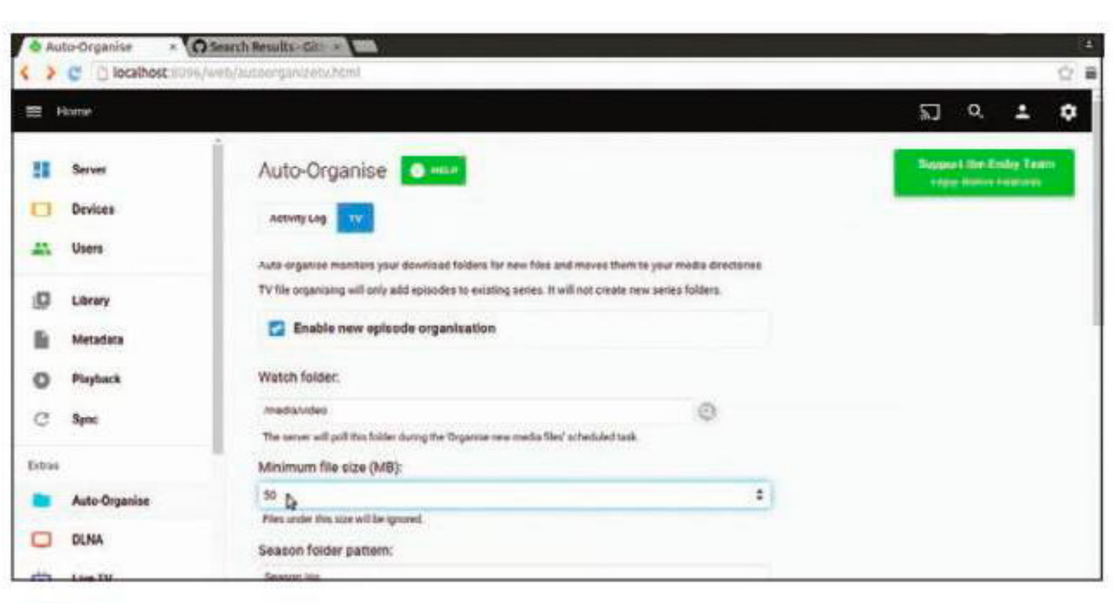
4 Настройте метаданные

Далее выберите Metadata в левой панели. Перейдите во вкладку Services, чтобы посмотреть, какие онлайн-скрейперы использует *Emby*. В большинстве случаев это можно пропустить. Далее перейдите на вкладку Subtitles, если хотите скачать субтитры для фильма и ТВ-сериала (для слабослышащих зрителей или субтитры на иностранных языках). Для скачивания дополнительной информации используйте вкладку Advanced.



5 Настройка воспроизведения

Раздел Playback содержит удобные настройки для оптимизации работы вашего сервера. Cinema Mode позволяет воспроизводить трейлеры для поддерживаемых сериалов и фильмов, а вкладка Streaming — настроить скорость передачи для клиентов, очень удобно для более медленных соединений. И, наконец, выберите Transcoding, если у вас не слишком мощный компьютер и вы не можете позволить себе посвятить все его ресурсы стримингу.



6 Настройте автоматическую организацию

Упростите добавление файлов новых серий в библиотеку, переключившись на Auto-Organise из ее собственного раздела в панели управления. Переключитесь на вкладку TV и отметьте Enable new episode organization. Выберите папку, где будут сохраняться новые серии, и настройте Episode file pattern, чтобы он соответствовал вашему соглашению об именовании. Остальные настройки очевидны и не требуют пояснений. Закончив, нажмите Save.

Сделаем с мультимедиа больше

Итак, основная часть готова; узнайте, что еще может сделать для вас *Emby*.

Хотя одна из основных причин настройки медиа-сервера — возможность доступа к вашей личной медиа-коллекции при первой же необходимости, не стоит останавливаться на достигнутом. В *Emby* имеются полезные инструменты и функции, способные сделать вашу работу с медиа-сервером еще лучше.

Представьте, что вы можете смотреть живые телетрансляции или программы в записи откуда угодно. Если у вас есть совместимый ТВ-тюнер и движок, можно интегрировать их с *Emby*, что позволит смотреть ТВ, а также получать доступ к вашим мультимедиа через ваш медиа-сервер.

Просмотр и запись live TV

В Live TV *Emby* встроена поддержка только одного типа тюнеров, HDHomeRun, и если у вас новый ТВ-тюнер, загляните сначала на www.silicondust.com, чтобы получить более подробную информацию. В порядке альтернативы, *Emby* поддерживает сторонние Live TV-решения через плагины, и если ваше ТВ уже настроено — на вашем основном медиа-сервере или еще где-то в сети — вы, вероятно, сумеете их интегрировать.

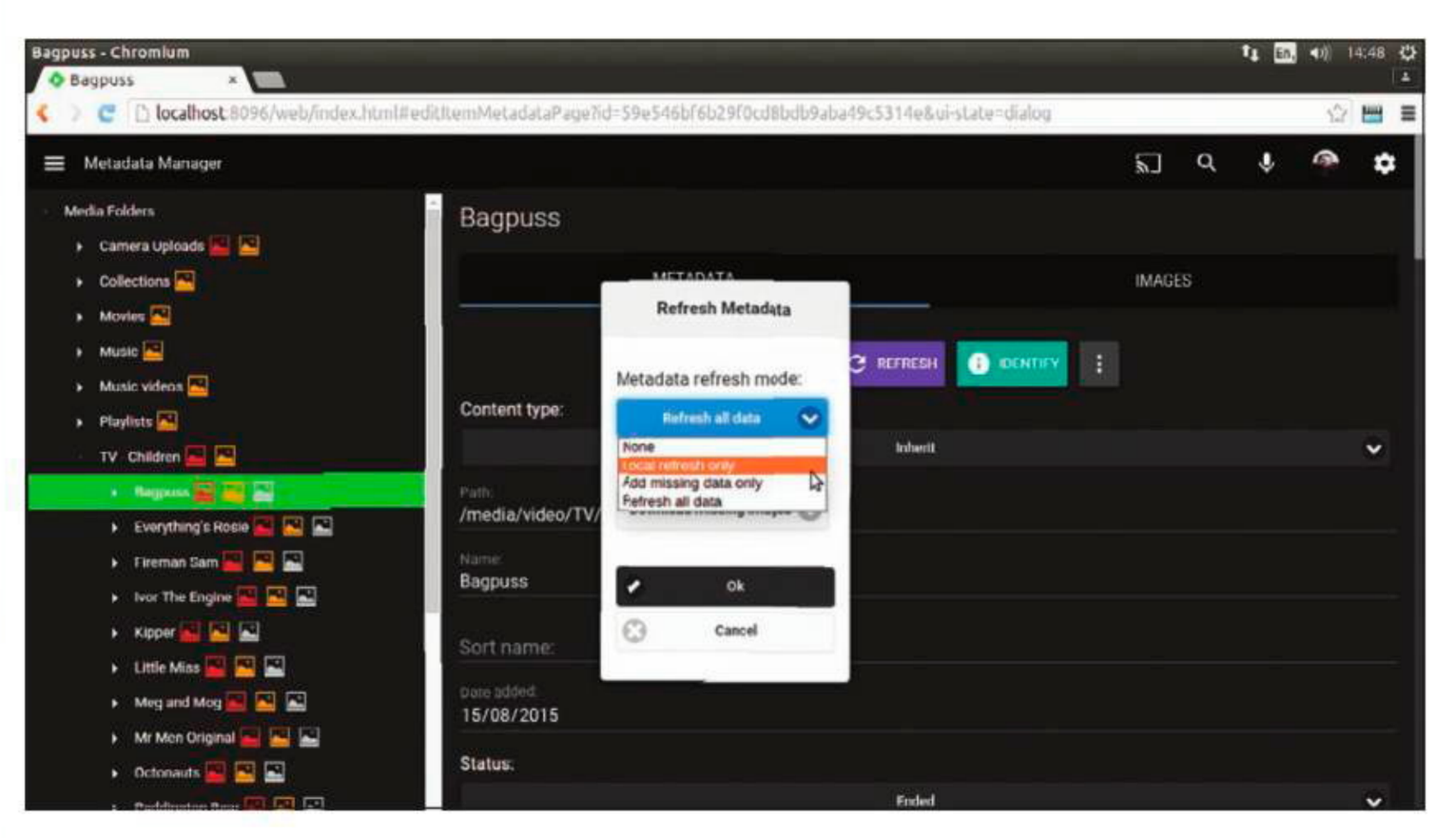
Лучше всего поддерживаются плагины для решений Windows, но опция Linux реализована в форме *TVheadend*, и ожидается прибавление — например, в не очень отдаленном будущем стоит рассчитывать на появление *MythTV*. На момент написания данной статьи *Emby* поддерживает более старый релиз *TVheadend* (3.4), но перед первой настройкой вашего сервера *TVheadend* стоит проверить, не появилась ли поддержка более новой ветви (4.x).

Подробные инструкции по добавлению *TVheadend* к вашему медиа-серверу вы найдете в разделе Download на http://tvheadend.org. Сначала проверьте, какую версию *TVheadend* (4.x или 3.x) поддерживает плагин, и добавьте соответствующий стабильный репозиторий. После этого установите *TVheadend*, введя `sudo apt-get install tvheadend` и нажав Enter.

Вам предложат ввести имя пользователя и пароль — подойдут любые. По окончании установки откройте браузер и перейдите на localhost:9981. Войдя, перейдите на вкладку Configuration и выберите 'DVB Inputs'. Там будет присутствовать ваше TV-устройство USB, поэтому выберите его и нажмите Add DVB Network by location... Выберите свою страну и трансмиттер (в Великобритании его местоположение можно найти на карте https://ukfree.tv/maps/freeview) и нажмите Add DVB network. Теперь уберите галочку с Autodetect muxes и отметьте Enabled в Adapter configuration. Нажмите Save, и начнется начальное сканирование — за ним вы можете следить в Information and capabilities или перейти на вкладку Services, чтобы посмотреть, какие найдены каналы. После завершения нажмите Map DVB services to channels....

Вы можете отслеживать процесс из вкладки Services, в столбце Channel Name. По завершении процесса вы технически должны получить доступ. Простейший способ проверить, работает ли ваш тюнер — установить *Kodi* (см. стр. 34), а затем — плагин TVheadend PVR, и посмотреть, сможете ли вы смотреть ТВ через них.

Проверив это, следуйте пошаговому руководству, чтобы настроить нужный плагин *Emby*. Ищите зеленую отметку рядом с External Services, которая должна показывать, что плагин настроен правильно. Откройте скользящее меню и выберите Live TV. Удостоверьтесь, что у вас есть доступ с вашего тюнера — оглядите разные разделы. В случае неполадок загляните в раздел Live TV форумов *Emby* (http://emby.media/community) за подсказкой.



› Поправьте нехватку или ошибки метаданных всесторонним инструментом *Emby* Metadata Manager.

Управление метаданными

У *Emby* — в большинстве случаев — не должно быть проблем с определением вашей музыки, сериалов или фильмов. Периодически бывают сбои, или другие мультимедиа — например, домашнее видео, фото или даже клипы — могут не распознаваться *Emby*, или *Emby* не сможет их обработать.

Доступ не из дома

Вам не обязательно иметь соединение с локальной сетью для доступа к размещенным на *Emby* мультимедиа — *Emby* можно настроить на стриминг через Интернет, чтобы вы могли наслаждаться своими мультимедиа в любом месте, где есть доступ к Интернету. Очень неплохо иметь быстрое соединение дома, в идеале — оптоволокно со скоростью загрузки не менее 5 Мб/с; и не забудьте, что стриминг очень быстро съест любую ограниченную полосу пропускания.

Если на вашем роутере включен UPnP, то *Emby* должен работать в Интернете без дополнительной настройки; в ином случае придется вручную настроить перенаправление портов и убедиться, что вы присвоили статический IP-адрес своему серверному ПК. Настройки, на которые вам надо взглянуть — и подкрутить их — находятся в Advanced > Hosting в разделе Manage Server Emby.

Есть два способа удаленного доступа к вашему серверу. Во-первых, вручную, вводом публичного IP-адреса вашего сервера (того, который выдан вашим ИП). Однако он может меняться, и если ваш динамический DNS хост настроен на что-то вроде no-ip.com, то введите это в поле External WAN Address в Advanced > Hosting: это позволит вам удаленно настраивать свой сервер через hostname.ddns.net:8096.

Второй метод проще: зарегистрируйте бесплатную учетную запись Emby Connect на http://app.emby.media, затем просто зайдите в эту учетную запись через приложение своего устройства — и можете получать доступ к своим мультимедиа откуда угодно.

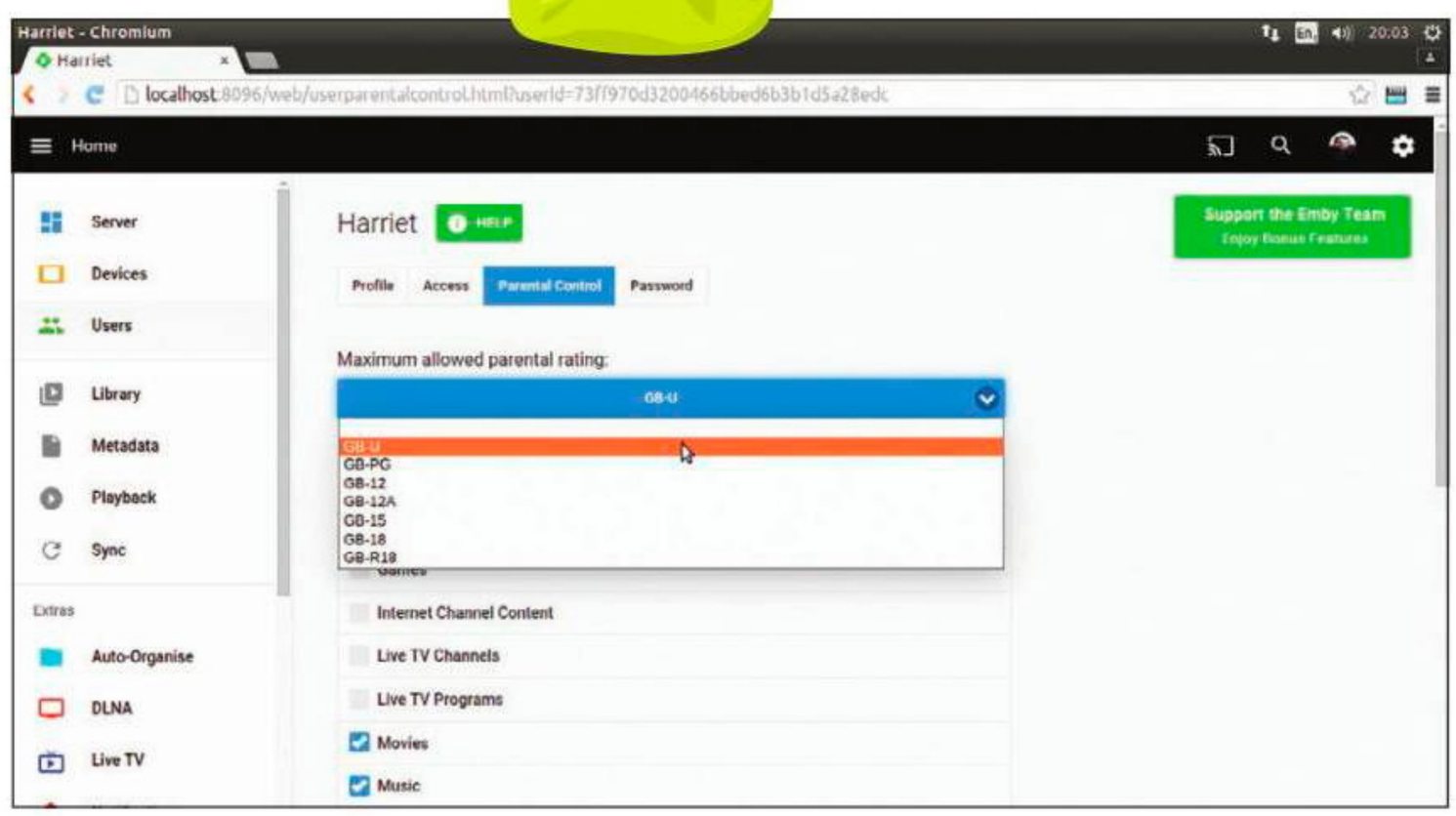
Еще лучше будет зайти на http://app.emby.media через браузер, тогда вы также сможете удаленно администрировать свой сервер.

www.linuxformat.ru/subscribe

Вот здесь-то и вступает в дело Metadata Manager — войдите в него из раздела Admin выпадающего меню *Emby*.

Metadata Manager позволяет посматривать свои мультимедиа по папкам — расширьте его, чтобы он показывал такую важную информацию, как ТВ-сериал, фильм или музыкант. Щелкните там, чтобы отредактировать эту информацию. Можно ввести данные вручную или нажать Identify, чтобы приложение поискало совпадения в соответствующей базе данных. Нажмите Save, чтобы включить то, что вы выбрали, или Refresh, чтобы попытаться скачать недостающие данные — опция Advanced позволит вам контролировать загрузку метаданных и изображений. Вы также увидите раздел Images, где можно управлять имеющимися изображениями или загружать собственные.

Можно получить тот же уровень управления музыкальными альбомами и треками, а также отдельными сериями сериалов — просто нажмите > рядом с названием, чтобы перейти на следующий уровень. А если метаданные недоступны для скачивания из Интернета — обычно такое случается с музыкальными клипами — *Emby* все равно старается помочь, предлагая вам заполнить соответствующие поля, в зависимости от типа мультимедиа, который вы выбрали для этого раздела.



Хотите допустить к своему серверу младших в семье? Настройте многопользовательский доступ.

Многопользовательский доступ

Одно из огромных преимуществ *Emby* — возможность предоставить каждому собственный индивидуально настраиваемый уровень доступа к серверу. Это делается в разделе Users в Manage Server. Нажмите + рядом с пользователями и сначала введите свое имя пользователя. Снимите галочку с Enable access to all libraries, если хотите ограничить доступ к своим мультимедиа, и нажмите Save.

Настроив пользователя, воспользуйтесь вкладкой Profile, чтобы при необходимости соединить его с учетной записью Emby Connected, а также ограничить ему доступ и управление разными частями вашего сервера. Вкладка Access позволяет ограничить доступ не только к библиотекам, но и к определенным устройствам. Parental Control позволяет установить ограничение по возрасту для определенного пользователя и заблокировать непроверенный контент по типу (и тэгам). Нажмите Access Schedule, чтобы ограничить доступ по дням и часам. И, наконец, Password позволит вашему подопечному защитить свою учетную запись паролем.

Когда всё будет настроено, пользователи смогут получить доступ к *Emby* через сеть или с поддерживаемого устройства и настроить индивидуальное отображение контента вашего сервера. Щелкнув по своему фото пользователя и выбрав Settings, они смогут настроить персональные предпочтения, например, заставку на экране home, настройки воспроизведения и т.д.

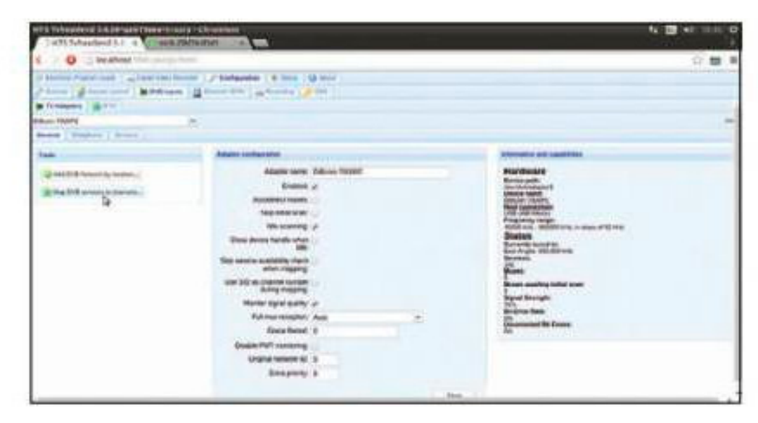
Это отличный способ дать вашим домашним личный доступ к вашему серверу. Вы даже сможете пойти дальше и настроить специальный гостевой доступ, отправив по электронной почте приглашения другим пользователям с учетными записями Emby Connect, чем позволите им обращаться к одной или нескольким своим библиотекам.

Доступ из консоли

Если вы планируете задвинуть свой сервер куда-нибудь в уголок, есть смысл настроить на нем консольный доступ с помощью SSH. Это обеспечит вам безопасный доступ и контроль через командную строку с другого ПК или устройства в вашей сети с помощью соответствующего приложения или программы (например, *Putty* на Windows или *Serverauditor* на iPad). Рекомендуемый сервер — OpenSSH (www.openssh.com), однако пользователям Ubuntu стоит заглянуть на https://help.ubuntu.com/community/SSH, чтобы изучить руководство по его настройке и установке.

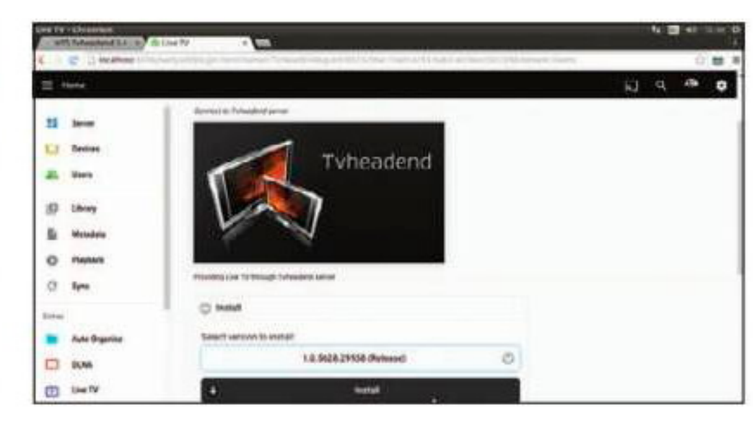
Если вы предпочитаете доступ на рабочий стол, то настройте этот доступ через VNC — установите сервер VNC на свой сервер (sudo apt-get install x11vnc) и заходите на него из любого подходящего клиента VNC (например, *Remmina* включена в Ubuntu 14.04.3).

Настройте поддержку live TV в Emby



1 Настройте ТВ-сервер

Если у вас нет ТВ-тюнера HDHomeRun, придется сначала установить поддерживаемый сторонний ТВ-пакет. На момент написания данной статьи единственным из основных поддерживаемых серверов Linux является TVheadend.org, поэтому, руководствуясь основным текстом по установке, настройте его с *Emby*. Чтобы проверить, работает ли он, установите *Kodi* и активируйте плагин *TVheadend* PVR.



2 Добавьте плагин

Откройте браузер и перейдите на localhost:8096 для доступа к интерфейсу *Emby*. В выпадающем меню нажмите Manage Server. Выберите Live TV в левой панели, перейдите на вкладку External Services, щелкните по плагину, который хотите добавить (здесь это *Tvheadend*), затем по Install. Перезапустите сервер, когда вам предложат это сделать — возможно, придется вручную вернуться на вкладку External Services.

3 Настройте плагин

Плагин появится в Installed Services. Щелкните по трем вертикально расположенным точкам рядом с его именем, выберите Settings и введите нужную информацию — возможно, придется сделать это через ваш интерфейс ТВ. С *TVheadend* вам понадобятся имя пользователя и пароль, которые вы задали. Сохранитесь и переключитесь снова на вкладку Tuners. Вы увидите в списке свой тюнер и плагин.

Просмотр с других устройств

Настроив медиа-сервер, наслаждайтесь своими мультимедиа на многих устройствах.

Когда *Emby* будет установлен и настроен, как мы показали, вы сможете получать доступ к своим мультимедиа через браузер на любом компьютере: введите в браузере 92.168.x.y:8096 (заменив 192.168.x.y на IP-адрес своего сервера), и затем зайдите для удаленного доступа к своему серверу.

Правда, web-интерфейс предназначен скорее для управления, чем для просмотра ваших мультимедиа; ну, а какие еще у вас есть опции и как вы сможете получить доступ к своим мультимедиа с других устройств?

ПК и ноутбуки

Пользователи Windows могут установить *Emby Theater* (http://bit.ly/EmbyTheater) — как и предполагает его название, это интерфейс, оптимизированный для серверов *Emby*, и наилучший способ работы с *Emby* в Windows.

Альтернативой на других платформах — включая Mac — является установка *Kodi* и дополнения *Emby for Kodi* (см. внизу). Вы даже можете настроить специальный медиа-центр, объединив дистрибутив OpenElec (http://openelec.tv) с *Emby for Kodi*. *Emby* также полностью поддерживает протокол DLNA, так что любой медиа-плейер или дистрибутив, совместимый с DLNA, сможет получить доступ к вашей библиотеке, хотя и без интересных постеров и дополнительных метаданных.

Raspberry Pi

Любая модель Raspberry Pi превращается в отличный медиа-плейер в качестве приставки к ТВ, хотя рекомендуется более новый и более мощный Pi 2. Наилучшие результаты получаются, если соединить его с OSMC (http://osmc.tv), дистрибутивом с *Kodi*, который также поддерживает дополнение *Kodi for Emby*. Загляните на http://osmc.tv/download/linux и скачайте инструмент установки устройств, который создаст загружаемый носитель для установки на ваш Pi — для этого вам понадобится 8-ГБ SD-карта класса 10 и считывающее устройство для SD-карт на вашем компьютере.

Телефоны и планшеты Android

Для Android вы обнаружите большой выбор UPnP-совместимых медиа-плейеров, однако очевидным выбором будет официальное приложение *Emby*. Оно отлажено и работает с MB Connect, что выдаст вам доступ к вашему медиа-серверу, даже если вы не дома. Официальное приложение бесплатно для Emby Supporters (см. http://emby.media/donate), а остальным обойдется в £3,53, и перед покупкой его можно протестировать.

В порядке альтернативы, если на ваш телефон или планшет уже хватает трат, вы можете получить эту функцию и даром, бесплатно установив полное приложение *Kodi* из магазина Google Play. Сделав это, следуйте пошаговому руководству внизу по добавлению требуемого репозитория к *Kodi* на вашем устройстве Android, что позволит вам добавить дополнение *Emby*.

Еще одна возможность взять свой носитель с собой — приложение *VidOn Player HD*: оно бесплатное, воспроизводит широкий ряд мультимедиа и даже позволяет скачивать контент с вашего сервера *Emby* на мобильное устройство для просмотра оффлайн, когда вы не дома. Недостатки? Текущая версия не работает с Android Lollipop 5.1 и новее.

Windows Phone

Emby не только предлагает специальное приложение для пользователей Windows Phone (поищите 'Emby' в своем Store): это единственное телефонное приложение, которое на данный момент бесплатно для всех пользователей. Есть также приложение Windows 8.1/10 Store для планшетов, но оно стоит £3,89.

Apple iPhone и iPad

На момент написания приложение *Emby* для iOS «почти завершено» — когда оно выйдет, готовьтесь выложить за него около £3–£4. А вот неофициальное приложение *MediaBrowser* предлагает столь же приятную работу по цене £2,99. Если вы не хотите платить за доступ, можете соединиться с помощью браузера *Safari* через localhost:8096 дома, или вне дома использовать https://app.emby.media. Или — если вас устраивает не столь гладкий интерфейс (но все же подробный) — установите вместо этого бесплатное приложение *VLC for iOS*. Откройте основное меню и выберите Local Network — после небольшой паузы, под Universal Plug 'n' Play должен в виде опции появиться *Emby*, позволив просматривать и получать доступ к вашим мультимедиа.

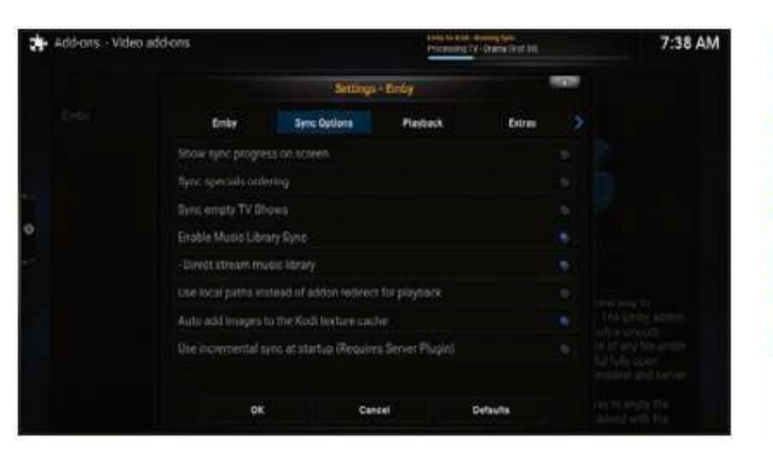
Apple TV

Раньше вам требовалось разблокированное Apple TV для работы с медиа-плейером, способным воспроизводить потоки UPnP, подобно *Emby*. Однако новое Apple TV 4 идет с магазином приложений, и,

Доступ к вашему серверу Emby через Kodi

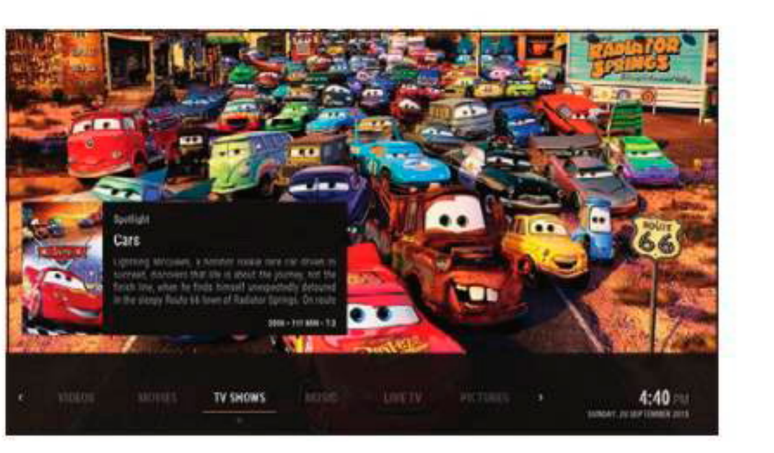
1 Установите дополнение Emby

В браузере перейдите на https://emby.media/downloads/emby-for-kodi и щелкните по ссылке repository, чтобы сохранить ZIP-файл на свой диск. Теперь откройте *Kodi*, перейдите в System > Settings > Add-ons > Install from zip file для определения репозитория *Emby*. Выберите свой и перейдите по Install from repository > Kodi Emby AddOns > Video add-ons > Emby для установки дополнения.



2 Настройте дополнение

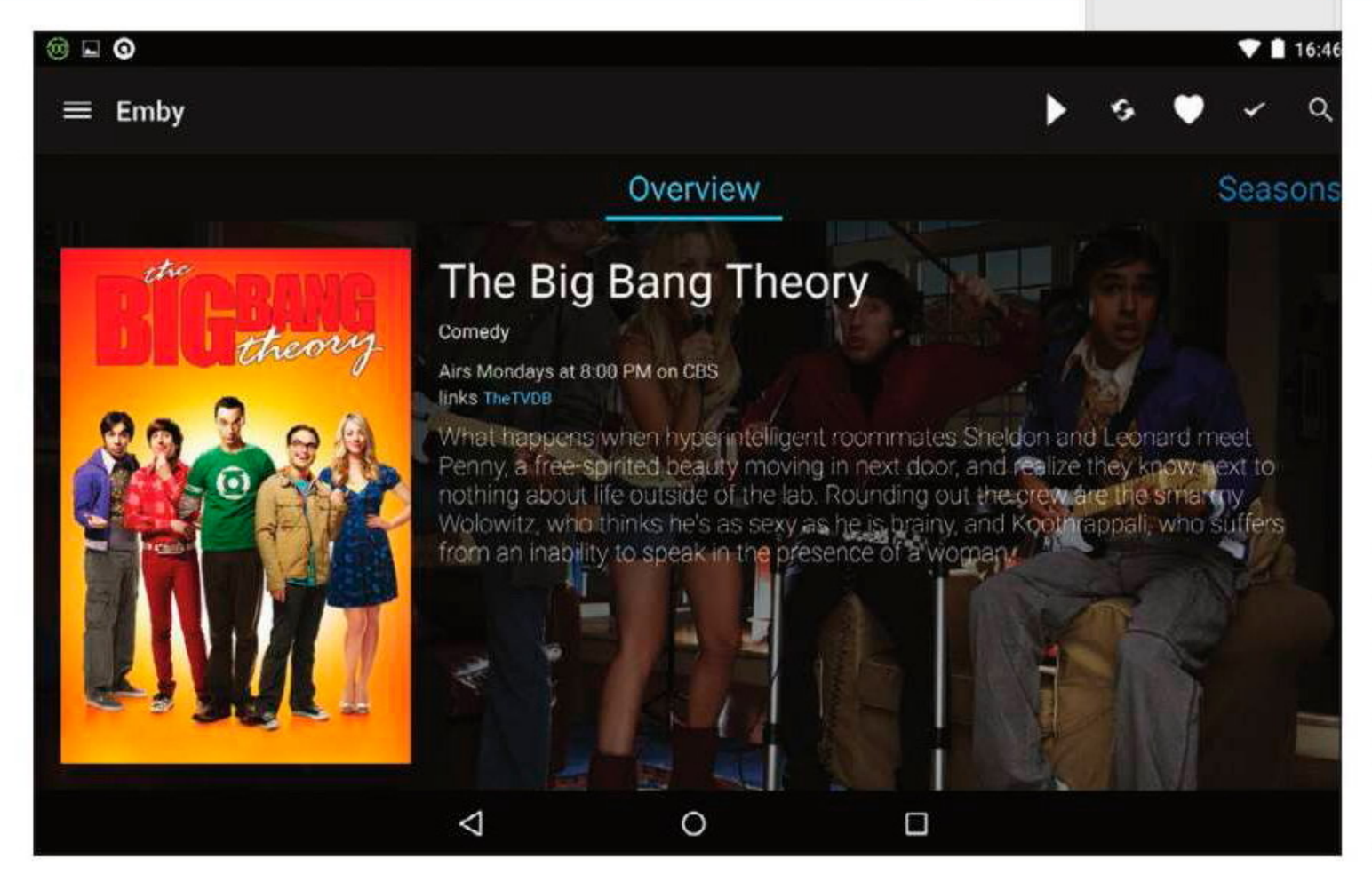
Нажмите Install. После этого *Emby* постарается определить ваш сервер — если хотите, нажмите Yes и выберите своего пользователя. Выберите, включать или не включать музыку, и нажмите Yes, чтобы включить прямой стриминг для доступа вне дома. Введите пароль. После этого щелкните по дополнению и выберите Configure для более тонкой настройки — неплохо начать с опций синхронизации, Sync Options.



3 Добавьте скин

Пока дополнение синхронизирует вашу библиотеку, нажмите OK, затем кнопку Home. Просмотрите свои библиотеки *Emby* через меню Video или Music Add-ons — нажмите Emby, чтобы просматривать и получать доступ к вашим мультимедиа. Вам не нравится внешний вид? Вернитесь в Add-ons > Install from repository > Kodi Emby Addons > Skin и попробуйте скин Arctic Zephyr: Exploded.

www.linuxformat.ru/subscribe



› **Лучше всего ваше устройство будет работать, если оно поддерживает официальное приложение *Emby*.**

вероятно, когда-нибудь *Emby* будет разработан для него. А пока что вы можете воспроизводить потоковый контент на своем Apple TV через AirPlay, но для этого потребуется iPad или iPhone в качестве посредника. Если вы используете Android, можете направить поток на Apple TV: *VidOn Player HD* поддерживает AirPlay через Settings > AirPlay. Чтобы добиться наилучших результатов, выберите Compatibility Mode.

Вещание через AirPlay

Кстати, хотите передавать музыку напрямую с сервера на колонки AirPlay, не привлекая свой мобильник? Загляните на http://bit.ly/Raop2, там есть руководство по установке и настройке *Pulseaudioraop2*, экспериментального ответвления *PulseAudio*, которое поддерживает более новые устройства AirPlay. Процесс будет длительным, но результат стоит ваших трудов — хотя рекомендуем предварительно сделать моментальный снимок вашей системы LVM, на случай, если вдруг что-то пойдет не так, как планировалось.

Smart TV / плейеры Blu-ray

Большинство Smart TV и плейеров Blu-ray предлагают возможности просмотра потоковых медиа с сервера DLNA/UPnP, как *Emby* — проблемой может стать поддержка формата, но если у вашего сервера достаточно мощности для перекодирования, то вряд ли это непреодолимо. Было бы неплохо перекодировать ваши мультимедиа с помощью кодека x264/H.264, инкапсулированного в файл MP4 с MP3-аудио, тогда всё должно получаться замечательно.

Если у вас Samsung TV с хабом серии Smart Hub D (2011 или выше), то вам будет доступно созданное сообществом приложение *Emby*: перейдите на https://emby.media/download, выберите Samsung Smart TV и нажмите на ссылку на инструкции по скачиванию и установке этого приложения.

Приставки

Существует специальное приложение *Emby* для приставок Roku, и если вы приобрели NowTV за £10 или £15, можете загрузить приложение через Developer Mode. (См. пошаговое руководство внизу, с объяснением, как это сделать).

Недавно вышли также специальные приложения *Emby* для устройств Android TV и Amazon Fire TV — вы можете скачать бесплатные пробные версии через соответствующий магазин, а полное приложение будет стоить £3,95 в качестве разовой покупки или бесплатно для Emby Supporters.

Google Chromecast

В Chromecast поддерживается видео только WebM и MP4/H264, поэтому, вероятно, вашему серверу придется как следует попотеть над перекодированием. Тем не менее, поддержка Chromecast встроена в приложения iOS и Android; а если вы установите расширение Google Cast в *Chrome* или *Chromium* на своем сервере (или ином ПК), то впоследствии сможете перенести вкладку, содержащую ваши мультимедиа, на свой Chromecast для воспроизведения на большом экране.

Игровые консоли

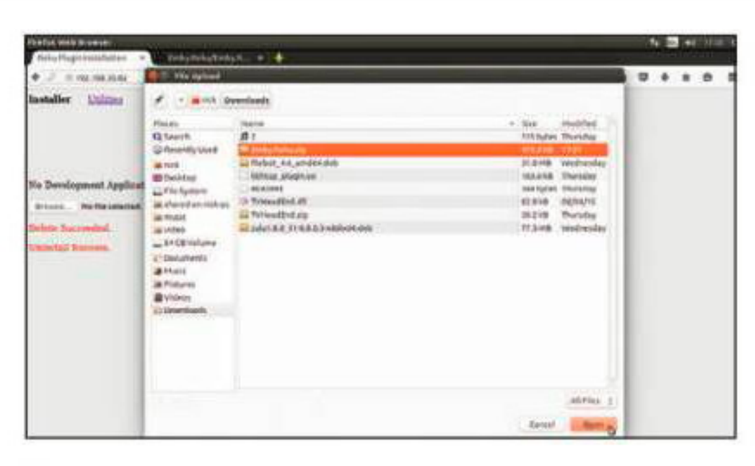
Есть специальный плагин для Xbox 360, добавляемый к *Windows Media Center* через https://emby.media/download. С Xbox One все сложнее, но грядущий релиз панели управления на Windows 10 должен скорее рано, чем поздно, обзавестись приложением *Emby* — загляните на форумы *Emby*, чтобы узнать о самых свежих обновлениях.

Playstation 3 может получить доступ к *Emby* с помощью своего сервера DLNA (убедитесь, что вы включили DLNA на PS3 через Settings > Network Settings > Media Center), а пользователи PS4 должны проверить, установлена ли у них самая последняя прошивка. LXF

Доступ к Emby через NowTV

1 Доступ к Developer mode

На пульте удаленного управления NowTV трижды нажмите на кнопку Home, потом два раза вверх, затем направо, налево, направо, налево и, наконец, опять направо. Вы увидите этот экран. Выберите Enable installer and restart, прочтите Условия, нажмите I agree, затем снова выберите Enable installer and restart. После перезагрузки выберите пароль для защиты вашего устройства от несанкционированного доступа.



2 Скачивание и установка

Откройте окно браузера и введите IP-адрес вашего NowTV. Введите 'rokudev' в качестве имени пользователя и свой пароль. Теперь откройте отдельную вкладку, перейдите на http://bit.ly/emby-nowtv, щелкните по ссылке Emby.Roku.zip > 'Raw' и сохраните закачку в своей папке **Downloads**. Снова перейдите на исходную вкладку, нажмите Browse, выберите ZIP-файл и щелкните по Install.

3 Наслаждайтесь мультимедиа

Закончив, нажмите на Next на своем Now TV. Запишите PIN, вернитесь в свой браузер, перейдите на https://emby.media/pin и введите PIN, свое имя пользователя и пароль Emby Connect, затем нажмите Submit. Как только PIN будет подтвержден, вы сможете просматривать и получать доступ к своим мультимедиа, размещенным на *Emby*, прямо со своего Now TV — и всё это бесплатно!

Интернет-архивариус

Джонни Бидвелл берет интервью, по сути, у цифрового библиотекаря — Алексис Росси, директора Internet Archive.

Алексис Росси [Alexis Rossi] является директором Internet Archive, прозванного «Александрийская библиотека 2.0» — правда, сама она в шутку именует его крупнейшим сайтом, о котором никто никогда не слышал. Мы встретились с ней, чтобы поговорить о больших данных, API, картонных фигурках, а также признаться, что сами используем архив для хранения образов DVD.

Linux Format: **Расскажите, что привело вас в Internet Archive?**

Алексис Росси: В колледже я хотела стать книжным редактором, правила кулинарные книги, брошюры и тому подобное. Работала в научно-популярном издательстве. А в 1994-м, можно сказать, влюбилась в Интернет — в саму возможность найти людей, которым интересно то же, что и мне (ведь родилась я в маленьком городке) — возможность общаться с друзьями на расстоянии в добрую половину глобуса, причем бесплатно. Так я и решила от публикации бумажных книг перейти в Интернет-публикации, и начала работать, пожалуй что, в первом официальном агрегаторе новостей, ClariNet. Сейчас его уже не существует, но появился он в 1989 году, и когда я пришла туда в 1994-м, они всё еще публиковали материалы в телеконференциях, но вскоре перешли в Интернет. Так что, похоже, в технологию меня затянуло потому, что я люблю общаться и делиться информацией.

О РОСТЕ INTERNET ARCHIVE

Мы выросли, теперь в Wayback Machine около 425 миллиардов объектов.

LXF: **Чем вы занимаетесь в Internet Archive?**

AP: Я директор, и отвечаю за все цифровые материалы, будь то данные из Wayback Machine, фильмы или что угодно. Я также отвечаю за проекты, связанные с доступом. Совсем недавно мы переделали сайт и по-прежнему кое-что совершенствуем, но все самое основное было представлено в мае. Работа велась года полтора. Кроме того, я много выступаю, рассказывая, в частности, и об архиве. К работе с людьми периодически обращаюсь с 2000 года.

LXF: **Расскажите немного об Internet Archive. С какого времени в нем хранятся данные?**

AP: Конечно. IA существует с 1996 года, тогда же мы и начали архивировать Интернет. Мы являемся некоммерческой цифровой библиотекой, зарегистрированной в штате Калифорния. Наш первый сервис, Wayback Machine, появился в 2001-м. С 2002 года мы стали размещать также музыку и видео. С того времени, мы значительно расширились, и сейчас в Wayback Machine 425 миллиардов объектов.

LXF: **Ого!**

AP: Именно что «ого»: у нас порядка 8 миллионов текстов, около 2 миллионов фильмов, 2,5 миллиона аудиотреков и записей круглосуточного вещания 60 телеканалов. Мы известны в том числе и как хранилище ПО.

LXF: **Ха-ха. У нашего журнала там тоже копилочка.**

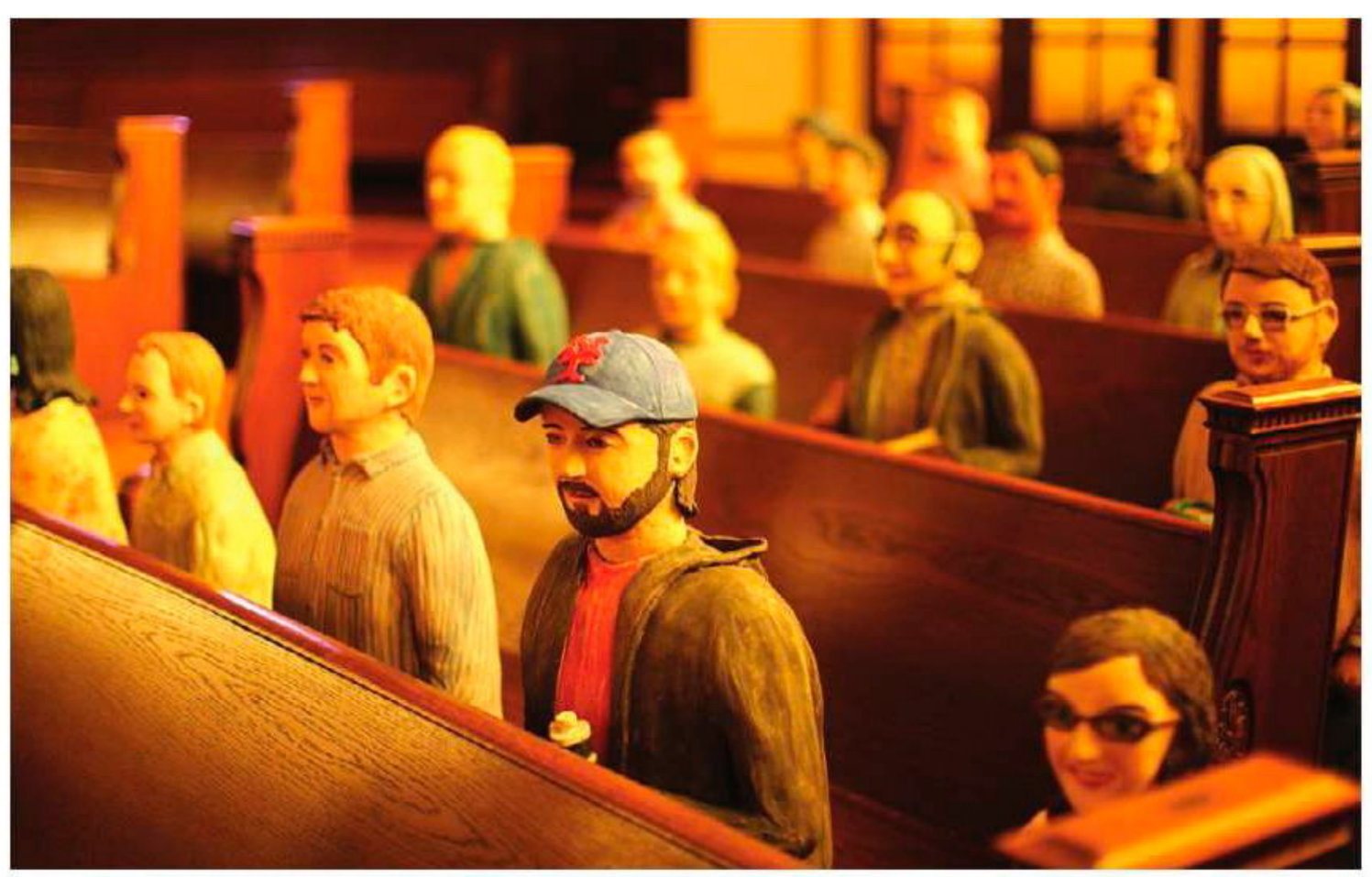
AP: Правда? И какая же?

LXF: **Раньше торренты с наших дисков на обложке мы размещали на собственном сервере, но вскоре места на диске стало не хватать. Куда проще оказалось выложить эти гигабайты данных на archive.org — ведь там поддерживается даже раздача торрент-файлов, так что мы ссылаемся на них из нашего архива, и пользователи ничего не замечают. Кто-то даже создал для нас отдельную коллекцию, за что мы очень благодарны.**

AP: Здорово, и вы пользуетесь нашим API типа S3?

LXF: **Нет, просто нажимаю удобную кнопку загрузки. Расскажите об этом API!**

AP: На http://archive.org/help доступна куча API для самых разных задач. Одним из них является интерфейс загрузки, похожий на S3 у Amazon, так что любой, кому доводилось им пользоваться, получает такую возможность прямо из коробки. Он действительно хорош для загрузки больших объемов данных. С его помощью можно и выкачивать информацию, но предназначен он все же для обратного действия. Если у вас много объектов, мы можем создать вам отдельную коллекцию, чтобы у вас было свое место в архиве. И закачивайте туда сколько угодно. Мы надеемся, что где-то через год вы сможете сами создавать коллекции, но пока это делаем мы. Мы предоставляем подробные инструкции по добавлению метаданных — без них добавлять информацию в архив бессмысленно, иначе никто не сможет ничего найти. »



› **Сотрудники, долго прослужившие в Archive, вознаграждаются керамической статуей высотой в 3 фута, работы скульптура Нуалы Крид [Nuala Creed]. Фото: Джейсон Скотт (Jason Scott, CC BY 2.0)**

LXF: Какие еще интересные способы применения есть у новых интерфейсов Archive?

AP: Мой любимый называется RECAP, он предоставляет доступ к документам PACER (PACER — это система с платной подпиской, дающая доступ к записям федерального суда в США). Программное обеспечение RECAP — совместная разработка Принстона и Гарварда. Авторы RECAP создали для пользователей, которые платят за доступ к PACER, плагин для браузера, позволяющий автоматически загружать записи на Internet Archive, где они будут доступны бесплатно всем и каждому. Раскрывая, таким образом, содержание правительственных документов, которые обязательно и должны быть доступны. Таким образом мы получили более миллиона судебных записей, и все это через API, ничего не надо делать вручную. Самый значительный вклад в данный проект принадлежит покойному Аарону Шварцу [Aaron Schwartz], он опубликовал ряд документов PACER в 2008 году. Этот корпус стал основой базы данных RECAP, просмотреть которую вы можете на http://archive.recapthelaw.com.

LXF: К слову о законодательстве: с недавнего времени в Европе действует так называемое «право на забвение», согласно которому Google должен удалять (или по крайней мере подумать об удалении) из результатов поиска ссылки, основанные на пользовательских запросах.

AP: Мы всегда удаляли из Internet Archive данные по разным причинам — в первую очередь, когда кто-то заявлял о нарушении авторских прав, заполняя форму DMCA [Digital Millennium Copyright Act — Закон об авторском праве в цифровую эпоху, — *прим. пер.*] Периодически мы изымаем данные из Wayback Machine и по другим причинам, когда кто-то пытается защитить свой домашний адрес или другую личную информацию.

LXF: А как насчет тех, кто просто желает подправить свое цифровое прошлое?

AP: Как правило, мы неохотно что-то удаляем; но с другой стороны, имея 420 миллиардов материалов, удалить пару из них — не проблема. Нужно всегда проверять, насколько это затрагивает чью-то личную жизнь.

LXF: Я где-то читал, что у вас хранится 12 петабайт данных, это так?

AP: О, сейчас уже гораздо больше. Уникальных данных около 24 петабайт, но по факту вдвое больше, поскольку мы все дублируем, и помимо всего есть еще объем памяти, который используют движок поиска, сайт и прочее. Думаю, всего на данный момент выходит около 55 петабайт крутящихся дисков.

LXF: И снова, ничего себе! С какой же скоростью растет этот цифровой монстр?

AP: Я бы сказала, что за год добавляется около 5 петабайт уникальных данных.

LXF: Если вы некоммерческая организация, то кто финансирует это громадное хранилище?

AP: Примерно 40% дохода приносит сканирование книг — библиотеки нанимают нас для сканирования бумажных изданий, а мы превращаем их в электронные книги. Еще 20% составляет архивирование Интернета. У нас есть доступный по подписке сервис под названием http://archive-it.org, для архивирования, позволяющий не-технарям сообщить нам, какие данные им хотелось бы у нас сохранять и как часто, и мы собираем для них эти материалы. Они могут их скачивать, в конечном итоге получая коллекцию сайтов с возможностью поиска по ним. Зачастую это библиотеки, архивы, музеи, государственные учреждения; например, они могут запросить архивацию всех правительственных сайтов Южной Каролины.

Мы также собираем коллекции доменного уровня — национальные библиотеки, такие как Австралийская национальная библиотека или Библиотека Конгресса, или кто-нибудь наймет нас для архивации всех библиотек Новой Зеландии и скажет: «Предоставьте мне всё, что есть, по Новой Зеландии», мы делаем такую подборку, передаем ее клиенту, а копию размещаем в Wayback Machine, и то же самое с Archive It.

Наша миссия в том, чтобы обеспечить всеобщий доступ ко всем знаниям, и всё, что мы делаем за деньги, будь то оцифровка книг или архивирование Интернета — в конечном итоге, всё это на благо общественности. Так что если кто-то платит нам, чтобы узнать всё о Новой Зеландии, то все эти данные в финале попадают в Wayback Machine. Еще 40% нашего финансирования поступают от фондов и пожертвований.

LXF: Извините за это короткое отступление, но в эпизоде *Футурамы* [*Futurama*] «Загадка Фрая [The Why Of Fry]» были такие помешанные на данных пришельцы, которые хотели собрать «всю информацию по вселенной». В итоге они создали похожее на Звезду смерти [отсылка к «Звездным Войнам», — *прим. пер.*] биологическое хранилище памяти под названием Инфосфера, а оно принялось само себя сканировать.

AP: Круто.

LXF: Да, эпизод отличный, и отчасти связанный с моим следующим вопросом: а как вам удается сканировать Интернет без такого рода рекурсивных проблем? И какие еще проблемы возникают при сборе данных?

AP: Ну, возможно, в плане резервирования данных было бы и неплохо, если бы кто-то или что-то архивировало Wayback Machine, но мы, ясное дело, сами себя не исследуем. Как, впрочем, и многие другие данные, защищенные паролями и платными подписками. Мы не можем сохранять что-либо динамическое — например, весь Google, поскольку роботам это не под силу. Технологий, с которыми роботам трудно работать, в Сети всё больше и больше. Так что мы постоянно упираемся в стенки, и все же собираем около миллиарда страниц еженедельно.

LXF: Расскажите немного о программах сбора информации.

AP: С 2003 года, для сбора данных мы используем программу Heritrix с открытым кодом, которую мы разработали совместно с национальными библиотеками Nordic. Мы также экспериментируем

www.linuxformat.ru/subscribe

с некоторыми новыми средствами, более удобными в том плане, что они напоминают скорее браузер, а не просто робот. Но проблем по-прежнему полным-полно, и команда Archive It проводит кучу времени в поиске новых способов сбора информации и просмотра различных видов медиа. Это удобнее делать на менее масштабных проектах Archive It, поскольку там они могут плотнее взаимодействовать с клиентами, ведь в таких крупных подборках, с которыми работает моя команда, в миллиард страниц за неделю, гораздо сложнее понять, что ты что-то упустил.

В итоге, приходится много анализировать, чтобы понять, где есть пробелы. Изменения необходимы, чтобы маленькие победы сменялись большими. Архивировать Интернет становится все труднее.

LXF: Какое открытое ПО вы используете?

AP: Ну, Wayback Machine сама по себе открытый код, как и наши программы для чтения книг, да и весь наш стек, насколько это возможно. Естественно, все это на Linux. У нас есть сайт под названием Open Library (https://openlibrary.org), где содержится миллионный каталог книг, доступных для скачивания (если позволяет авторское право) или просмотра. Все это Open Source, и люди вносят сюда свой вклад. У нас был опыт открытого проекта, и в основном все изменения делали мы. Естественно, какие-то вещи мы выгоды ради не открываем. Но мы также используем *JWPlayer* для воспроизведения аудио и видео, и следим, чтобы все переводилось в открытые форматы — поэтому мы все еще переводим из MP3 в OGG, хотя многие его не используют.

Мы пытались поработать с Wikipedia — у них есть Wikimedia Commons, где хранится много медиа-записей, но им надо, чтобы это все содержалось в открытых форматах. А люди не умеют создавать файлы OGG, и в итоге Wikimedia много чего недополучает. И мы затеяли небольшой эксперимент с Wikimedia в Германии, разрешив людям загружать все, что угодно, хоть в MP4, хоть в WMV, а мы переводили это в OGG, чтобы Wikipedia забирала это и размещала в Wikimedia Commons. Я считаю, что такое переформатирование очень важно, даже при том, что открытые форматы на нашем сайте не пользуются самым большим спросом.

О ТОМ, КУДА ИДУТ ДЕНЬГИ

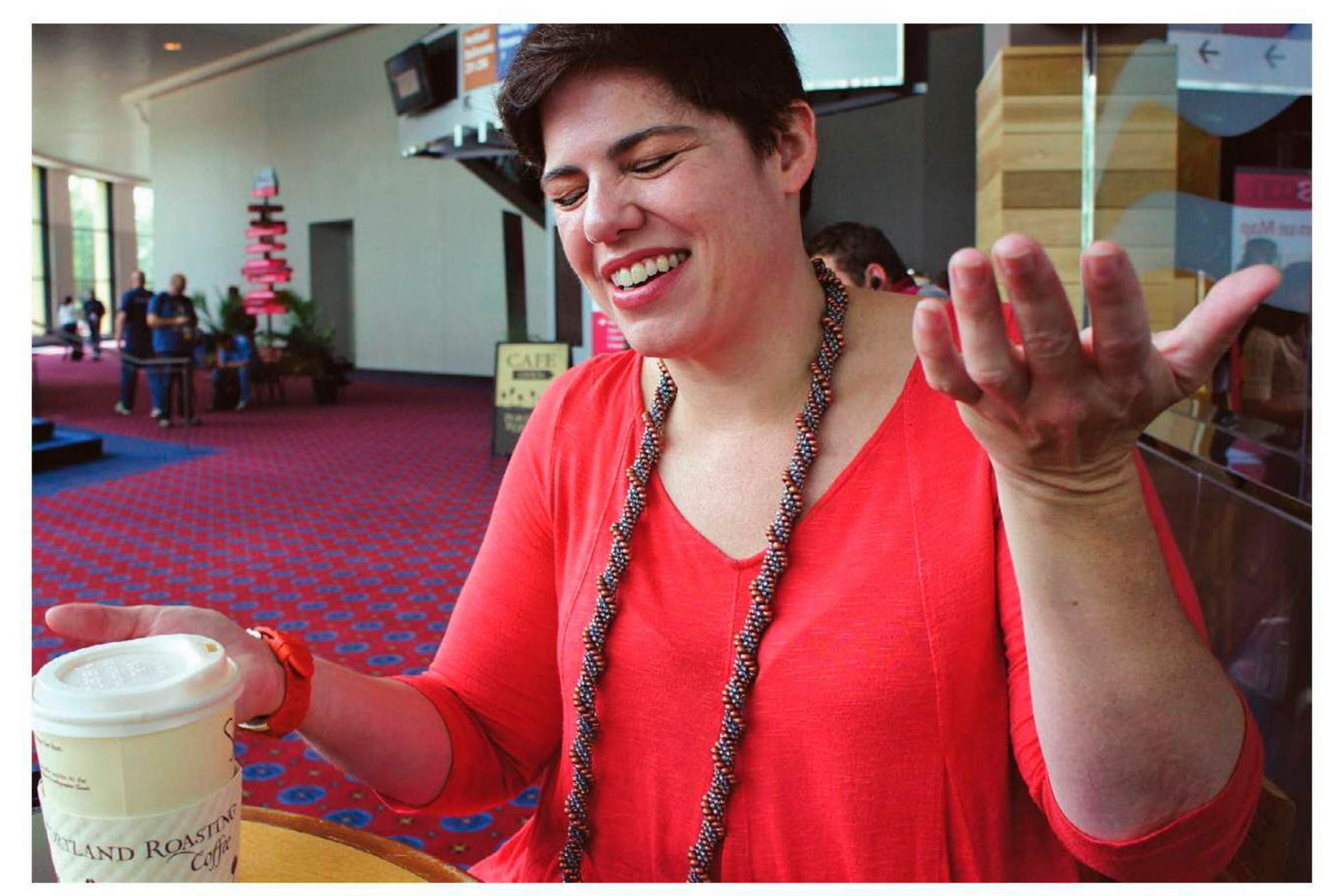
Вся работа, которую мы делаем за деньги... на благо общества.

LXF: Расскажите о вашем докладе на OSCON в этом году.

AP: Мы подготовили его совместно с Вики Брассер [Vicky Brasseur], которая раньше работала в Archive. Она живет здесь, в Портленде, и теперь является частным консультантом, но по образованию она инженер. Так вот я говорила об Internet Archive в целом, а она рассказывала об API, с помощью которых можно закачивать и выкачивать материалы. Мне кажется, это очень ценно для людей, организация у нас дружная, но не идеальная. У нас не самая лучшая документация, и 15 человек не дожидаются вашего запроса, готовые ответить. И я думаю, полезно выступать на таких мероприятиях, как OSCON, поскольку это дает людям шанс узнать, чем мы располагаем, и понять, что у нас можно запрашивать. Мы хотим помогать людям, а значит, хотим чтобы Archive пополнялся и был востребован.

LXF: На OSCON царит атмосфера разнообразия и дружелюбия, но каков ваш личный опыт в том, что значит быть женщиной в сфере технологии?

AP: Боже мой! Ну, сейчас все куда лучше. В 1996-м, когда я начинала, все было довольно, м-м, любопытно: бывало, меня просили покинуть собрание со словами «Ну, дальше, дорогая, уже наши технарские дела», или «секретарша ушла на перерыв, ты ее не подменишь — у нас же больше женщин нет, а на звонки отвечать надо?» Теперь это прекратилось, Internet Archive — компания очень разношерстная, кого только нет. Но мужчин по-прежнему больше, такова уж специфика инженерных талантов. Как и на подобных конференциях: сегодня с утра я встретила здесь около 400 человек, из них женщин — всего с десяток. LXF



Сергей Бронников

Живая миграция

Благодаря Игорю Штомпелю, *Linux Format* узнал о достижениях в живой миграции контейнеров с *OpenVZ* из первых рук — от Сергея Бронникова.

www.linuxformat.ru/subscribe

Сергей Бронников, менеджер Open-Source-проекта *OpenVZ*, занимается развитием *OpenVZ* и взаимодействием с сообществом. Ранее в компании Parallels как SQA-менеджер [SQA — Software Quality Assurance, контроль качества ПО] занимался тестированием облачного сервера *Virtuozzo* и *Virtuozzo for Windows* (ранее — *Parallels Cloud Server* и *Parallels Containers for Windows*).

Linux Format: **Расскажите, чем вы занимаетесь в Open Source.**

Сергей Бронников: Процесс разработки наших продуктов всегда тесно переплетался с открытыми проектами: это и непрекращающийся процесс интеграции патчей ядра *OpenVZ* в основную ветку ядра Linux, и публикация исходного кода компонентов *Virtuozzo* под свободной лицензией. А в последнее время мы стали более активным участником мира Open Source за счет участия в сторонних проектах.

Все проекты с открытым исходном кодом, в которых мы участвуем, можно поделить на две категории:

» Проекты, которые изначально были организованы сотрудниками компании, и результат этих проектов является частью наших коммерческих продуктов. Примеры таких проектов — уже успевший стать известным CRIU, проект P.Haul, библиотека для управления контейнерами *LibCT*, проект по управлению памятью в контейнерах *vcmmd*.

» Проекты, в которых мы участвуем на правах контрибьюторов: интеграция *Virtuozzo* с компонентами *OpenStack* (Nova, Cinder), разработка драйвера для *Virtuozzo* в проекте LibVirt, разработка недостающей функциональности в QEMU, участие в разработке библиотеки *runc* для запуска контейнеров.

Это все, что касается участия в разработке открытых проектов. А если говорить более глобально, то мы свою цель сформулировали давно: сделать наши контейнеры самыми популярными. И эта цель поставлена в виде ошибки номер один — OVZ-1 — в нашем публичном баг-трекере. Закроем ее как исправленную, когда поймем, что достигли цели.

LXF: **Сколько активных разработчиков на сегодняшний день?**

СБ: По понятным причинам, наиболее активными разработчиками компонентов *Virtuozzo/OpenVZ* являются сотрудники компании. Около 30 моих коллег заняты выпуском обновлений для стабильных версий продуктов и разработкой новых версий. Помимо штатных сотрудников, в разработке компонентов *Virtuozzo* участвуют инженеры, которые никак не связаны с компанией, и их количество всегда зависит от разных факторов: популярность и востребованность проекта, его зрелость, уровень знаний, необходимый для участия в проекте, и т.д.

Компонент может являться востребованным, но количество его активных разработчиков — малым. Как, например, с утилитой для управления контейнерами *vzctl*: активный разработчик только один — Кирилл Колышкин, а в разное время заплатки-патчи [patch] для *vzctl* присылали более 200 человек. То есть людей зачастую устраивала существующая функциональность, и они только исправляли баги или добавляли нужные им опции.

Миграция Linux-контейнеров — не единственный сценарий использования CRIU, технология находит применение и в других областях. Поэтому в разработке успели поучаствовать около 60 человек, хотя наиболее активными участниками являются около 13. Чтобы не утомлять вас сухими цифрами, советую посмотреть количество контрибьюторов для каждого компонента *Virtuozzo* в учетной записи *OpenVZ* на GitHub.

LXF: **В августе 2015 года на конференции Linux Plumbers Павел Емельянов представил P.Haul. В чем его особенности и назначение? Какое место эта технология занимает среди *OpenVZ* и CRIU?**

СБ: Технология P.Haul позволяет выполнять живую миграцию контейнеров с одного физического сервера на другой. С точки зрения реализации код живой миграции контейнеров всегда был сосредоточен в ядре Linux, а со стороны пользователя была утилита, которая только управляла миграцией. То есть все «волшебство» по замораживанию контейнеров происходило в ядре. Мы предпринимали попытки интегрировать свои патчи для «заморозки» и «разморозки» контейнеров в основную ветку ядра Linux, но по разным причинам этого сделать не получалось. Впрочем, этого не получилось и у других проектов, которые также были заинтересованы в добавлении функциональности сохранения и восстановления состояния процессов в «ванильном» ядре. Поэтому мы решили сделать ход конем: вынести большую часть кода из ядра Linux в пространство пользователя, а ядро использовать только как источник информации о процессах. У нас это получилось, и родился проект CRIU (Checkpoint and Restore In Userspace). Утилита, разрабатываемая в рамках этого проекта, позволяет «заморозить» процесс (или группу процессов), сохранить состояние в файлах на диске и потом восстановить состояние этих процессов из файлов. Процесс продолжит работать как ни в чем не бывало. Но сохранение и восстановление процессов — это только часть технологии «живой» миграции контейнеров. Основная сложность технологии в том, что нельзя просто так взять и «заморозить» процессы и начать переносить состояние этих процессов на другой сервер. Потому что это будет долго, и «живой» миграции не получится. Вместо этого надо либо сначала перенести память процессов, а потом «заморозить» их и перенести всё остальное; или сначала всё перенести, а память подключить как swap, чтобы она переносилась на другой сервер по мере востребованности процессами. На этом этапе и нужен P.Haul, который управляет процессом миграции. Дополнительно он может заниматься переносом файловой системы контейнера на другой сервер, если не используется разделяемое хранилище данных. Собственно, название проекта и отражает его назначение — Process Hauler, «перевозчик процессов».

О ПРОЕКТЕ CRIU

Решили вынести часть кода из ядра Linux в пространство пользователя.

LXF: **Реализована ли поддержка P.Haul за пределами *OpenVZ* — например, в *LXC* или других проектах?**

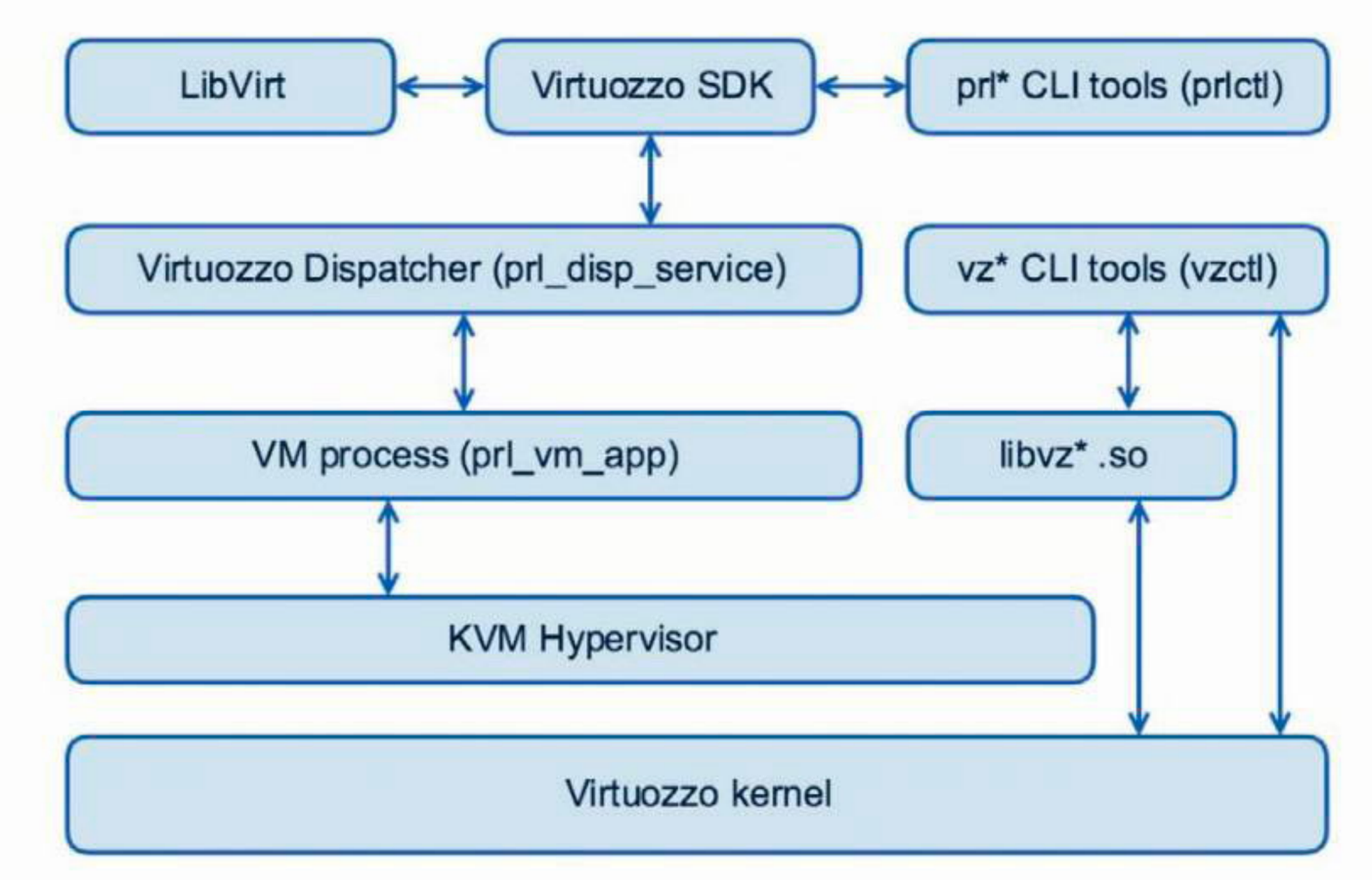
СБ: *OpenVZ* — не единственная реализация технологии контейнеров для Linux, поэтому при разработке новых компонентов мы стараемся их проектировать так, чтобы была возможность их использования в других проектах. Например, возможность заморозки и восстановления контейнеров с помощью CRIU была добавлена в *LXC* примерно год назад разработчиками Canonical при участии разработчиков CRIU. Разработка функциональности C/R для Docker продолжается, и есть надежда, что в этом году она появится в основной ветке репозитория. С проектом P.Haul похожая история, но относительно проектов все с точностью до наоборот. Патчи для интеграции P.Haul с Docker были приняты разработчиками P.Haul примерно месяц назад. А интеграция с *LXC* еще впереди: разработчики *LXC* выразили заинтересованность в интеграции с P.Haul и обещали сделать ее в ближайшее время. В *Virtuozzo 7* живая миграция будет реализована с помощью CRIU/P.Haul уже в Beta 3.

LXF: **Расскажите о портировании патчей в ядро RHEL. Как организован процесс, в чем его особенности? Как организовано взаимодействие проекта *OpenVZ* и Red Hat?**

СБ: Чтобы полнее описать процесс разработки, нужно сделать небольшой экскурс в историю разработки *OpenVZ*. На ранних этапах разработки контейнеров *Virtuozzo* мы использовали «ванильные» Linux-ядра. То есть на новый релиз ядра от Линуса Торвальдса мы накладывали наши патчи с реализацией контейнеров, тестировали своими силами и выпускали новое ядро *OpenVZ*. Но в 2006 году мы переключились на использование ядер из дистрибутива Red Hat Enterprise Linux и новое ядро стали делать на основе ядра из RHEL 4. Red Hat Enterprise Linux — это дистрибутив для предприятий от компании Red Hat, компании, чье имя всегда стоит на самом верху списка тех компаний, которые внесли самый большой вклад в разработку ядра Linux.

Конечно, никакое ядро не может быть идеальным и безошибочным, но ядро из их дистрибутива — довольно хороший пример качественного и стабильного ядра.

Разработка ядра для RHEL выглядит так: инженеры из Red Hat создают ветку последней версии «ванильного» ядра, для которой объявляется долгосрочная поддержка (такие ветки поддерживает Грег Кроа-Хартман), тестируют его, попутно »



› Рис. 1. Архитектура первой версии *Virtuozzo*.

исправляя найденные ошибки, и портируют новые драйвера и исправления проблем в безопасности из основной ветки. Такая «подготовка» ядра к релизу продолжается примерно полгода или немногим больше. Так что к моменту релиза это ядро уже «древнее и устаревшее» — во всяком случае, так кажется, если смотреть на его версию. На самом деле никакое оно не древнее и не устаревшее, а просто более стабильное и безопасное. После релиза это ядро очень хорошо поддерживается — добавляется поддержка современного «железа», оперативно исправляются проблемы в безопасности. И все это на протяжении нескольких лет.

После релиза новой версии дистрибутива RHEL мы берем исходный код Linux-ядра этого дистрибутива и кропотливо начинаем переносить все наши патчи с предыдущей версии на новое ядро. Мы этот процесс называем «ребейзом [от *англ.* rebase]» на новое ядро. Этот процесс не всегда прост, потому что мажорные версии ядер содержат очень много изменений в разных подсистемах ядра и часть патчей приходится переделывать, а часть из них не появляется в новом ядре из-за того, что нужная функциональность появилась в основной ветке Linux-ядра. Так получилось с «живой» миграцией, поддержкой NFS в контейнерах, пространствами имен IPC, MNT, UTS, PID и NET, технологией управления памятью в контейнерах, то есть вся та функциональность, которую мы ранее «продали» в upstream [в «вышележащее» ПО, основную ветку]. До 2015 года мы весь исходный код ядер *OpenVZ* выкладывали на сайте в виде архивов; из-за этого следить за процессом разработки и сравнивать исходный код разных версий было неудобно. Поэтому разработку на базе ядра RHEL7 мы ведем в публичном репозитории, и все патчи проходят через почтовый список рассылки devel@openvz.org. В репозитории ядра все «переезды» на новую версию ядра RHEL отмечаются тэгами: 19 февраля 2015 года мы начали использовать ядро RHEL7 kernel-3.10.0-123.1.2.el7, а текущая версия нашего ядра базируется на версии rh7-3.10.0-229.7.2. В разработке ядра *OpenVZ* мы активно используем ветки, это позволяет нам разделять функциональность разной степени готовности и реже ломать сборки продукта.

Нужно отметить, что сотрудничество разработчиков *OpenVZ* и Red Hat взаимовыгодное: мы используем их качественные ядра, но сообщаем обо всех найденных проблемах во время тестирования. В последних анонсах Red Hat отметил Владимира Давыдова за обнаружение серьезных уязвимостей CVE-2014-0203 и CVE-2014-4483 в последнем обновлении ядра RHEL6 (вторая проблема, кстати, была найдена с помощью одного из наших автоматических тестов, использующих Linux Test Project). Василий Аверин получил благодарность за обнаружение ошибки CVE-2014-5045, а Дмитрий Монахов — за CVE-2012-4508 и CVE-2015-8324.

LXF: Расскажите об архитектуре проекта *OpenVZ*. В чем ее особенности? Как устроен *OpenVZ* изнутри, что представляет собой на уровне исходного кода — патчи к ядру или комбинацию патчей и другого, а также на уровне исполнения — в чем выражается его работа как программы?

СБ: Процесс разработки я подробно описал в ответе на предыдущий вопрос.

Архитектура *OpenVZ* стабильной версии не представляет собой ничего интересного и сводится к взаимодействию двух компонентов: ядра Linux и пользовательской утилиты *vzctl*, которая управляет контейнером с помощью системных вызовов к ядру. Архитектура новой версии *Virtuozzo* гораздо интереснее. Но прежде чем рассказывать про компоненты *Virtuozzo*, я сделаю небольшой экскурс в историю разработки *Virtuozzo*. Первая коммерческая версия продукта с Linux-контейнерами имела одинаковую с *OpenVZ* архитектуру: ядро Linux и пользовательские утилиты. А первая версия *Virtuozzo* (она же *Parallels Cloud Server* и *Parallels Server Bare Metal*), которая объединила два разных типа виртуализации, уже имела более продвинутую архитектуру, на базе клиент–серверной модели: диспетчер, управляющий виртуальными машинами и контейнерами, и интерфейс клиента в виде GUI и консольных утилит, работающих через SDK *Virtuozzo* (как показано на рис. 1).

Такая архитектура позволила предоставлять API для интеграции с решениями других компаний. В ранних версиях *Virtuozzo* (4, 5) утилиты для работы с контейнерами и виртуальными машинами были недостаточно интегрированы; для операций с виртуальными машинами использовались универсальные идентификаторы, а для контейнеров — номера, но в *Virtuozzo 7* мы исправили эти недостатки. Для обоих типов окружений можно будет использовать одинаковые утилиты: *prlctl*, *pbackup* и т.д.

LXF: Реализацию какой функциональности можно ожидать в Virtuozzo в ближайшее время? Что в долгосрочных планах?

СБ: Переход на использование *KVM* и *cgroups*.

LXF: Частично вы об этом говорили, но хотелось бы заострить внимание. Скажите, какие ветки ядер *OpenVZ* поддерживаются на данный момент? Чем они различаются?

СБ: Так как ядро *OpenVZ*/*Virtuozzo* базируется на ядре от Red Hat, то мы различаем пять веток ядер *OpenVZ*:

» по две ветки -testing и -stable для ядер, базирующихся на RHEL5 и RHEL6. Эти ядра используются в текущей версии *OpenVZ* и коммерческих версиях *Virtuozzo* 5 и 6.

» ветка -testing с ядрами, основанными на RHEL7. Эти ядра попадают в тестовые сборки дистрибутива Virtuozzo 7, который находится в разработке.

Если говорить о ветках в репозитории исходного кода ядра *OpenVZ*, то там чуть сложнее. Для каждого «ребейза» на новое ядро Red Hat, для серьезных изменений и новой функциональности мы делаем отдельную ветку в репозитории (рис. 2).

LXF: Расскажите о масштабировании в *OpenVZ*.

СБ: Контейнерная виртуализация в Linux — это комбинация технологии изоляции процессов (пространства имен) и технологии управления ресурсами (CGroups). Каждая из этих технологий не меняет архитектуру ОС. Как следствие, контейнеры *OpenVZ*/*Virtuozzo* масштабируются точно так же, как и само ядро Linux — до тысячи CPU и терабайтов памяти. В разработке *OpenVZ*/*Virtuozzo* мы исходим из практических задач, а основной сценарий использования наших контейнеров — это услуги хостинга. Хостинг-провайдер всегда заинтересован в том, чтобы разместить максимальное количество изолированных окружений на одном физическом сервере, чтобы получить максимальную прибыль. Поэтому одна из метрик качества *Virtuozzo* — это плотность размещения контейнеров. Мы измеряли плотность размещения контейнеров на серверах разных конфигураций (вплоть до 64 CPU, 256 ГБ ОЗУ) и всегда упирались в ресурсы сервера (процессор, диск или память), но не проблемы масштабирования *Virtuozzo*.

LXF: В чем отличие *OpenVZ* от других технологий виртуализации (*VMware*, *Хел* и др.)? Преимущества?

СБ: Основное отличие *Virtuozzo* от похожих решений — это возможность использования обоих

› Рис. 2. Дерево версий исходного кода ядра *OpenVZ* в репозитории.

типов виртуализации в рамках одного продукта. Больше не нужно выбирать между виртуальными машинами и контейнерами. *Virtuozzo* позволит вам использовать как безопасные Linux-контейнеры с хорошей производительностью и высокой плотностью, так и ВМ. Улучшения в интеграции коснулись и пользовательских утилит (*prlctl*, *prl_disk_tool*) и API (Virtuozzo SDK, общий драйвер LibVirt).

Примеры использования утилиты *prlctl* для создания и старта контейнера и виртуальной машины:

```
prlctl create LinuxFormat_1 --vmtype=ct; prlctl start LinuxFormat_1
prlctl create LinuxFormat_2 --vmtype=vm; prlctl start LinuxFormat_2
```

Вообще в последнее время появилось много продуктов для виртуализации. И подчас даже продвинутому пользователю трудно разобраться во всех плюсах и минусах похожих решений. Мы решили помочь пользователям и подготовили таблицу на сайте проекта, в которой сравнили функциональность коммерческих версий *Virtuozzo* (5, 6, 7), бесплатных версий (*OpenVZ* и *Virtuozzo 7*), *LXC*, *MS Hyper-V*, *RHEV* и *Citrix Xen*. Несмотря на то, что таблица содержит только утвержденную функциональность «семерки», она позволит вам понять, какие задачи вы сможете решать с помощью *Virtuozzo* и достаточно ли вам будет функциональности открытой версии. Основными отличиями между коммерческой и бесплатной версиями *Virtuozzo 7* будет дополнительная поддержка гостевых ОС семейства Windows (драйвера для гостевых ОС будут снабжены сертификатами Microsoft), web-интерфейс для управления серверами *Virtuozzo*, отказоустойчивое и распределенное хранилище Virtuozzo Storage, возможность использования кластера с High Availability и каталог образов для виртуальных машин и контейнеров.

Доказывать, что *Virtuozzo* лучше, чем имярек, дело неблагодарное, потому что лучше отталкиваться от задач, которые планируется решить с помощью виртуализации. Ситуация с контейнерами похожа на состояние дел в автопроме: кто-то из автопроизводителей проектирует свои двигатели, тестирует их, внедряет новые технологии и т.д. А кто-то покупает уже готовые технологии и занимается только производством. Мы занимаемся контейнерами с 2000 года (как для Linux, так и для Windows), и за 15 лет накопили немалый опыт в этой области. Новые технологии Linux-контейнеров появлялись в наших продуктах, когда не было аналогов: «живая» миграция контейнеров, файловая система контейнера в файле (ploop) и т.д. Пространства имен, которые являются базовыми технологиями Linux-контейнеров, появились в «ванильном» ядре благодаря работе наших инженеров. А сейчас используются в любом продукте с контейнерами. Это всё стало следствием того, что вся функциональность разрабатывалась с упором на реальную эксплуатацию решения, а не на предоставление механизмов для реализации той или иной возможности, как поступают в Linux upstream containers.

LXF: Будет ли расширяться поддержка «гостевых» ОС?

СБ: Мы регулярно добавляем поддержку новых гостевых ОС. Обычно поддержка новой ОС появляется в следующем обновлении *Virtuozzo* после выхода этой ОС, хотя тестирование новых версий гостевых ОС мы начинаем задолго до выхода финальной версии. *Virtuozzo 7* включит поддержку актуальных версий ОС семейства Windows и Linux.

LXF: Планируется ли расширение пользовательского набора утилит?

СБ: Судите сами. Для текущей версии *OpenVZ* мы поддерживаем три основные утилиты:

» **vzctl** для управления контейнерами;
» **ploop** для управления образами дисков;
» **vzstats** для сбора статистики об использовании *OpenVZ*.

В *Virtuozzo 7*, помимо ploop и vzctl появляются:

» **vzstat** — для мониторинга состояния виртуальных машин и контейнеров.
» **vztt** — для управления шаблонами контейнеров.
» **prlsrvctl** — для управления параметрами всего сервера *Virtuozzo*.
» **prl_disk_tool** — универсальная утилита для управления образами виртуальных машин и контейнеров.
» **pbackup** — универсальная утилита для создания резервных копий виртуальных машин и контейнеров.
» **prlctl** — универсальная утилита для управления контейнерами и виртуальными машинами. Эта утилита в следующей версии полностью заменит *vzctl*, поэтому мы настоятельно рекомендуем нашим пользователям использовать именно ее, а не *vzctl*.
» **prl_nettool** — для управления сетевыми настройками гостевой операционной системы и ОС внутри контейнера.
» **vcmmd** — для управления памятью контейнеров.

В будущем, по мере развития продукта, мы не исключаем появления новых утилит. Исходный код всех перечисленных выше утилит доступен в основном репозитории src.openvz.org и зеркалах на GitHub — http://github.com/OpenVZ.

LXF: Что можно сказать о развитии официально неподдерживаемых утилит?

СБ: Параллельно с развитием *OpenVZ* появилось множество утилит, которые развивались вне проекта *OpenVZ*. По большей части эти утилиты дублировали функциональность утилит из платной версии *Virtuozzo* — например, утилиты для создания резервных копий контейнеров. Так как эти утилиты развивались вне проекта и людьми, не связанными с проектом, то о планах по развитию лучше спросить авторов этих утилит. Мы в свою очередь, насколько это возможно, стараемся не ломать совместимость в новых версиях *Virtuozzo*, чтобы оставалась возможность использования этих инструментов. LXF

Пробуем тестовые версии Virtuozzo 7

Разработка *Virtuozzo 7* в самом разгаре, но так как процесс разработки открыт, то все тестовые сборки доступны для тестирования (https://download.openvz.org/virtuozzo/factory/x86_64/iso/). Наличие сервера для этого совершенно не обязательно.

Можно использовать образ виртуальной машины для *Vagrant*:

```
$ vagrant init OpenVZ/Virtuozzo-7.0
$ vagrant up --provider virtualbox
$ vagrant ssh
```

Или использовать образ для виртуальной машины в Amazon EC2: надо в AWS Marketplace (https://aws.amazon.com/marketplace) найти образ *Virtuozzo 7* и запустить экземпляр [instance], используя этот образ. Использование экземпляра с минимальными параметрами (t2.micro) будет бесплатным.

Искусство Open Source

Открытый код питает все отрасли искусства. Джим Тэккер изучает, как *Blender* покоряет анимацию и киноэффекты.

Blender использовался при создании анимаций для национального телевидения и рекламы для Coca-Cola, Pizza Hut и BMW. Он создает первоклассные маркетинговые образы для брендов от Puma до Philippe Starck. Он даже использовался для фильмов, номинированных на Оскар. И что лучше всего, это ПО с открытым кодом.

Blender — это классическая история аутсайдера. Начавшись как набор 3D-инструментов для внутреннего пользования маленькой голландской фирмы анимаций, на ранних стадиях он пережил финансовые трудности и даже крах своего первого дистрибьютора, а затем добился признания широкой публики. Превысившее четыре миллиона загрузок ежегодно, сейчас это наиболее широко используемое в мире ПО 3D.

Но что важнее для целей данной статьи, это ПО, завоевавшее уважение профессиональных художников. Некогда отметавшийся в сторону как инструмент для любителей, *Blender* ныне восхваляется крупнейшими мировыми анимационными компаниями. Каких-то пять лет назад новость, что в приложении добавилась очередная новая функция, обычно встречала на тематических форумах ответ «Да, но... это *Blender*». Сейчас же ответ звучал бы скорее как «Вот бы моя программа такое могла».

Может, *Blender* и не лидер рынка; для создания большей части профессиональной анимации, визуальных эффектов и проектов по разработке игр все еще применяются коммерческие инструменты, в частности, разработанные Autodesk — по крайней мере, на Западе. Но он способен на прекрасную работу.

> Blender, классическая история аутсайдера, сейчас наиболее широко используемое ПО 3D.

На следующих 4-х страницах мы встретим компании, применяющие *Blender* для коммерческих проектов, от иллюстраций к коробкам для хлопьев до визуальных эффектов для *Red Dwarf*. Мы изучим, как ПО питает международную сеть анимационных студий на всех континентах, кроме Антарктики. И даже попробуем ответить на вопрос: «Если *Blender* так хорош, почему он не используется в большем количестве фильмов Голливуда?»

Но прежде чем мы сделаем это, отведем время на краткий урок истории...

› Полноценный конвейер 3D-анимации, *Blender* — идеальный инструмент для рекламы CG, как, например, долгоиграющие серии рекламных роликов для шин Bridgestone от Character Mill.

› *Big Buck Bunny* (2008 г.) стал переломным короткометражным фильмом в истории Blender Institute. Его достояние — тоже открытое — повторно применялось везде, от детских книг до рекламы телефона Google Nexus S.

На протяжении всей своей истории *Blender* был неразрывно связан с одним человеком: это голландский программист, продюсер, универсал и всемирный бродяга Тон Роозендааль [Ton Roosendaal]. Именно он стал соучредителем NEOGEO, анимационной студии, где в 1995 г. *Blender* возник как домашний 3D-инструментарий.

Изначально казалось, что *Bender* суждено стать куда более заурядным продуктом: Not a Number [Не число], компания, которую Роозендааль запустил в 1998 г. на рынок ПО, сперва планировала производить бесплатную версию для создания онлайн-контента и платную — для всего остального.

Но это не очень-то сработало. Даже тогда рынок ПО 3D был насыщен именитыми конкурентами, включая ранние версии ныне вездесущего *Maya* от Autodesk и *3ds Max*. Ранние продажи *Blender* разочаровывали, и в 2002 г. инвесторы Not a Number решили махнуть на все рукой.

Оказавшись перед выбором — похоронить плоды семилетней работы или перевести ее в открытый код, Роозендааль предпочел открытый код. Краудфандинговая кампания быстро собрала €100 000, необходимые для выкупа кодовой базы, и 13 октября 2002 г. *Blender* вышел под лицензией GPL.

Став председателем зарождающегося Blender Foundation, Роозендааль начал ускорять разработку ПО с помощью серий «открытых фильмов» — созданных при помощи краудфандинга анимированных короткометражек, используемых как платформа для разработки новых инструментов, годных для серьезного производства. Сами по себе фильмы вышли под лицензией Creative Commons, поскольку для их создания были использованы все активы CG.

К 2007 г. у Роозендааля была постоянная студия, Blender Institute, для осуществления надзора за работой. В следующем году в Institute случился первый резкий подъем продаж комедийного короткометражного фильма *Big Buck Bunny*. Его подбор персонажей-животных доказал свою неотразимость, когда изображения из фильма повыскакивали везде, от брошюр для бойскаутов Америки до рекламы гуглофонов.

Последующие открытые фильмы помогли укротить специфический интерфейс пользователя *Blender* и разработать новый инструментарий для работы с визуальными эффектами в режиме live. К 2014 г. Institute был готов к своему самому амбициозному проекту: полнометражному открытому фильму под кодовым названием «Проект Крыжовник [Project Gooseberry]».

К сожалению, Крыжовник не сумел достичь своей цели в €3,5 миллиона и в конечном итоге был свернут до сюрреалистичного анимационного короткометражного фильма *Cosmos Laundromat — First Cycle*, вышедшего в 2015 г. Но, тем не менее, краудфандинг стал самым успешным в истории для анимационного проекта, причем представленный список компаний-партнеров *Blender* показал нечто очень важное: уже на тот момент существовала целая международная сеть студий, использующих *Blender* для коммерческого производства. »

Другие ключевые инструменты анимации с открытым кодом

Даже Blender Institute использует в своих анимациях не только *Blender*. Представляем другие инструменты с открытым кодом, использовавшиеся для создания «открытых фильмов». Мы подробно рассмотрим некоторые из них в будущих статьях этой серии.

› Krita (krita.org)
Главный инструмент Institute для концепт-арта и раскадровки, *Krita* пережил новое рождение. Его недавняя успешная кампания на Kickstarter стремится сделать его «быстрее, чем *Photoshop*».

› GIMP (gimp.org)
Хотя *GIMP* не видел крупного обновления добрых три года, это все еще надежная рабочая лошадка для общих задач редактирования изображений, таких как создание текстурных карт, используемых в 2D-анимации, или корректировки предоставленных кадров.

› LibreOffice (libreoffice.org)
Вездесущий офисный пакет, используемый в Institute для ряда задач: *LibreOffice Draw* часто применяется для сборки раскадровок под печать или распространения в формате PDF.

› Flamenco (flamenco.io)
Разработанный для использования в последних короткометражных фильмах Institute и все еще пребывающий на стадии beta, этот менеджер рендеринга распространяет анимационные работы по сети компьютеров, используемых для их рендеринга. С виду не эффектно, но жизненно важно.

› Attract (cloud.blender.org)
Также разработанный для *Cosmos Laundromat*, инструмент управления кадрами *Attract* делает очередную тяжкую работу: отслеживает, кто и чем занимается в процессе производства. Немного позже он станет частью Blender Cloud.

› Неиспользованное концептуальное изображения для короткометражки *Cosmos Laundromat* от Institute, созданные в *Krita* Давидом Ревуа [David Revoy].

› Каждый фильм на *Blender* выступает как платформа для добавления новых функций в ПО. Сюрреалистичный *Cosmos Laundromat* (2015 г.) добавил в этот коктейль инструменты управления проектом.

› Большинство фильмов Blender Institute были анимационными короткометражками. *Tears of Steel* (2012 г.) доказал, что ПО способно также справляться с визуальными эффектами в «живой» съемке.

Одна из этих студий — London Gecko Animation (www.geckoanimation.com), чьи работы разнятся от рекламных роликов для Puma до спецэффектов для комедийных сериалов *Red Dwarf*.

«Я натренировался в *Maya*, когда работал на другие компании, — говорит соучредитель студии, Джонатан Лэкс [Jonathan Lax]. — Но со временем все больше и больше интересовался открытым кодом. Более крупные программы не помогут вам, если вы работаете над множеством различных частей программ, чем обычно приходится заниматься небольшой команде».

Коммерческое ПО, как правило, более жестко нацелено на выполнение специализированных задач, тогда как *Blender* обеспечивает полную настройку студии на компьютере, в том числе инструментов для создания и анимации 3D-моделей, рендеринга готовой анимации и даже ее интеграции в кадры живой съемки или редактирования видео.

«*Blender* очень гибок, — говорит Лэкс. — Поскольку эта программа делает так много и так великолепно это интегрирует, можно работать над несколькими участками производства одновременно».

Вовсе не грех, что *Blender* бесплатен. Однако это не главная причина использовать его, настаивает Лэкс. «Когда вы только начинаете, очень удобно иметь ПО, не требующее лицензии. Но то, что в разработке открытого кода не крутятся деньги — заблуждение. Я вложил в *Blender* через пожертвования Blender Foundation больше денег, чем в любое коммерческое программное обеспечение».

Победа над коммерческими

Одна из причин для этих пожертвований — обеспечение своевременного исправления ошибок, еще одно ключевое преимущество *Blender*, по словам Джеймса Нила [James Neale], основателя Sydney Character Mill (http://charactermill.com) — студии из пяти сотрудников, работающей над проектами от анимированных ТВ-роликов до книжных иллюстраций.

«Разработчики *Blender* обычно сами используют свои инструменты, и им легко понять наши требования по исправлениям, — говорит Нил. — Если я сталкиваюсь с ошибкой или мне нужен обходной путь, я могу обратиться к сообществу и получить ответ в течение 15–20 минут».

Как и Лэкс, Нил раньше использовал коммерческие программные продукты, но говорит, что когда дело дошло до клиентской поддержки, потребности более крупных студий отодвигали студии меньшего масштаба на второй план. «Мы обнаружили, что сообщество *Blender* куда лучше откликается на наши запросы, чем поставщики коммерческого ПО, — утверждает он. — За 10 лет применения проприетарного ПО 3D мы не получили никакой полезной поддержки ни от поставщика ПО, ни от перепродавца — разве что проблемы с регистрацией наших лицензий».

Blender также все лучше и лучше работает с другими инструментами, необходимыми студии для коммерческих графических работ. Большинство движков рендеринга, используемых для конвертирования сырых 3D-данных в готовые 2D-изображения, теперь поддерживают ПО — включив, ранее в 2015 г., *RenderMan*, получивший награду Pixar Academy.

На самом деле, поддержка в *Blender* ряда сторонних инструментов лучше, чем у коммерческого ПО, поскольку тот факт, что исходный код свободно доступен, упрощает необходимую доработку.

Открытая технология и Голливуд

Компании Big Hollywood, занимающиеся визуальными эффектами и анимацией, часто пишут собственное проприетарное ПО 3D. Но это не значит, что они не используют технологию открытого кода.

Industrial Light & Magic задала тон, открыв в 2003 г. код OpenEXR (http://openexr.com), это проприетарный формат файла для HDR-изображений, сейчас хорошо знакомый большинству фотографов.

Sony Pictures Imageworks последовала примеру по целому набору технологий с открытым кодом (http://opensource.imageworks.com), включая Alembic, который теперь является промышленным стандартным форматом для передачи сложной 3D-геометрии между программными пакетами.

Disney сделал то же самое с *Ptex* (http://ptex.us), который управляет отображением на поверхности 2D-карт текстур, определяющих локальную окраску 3D-модели, а DreamWorks выпустили библиотеку *OpenVDB* (http://openvdb.org) для объемных эффектов вроде клубящихся облаков. Вы можете видеть такие облака внизу стоп-кадра из *Как приручить дракона 2* [*How to Train Your Dragon 2*].

› *OpenVDB* обеспечивает стандартный формат для объемных эффектов, вроде этих облаков из *Как приручить дракона 2*.

Крупные студии открывают код своих технологий, поощряя наличие стандартных форматов, в которых данными можно обмениваться, когда компании сотрудничают над фильмом, как часто бывает в случае с визуальными эффектами — но преимущества ощущались каждым, поскольку они быстро внедряются в готовые к работе программы.

Сам по себе *Blender* использует несколько этих библиотек кода: последний релиз, *Blender 2.76*, внедряет *OpenSubdiv*, технологию Pixar для «высокопроизводительной оценки подразделения поверхности». (Помимо всего прочего, она позволяет аниматорам более точно предварительно просматривать движения их персонажей).

Судя по всему, работа нашла одобрение даже в Pixar: там перевели *OpenSubdiv* с лицензии Microsoft Public Licence на Apache Licence, чтобы *Blender* смог использовать ее технологию.

www.linuxformat.ru/subscribe

Нил использует для создания интерактивных приложений популярный игровой движок *Unity*, и указывает на то, что он может прямо прочесть файлы сцены *Blender*, а не требовать экспортировать данные в промежуточный формат файла. «Вы просто сохраняете файл **.blend** в структуру папки и используете в качестве актива, — говорит он. — Это не будет работать с файлами сцены *3ds Max* или *Maya*».

Но если *Blender* так хорош, почему его не использует больше студий? Известно, что из пяти номинаций на Оскар за лучший анимационный фильм в 2015 г. только один — независимый ирландский анимационный фильм *Песнь моря* [*Song of the Sea*] — использовал *Blender*, тогда как Autodesk может заявить, что его программы сыграли «значительную роль» в создании оставшихся четырех.

В словах «известно, что» скрывается нюанс, к которому мы вернемся позднее. Но в общем, студии, использующие *Blender* в качестве основного инструмента производства, более мелкие, чем использующие коммерческое или проприетарное ПО — обычно из 20 или менее сотрудников, в отличие от сотен художников, работающих над крупным голливудским фильмом.

Одной из причин Джеймс Нил называет нехватку соответствующего персонала. Множество художников, нанятых Character Mill, знакомы только с коммерческим ПО, и работе в *Blender* их нужно учить.

«Основное ограничение — это поиск компетентных людей, которые постоянно используют *Blender*, — говорит Нил. — Каждый раз, когда мы берем кого-то нового, нам приходится его обучать; если у нас жесткие сроки, нам нужен человек, способный справиться с этой работой с ходу».

Преодоление предрассудков

Blender Network, социальная сеть для профессионалов *Blender*, запущенная в 2012 г. Blender Foundation, теперь значительно упростила прием на работу специалистов со всего мира, но Нил говорит, что ПО следует более широко изучать в школах и университетах. «Художников *Maya* так много просто потому, что этому учат в школах, — поясняет он. — А раз есть большой выбор художников, крупные студии продолжают пользоваться ею. Это проблема курицы и яйца».

› Фотореалистичные визуальные эффекты Gecko Animation для *Words Can Hurt*, рекламы для NSPCC [Национальное общество по защите детей от насилия, — *прим. пер*], в которой особо выделялась эмоциональная травма, были полностью созданы в *Blender*.

Эту проблему Blender Foundation теперь решает с помощью инициативы Blender 101; ее цель — разработать версию *Blender* для обучения старшеклассников 3D. Но другой камень преткновения преодолеть труднее: это устоявшееся убеждение, что использование «свободных» инструментов непрофессионально. «Всегда есть некое клеймо на использовании ПО с открытым кодом, — говорит Джонатан Лэкс. — Я основываю всё на качестве конечного результата, [но] большинство клиентов выражают опасения, когда я сообщаю о ПО, которое применяю».

По словам Тона Роозендааля, это проблема усугубляется тем, что множество художников из Азии и Южной Америки — где на данный момент располагаются самые крупные студии, использующие в своем производстве инструменты с открытым кодом — перешли на *Blender* со взломанных копий коммерческого ПО. «Они перестали быть нелегальными пользователями, превратившись в законопослушных, но не изменили своей привычке не обращаться за поддержкой и испытывать неловкость за употребление свободного ПО», отмечает Роозендааль.

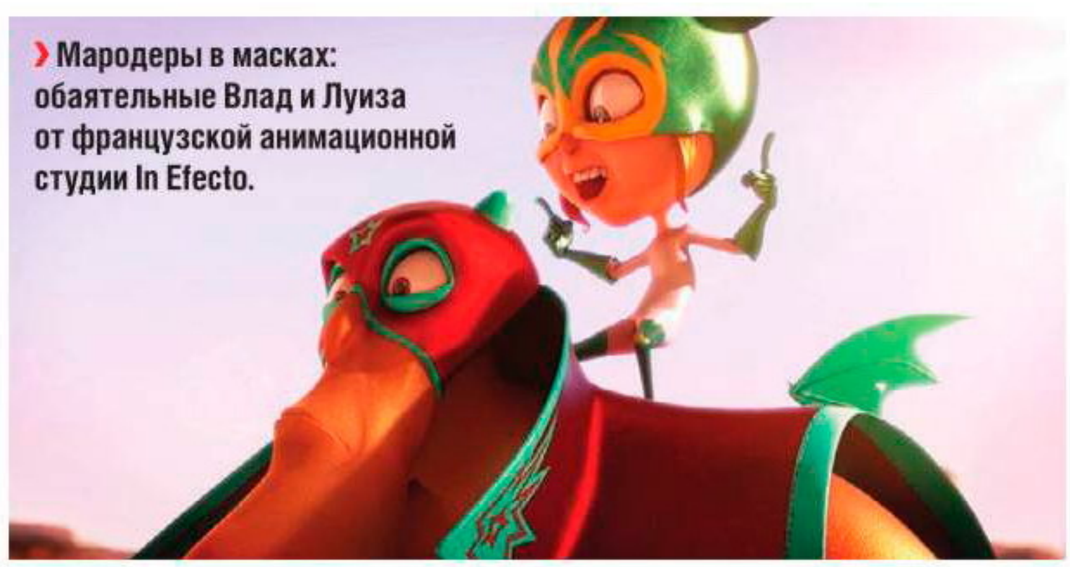
Нежелание открыто высказываться усложняет получение реального представления о том, как широко используется *Blender* в производстве. Когда мы собирали материал для этой статьи, нам очень часто говорили о том, что множество художников в крупных игровых студиях и студиях визуальных эффектов на самом деле используют ПО, но не особо об этом распространяются.

Но когда-нибудь это может измениться. В 2015 г. Pixar объявила о том, что *Blender* является одним из малого числа 3D-приложений, готовых к использованию и официально поддерживаемых для использования в студии, и упомянула эту программу в пресс-релизе по поводу последней версии *RenderMan*. »

Пять отличных анимаций Blender со всего мира

Babioles
http://bit.ly/babioles
Показанный на французском Canal+, этот анимированный сериал показывает три очаровательные игрушки и жуткие вещи, которые происходят с ними. Режиссер Матье Овре [Mathieu Auvray] выложил около 30 серий на YouTube.

Vlad & Louise
http://bit.ly/vladlouise
Еще один анимационный сериал — Vlad & Louise, его героиня — восьмилетняя девочка, которая находит выигрышный билет в коробке хлопьев и становится мексиканской рестлершей в стиле вольной борьбы. Французская студия In Efecto планирует выпустить 52 серии по 13 минут, но пока вам остается только довольствоваться милым трейлером.



› Мародеры в масках: обаятельные Влад и Луиза от французской анимационной студии In Efecto.

Adit & Sopo Jarwo
http://bit.ly/aditsopojarwo
Возможно, крупнейшая студия (в ней работает более 300 человек), использующая *Blender* в качестве основного инструмента производства, индонезийская MD Animation производит огромное количество анимационных сериалов, включая удостоенный наград *Adit & Sopo Jarwo*. Ролики можно найти на YouTube.

Song of the Sea
http://bit.ly/songoftheseamovie
Этот ирландский анимационный фильм получил хвалебные отзывы после выхода — не говоря уже о номинации на Оскар за лучший анимационный фильм. Большая часть анимации выполнена в 2D, но некоторые транспортные средства и эффекты были созданы с применением *Blender*.

Sabogal
http://3da2animation.com/sabogal
Этот жесткий триллер — не самый первый художественный фильм, который создан преимущественно в *Blender* (самым первым был аргентинский *Plumíferos*), но, бесспорно, лучший из таких фильмов.

И, как указал Роозендааль, то, что крупные студии обычно более охотно говорят о коммерческих 3D-инструментах, не означает, что они применяют их в качестве основных для производства. «У людей есть нечто вроде романтической фантазии о том, что крупные студии применяют для создания фильмов настольное [коммерческое] ПО, и с чего бы им применять *Blender*, — говорит он. — Но это не так. Вряд ли у них такое ПО. У них есть свои инструменты для внутреннего употребления».

«Студии вроде Pixar великолепны, но они — лишь вершина пирамиды, — добавляет он. — Ниже находятся миллионы людей, работающих с 3D, и этим людям тоже нужны инструменты».

Что ждет Blender дальше?

Этот год, похоже, будет значимым для *Blender*. Отказавшись от недавнего цикла небольших трехмесячных обновлений, Blender Foundation хочет взять годичный отпуск, чтобы сфокусироваться на крупных проектах, нацеленных на подготовку ПО и пригодных даже для более крупных производств.

Это, как признает Роозендааль, колоссальная задача для некоммерческой организации, которая сейчас поддерживает менее 10 штатных сотрудников, особенно учитывая возрастающую сложность работы. «Многие из этих улучшений невозможны в обычном проекте открытого кода, над которым трудятся добровольцы, — говорит он. — Придется принять на борт профессионалов. И единственный способ сделать это — обратиться в студии и сделать их участниками будущих разработок *Blender*».

В конечном итоге, цель Роозендааля — создать полнометражный открытый фильм: и как окончательное доказательство возможностей *Blender*, и как средство дальнейшей стимуляции сообщества профессиональных пользователей.

«Найдя достойное финансирование — скажем, $12–15 млн — чтобы сделать качественный анимационный художественный фильм, мы сможем принять разработчиков *Krita* и *GIMP*, [и нанять] больше разработчиков *Blender*, — говорит он. — С таким фильмом плюс конвейером с полностью открытым кодом начинается нечто очень интересное. Можно даже организовать обучение технических специалистов, продолжающих работу в других студиях».

Вопросов остается масса. Но, глядя на то, чего достиг *Blender* за последние 20 лет, вы, скорее всего, захотите сделать на него ставку. LXF

› 10-й сезон *Red Dwarf* смешал в себе традиционные для научно-фантастической комедии миниатюры с цифровыми эффектами *Blender*, сделанными Gecko.

Полезные ресурсы

› **www.blender.org** Официальный сайт *Blender*, размещающий ПО для скачивания и новости о новых функциях.

› **www.blendernation.com** Ежедневные новости от сообщества *Blender* плюс немного бесплатных видеоруководств.

› **www.blenderartists.org** Тусуйтесь или спрашивайте помощи и рекомендаций у других цифровых художников на одном из известнейших форумов *Blender*.

› **www.blendswap.com** Бесплатные активы всех форм и размеров для пользователей *Blender*: от моделей до технических ресурсов.

› **https://cgcookiemarkets.com** Более новый онлайн-рынок, чем Blend Swap, для обмена коммерческими ресурсами.

› **www.blendernetwork.org** Официальный справочник от Blender Foundation и социальная сеть для профессионалов *Blender*.

От Blender до Pixar

Колина Леви [Colin Levy] ПО с открытым кодом привело на работу в Pixar, одну из крупнейших анимационных студий в мире. Начав эксперименты с *Blender* еще в старших классах, к 2006 г. он уже был ассистентом на конференции CG-индустрии SIGGRAPH, где встретил Тона Роозендааля из Blender Foundation.

«Тогда я не очень много общался с Тоном, — говорит Леви. — Но я думаю, он не упускал меня из виду, поскольку три года спустя я получил от него письмо по e-mail с вопросом, не заинтересован ли я в работе над его следующим проектом».

Этим проектом был *Sintel*, третий из «открытых фильмов» — короткометражек, сделанных с использованием только открытого ПО и выпущенных впоследствии под лицензией Creative Commons. Частично в качестве демонстрации ПО, а частично как способ разработать новые инструменты, *Sintel* также самое амбициозное создание Foundation на сегодняшний день — и, с бюджетом в €400000, вторым по стоимости анимационным фильмом за всю голландскую историю.

«Это было абсолютно неожиданно, — говорит Леви, который тогда был на середине своей учебы в Колледже искусств и дизайна Саванны [Savannah College of Art and Design]. — Мне повезло: прием меня на работу мог бы оказаться весьма неумным решением».

Sintel — не классический комедийный сериал: это история молодой женщины, которая приручила детеныша дракона. 15-минутный анимационный фильм исследует то, что Леви называет «довольно глубокими» темами — материнство, потеря и смерть, дружба и расставание.

Это также очень красиво. Хотя это явно работа небольшой команды (в лучшем случае, всего лишь 14 человек), некоторые кадры *Sintel* вполне могли бы занять достойное место в какой-нибудь голливудской картине.

Pixar явно сочла молодого режиссера многообещающим: Леви был принят на стажировку в студию. «На самом деле, это было мое третье заявление о приеме на работу, — говорит он. — В прошлый раз, когда я был на втором курсе колледжа, у меня было самое худшее собеседование в моей жизни. К последнему я уже несколько вырос как личность и художник. Но главным отличием было то, что у меня в активах был *Sintel*, и это определенно был не просто студенческий фильм».

Однако на самом деле Pixar заинтересовал даже не тот факт, что Леви снял *Sintel*. Он также делал всю постановочную работу в фильме: ставил хореографию действий 3D-персонажей на виртуальной площадке и выстраивал снимки, чтобы превратить их в кинематографичную историю. Именно эти его умения выбили ему стажировку — и, в конечном итоге, штатную должность.

За прошедшие 5 лет Леви поработал для студий художником-макетчиком в таких фильмах, как *Университет монстров* [*Monsters University*], и впоследствии стал оператором-постановщиком в *LAVA*, анимационной короткометражной картине, которую Pixar выпустил вместе с *Головоломкой* [*Inside Out*].

Он также продолжает использовать *Blender* в своей собственной работе. В своем текущем побочном проекте — научно-фантастическом проекте с live action, пока без названия — все визуальные эффекты создаются с его помощью. «Я все еще действительно наслаждаюсь работой в *Blender*, — говорит он. — Это сказочный инструмент для создания короткометражных фильмов с десятком или около того художников».

«Я чувствую себя глубоко обязанным Тону, и Foundation, и *Sintel*, — подводит итог Леви. — Не думаю, что я сейчас работал бы в Pixar, не будь *Blender*».

› *Sintel*, третий открытый фильм на *Blender*, обеспечил Колину Леви пропуск в Pixar.

www.linuxformat.ru/subscribe

Записки web-серфиста

Постоянно бороздящий просторы Всемирной сети **Петр Семилетов** невольно поддался обаянию Бледной Луны...

Мне повезло — я застал все вехи развития WWW, того измерения Интернет, что основано на протоколе HTTP. Для многих, впрочем, Интернет и WWW — одно и то же.

В середине 1990-х в Киеве начали проходить первые компьютерные выставки, в помещении Торгово-промышленной палаты. Бесплатный вход, внутри раздают на шару трехдюймовые дискеты с демонстрационными версиями нового софта, молодежь важно берет с каждого стенда распечатку нехитрого каталога товаров какой-нибудь фирмы. Открыв рты, толпа окружает компьютер, где запущен первый *Doom* или впечатлявшая тогда стрелялка с живым видео — *Mad Dog McCree*.

И вот в одном закутке стояло несколько рабочих станций Sun, под управлением системы Solaris. На экране был открыт браузер *Netscape Navigator*. Я сел на высокий стульчик рядом, и впервые в своей жизни зашел в Интернет, на сайт компании Disney (других адресов я тогда не знал, а поисковиков еще не существовало). Мультяшные картинки на экране! Меня это проняло, поразило — сайт физически находится за океаном, а я тут получаю к нему доступ!

Нас теперь удивляют другие вещи; а так рассудить — не столь и много осталось чудес. Новые модели мобильных телефонов уже не вызывают чувства необходимости именно этого предмета, этой марки; они все стали одинаковыми. Никакого чуда — а то ведь было, что ходил с Моторолой C650 и всё из кармана вытаскивал, не мог насмотреться да клавиши потискать вдоволь. Теперь сунул в карман и пошел. Обыденно.

Web-браузеры развивались — из средства просмотра гипертекстовых страниц превратились в игровые приставки, телевизоры и магазины. Возникло новое измерение бытия. Первый в мире браузер *Mosaic* сменился *Netscape Navigator*, появился *Internet Explorer*, *Opera*, *Netscape* породил *Mozilla*, *Mozilla* обратилась в *Firefox*, оставшись-таки *Mozill*'ой. Явился в мир *Chrome*. Но что это? Раньше каждую свежую версию любимого браузера — не скажу какого — ждал с предвосхищением: а что там будет нового? Затем чувство сменилось иным: а что испортят? Чем еще огорчат?

Больше других браузеров я пользовался *Firefox*. Скорее, мне подходил не столько сам браузер, сколько подобранный мною за годы набор дополнений вроде Scrapbook, при помощи которого я люблю сохранять страницы впрок.

Однако с некоторых пор *Firefox* начал меня раздражать. Вот *Firefox* вылетает в связи с определенным использованием плагина Flash. Вот — уже без всякого Flash. Вот разработчики сообщают, что порешили перевести весь честной народ на протокол HTTPS, а при заходе на обычные http-сайты браузер будет сокращать количество поддерживаемых технологий, и так далее. Недавно, когда я нажал кнопку «Назад», будучи на собственном бесхитростном краеведческом сайте, браузер выдал мне грозное предупреждение — мол, соединение-то не защищено!

Куда же деваться? Браузер *Vivaldi* мне нравится, но для основного еще сыроват. *Chromium* мне не нравится, но я использую его там, где *Firefox* вылетает. Старая, привычная версия *Opera* — всегда под рукой, но долго ли она еще будет запускаться?

К тому же я, как многие люди, привык к определенному укладу и набору вещей. Мне нужно каждое утро заводить механические часы «Свет» выпуска начала 1950-х годов, купленные мною по дешевке на барахолке. В правом дополнительном кармашке джинсов определенной фирмы у меня должен лежать складной ножик Victorinox с гладкой темно-красной рукояткой. А если велосипед, то непременно белорусский дорожник «Аист», и другого мне не надо.

То же с браузером. Я не мыслю работы без Scrapbook. Лишенный его, я ощущаю себя, как рыба на воздухе, или как без сотового телефона. Иногда забываю пароли — тоже хорошо бы дополнение, которое делает тайное явным и показывает мой пароль буквами вместо звездочек. Я не параноик, каковыми считают пользователей разработчики браузеров. Я не хочу видеть звездочки или точечки вместо букв. Никто не дышит у меня за плечом, подглядывая пароли. В *Firefox* я нужные мне дополнения подобрал, и переход на другой браузер в качестве основного для меня был немыслим без возможности использования этих же дополнений.

Не помню как, я вышел на сайт браузера *Palemoon* — palemoon.org. Бледная Луна! Логотип соответствующий. Сразу видно, рисовал иностранец. Наша-то Луна выглядит как лицо. Прежде в народе считали, что она — солнце покойников, а лик на ней — Каина, в вечное ему наказание.

Почитал на сайте описание, заинтересовался. Браузер, основанный на *Firefox*. Форк! Разрабатывается Moonchild Productions — это один человек, голландец M. C. Straver, который живет в Швеции и работает под псевдонимом Moonchild, «Дитя Луны». Помимо программирования, он еще и рисует — работы Стравера представлены на сайте www.wolfbeast.com.

В разработку *Palemoon* Стравер ушел с головой, о чем свидетельствует активность на форуме проекта и постоянные обновления браузера. Форум вообще живой, вокруг *Palemoon* за пять лет жизни проекта сплотилось сообщество пользователей.

Не стать ли мне одним из них?

› ***Palemoon*: окно «О программе».**

Сначала я скачал предлагаемую по умолчанию маленькую программу установки. Она, дескать, докачает всё нужное и установит. Не срослось.

Я не стал особо вникать в причины, что же не понравилось этой программе в моем дистрибутиве Mageia 4 (пятый я не ставлю из-за косяков между *X.Org* и драйверами видеокарты от ATI). Однако что-то удержало мое внимание. Я загрузил полный архив с *Palemoon* и, распаковав его, запустил.

И тут снова, как коряво пишут в заметках путешественники, неожиданное препятствие! Каждый раз, когда я щелкал по главному меню, браузер стремительно вылетал. Оказалось, в настройках KDE, в разделе Оформление приложений, Стили и шрифты GTK надобно выбрать что угодно, только не Oxygen-GTK, который стоит по умолчанию. Темы Clearlooks оказалось достаточно, заодно остальные *GTK*-программы стали как-то удобнее, не такими мелкими для глаз.

После этого небольшого шаманства *Palemoon* заработал как часы. Я снабдил его символической ссылкой на Flash-плагин (в каталоге **palemoon/browser** надо сделать симлинк на общесистемный каталог **/lib64/mozilla/plugins**, или какой другой) и проделал действия, при которых *Firefox* взбрыкивал. *Palemoon*, к моему удивлению, не взбрыкивал. Здорово!

С Flash вообще любопытно, кроме вылетов. Как известно, Adobe почти не обновляет этот плагин для Linux, выпуская разве что исправления к версии 11.2. И *Firefox* ругается, цитирую, что «Shockwave Flash является уязвимым и должно быть обновлено» — при этом предлагает включать плагин по запросу. Захóдите на сайт, а вам предлагают включить плагин по запросу. А то уязвится. Тот же самый плагин в *Palemoon* не вызывает у браузера подозрений, и опция «всегда включать» на странице плагинов — доступна, в отличие от *Firefox*, который считает, что лучше вас знает, как надо поступать с вашими плагинами.

Я обрадовался нормальной работе Flash и продолжил знакомство с *Palemoon*. Установил в него »

› Сайт журнала в *Palemoon*.

«тяжелое» дополнение от *Firefox* — Scrapbook. И подключил к последнему хранилище из того Scrapbook, который установлен в *Firefox*.

Всё подхватилось отлично, теперь я могу пользоваться этим дополнением в обоих браузерах, с общей «базой данных», хотя не рискую что-то добавлять в Scrapbook или удалять оттуда, когда оба браузера запущены одновременно. Вдруг это нарушит целостность хранилища?

Тут я понял, что *Palemoon*, кажется, останется со мной надолго. И окунулся в его изучение.

В 2014 г. *Firefox* с выходом версии 29 обрел новый пользовательский интерфейс — Australis, благодаря которому иконки стали меньше и появилось некое ощущение работы с *Chrome* от Google. *Palemoon* же держится старого классического интерфейса *Firefox* версий с 4 по 28, и особых изменений не предвидится. Так хочет разработчик и пользователи. Различия между *Firefox* и *Palemoon* не позволяют, однако, в большинстве случаев предугадать, насколько хорошо дополнения от *Firefox* будут работать в *Palemoon*. Впрочем, на сайте последнего предлагается постоянно обновляемый список совместимости. Например, «тяжелый» TabMix Plus, который предлагает свой движок управления вкладками, вписывается в *Palemoon* гладко. Другое дело — мелкие дополнения, возникшие для возвращения в *Firefox* частей старого интерфейса: они воздействуют на Australis и в *Palemoon* попросту излишни. В нем и так старый интерфейс. Со страницы дополнений *Palemoon* предлагается сразу две ссылки — на сайт Mozilla/*Firefox* и на страницу собственных, заточенных под *Palemoon* дополнений. Там же — темы оформления и языковые пакеты, так что не забудьте скачать русский. Но чтобы пользоваться переведенным интерфейсом, этого почему-то мало.

По умолчанию локаль — английская, ее надо переключить. Для этого идем на страницу по адресу about:config и получаем доступ ко всем настройкам браузера. Находим там параметр intl.locale.matchOS и щелчком мыши переключаем его из false в true. После перезагрузки браузер будет использовать системную локаль. Другой вариант — переопределить параметр general.useragent.locale, задав ему значение ru. Существует также дополнение для быстрого переключения локали — Zing! Locale Switcher.

Поскольку *Firefox* и *Palemoon* похожи, я расскажу, чем *Palemoon* отличен от первого, подразумевая, что остальное работает так же, как в *Firefox*, если не лучше. Стоит поведать и о «родных» для *Palemoon* дополнениях.

Palemoon Commander — рай для любителей настроек, но этот инструмент более полезен для пользователей Windows, для них там предназначена половина опций. Ведь изначально *Palemoon* выпускался только для Windows, это позже появились и стали выходить стабильные сборки под Linux.

Дополнение Palemoon Commander предоставляет также тончайшую настройку работы с сетью, движка JavaScript, кэша, вкладок — словом, всё тайное становится явным, но по большому счету сокрытое отлично работает по умолчанию.

К Palemoon Commander по назначению вплотную примыкает встроенное средство настройки, что прячется в меню Настройки: Настройки панели состояния. Это окно с пятью вкладками параметров для одной только нижней панели состояния! Той самой, от которой отказались в *Firefox*! А в *Palemoon* эта панель — одна из важнейших составляющих интерфейса. В ней отображаются ссылки, когда вы наводите мышь на странице на какой-нибудь переход, а также выводятся разные сообщения и помещаются кнопки управления дополнениями. На нижнюю панель вы и сами можете добавить какие-нибудь полезные кнопки.

Из внешних дополнений поведаю еще о нескольких. AdBlock Latitude — очередной вариант устранителя рекламы AdBlock, который режет и ту рекламу, которую по разным причинам пропускает AdBlock. А вот установка классического Adblock Plus в *Palemoon* с некоторых пор блокируется, по причине несовместимости на уровне используемых средств JavaScript. Эта блокировка летом 2015 г. вызвала на некоторых форумах многостраничные разборки между пользователями *Palemoon* и *Firefox*. Кстати, набор фильтров у AdBlock Latitude и Adblock Plus — с одного и того же сайта.

Дополнение Show My Password отображает пароли буквами. Впрочем, в *Palemoon* и *Firefox* есть штука, которой мало кто пользуется, однако она полезна, если вы забыли пароль. Да и, выходит, конспиративные меры по отображению паролей символами-заменителями — нелепость, ибо достаточно пойти в Настройки > Защита и нажать кнопку Сохраненные пароли. Будет вам окно, где еще одна нелепость — кнопка Отобразить пароли. Казалось бы, я зашел в это окно, чтобы увидеть пароли, а мне дают кнопку. Да еще переспросят при ее нажатии — в самом ли деле я желаю видеть пароли? Нет, я безумно щелкал по всему экрану мышью и пришел сюда случайно! Такая трогательная забота о пользователях раздражает. «Засекретили» пароли, сами дали в руки средство эти пароли «рассекретить», да еще полагают в пользователе личность, крайне неуверенную в своих поступках.

Не могу обойти вниманием тему оформления Past Modern Revisited, с ней *Palemoon* выглядит как классическая старая *Mozilla* версии эдак третьей. Вообще тем оформления — в традиционном понимании — для *Palemoon* существует, пожалуй, больше, чем для *Firefox*. В *Firefox* сейчас почти все темы — это оформление одной-единственной темы. Вы меняете фоновую картинку, цвета — и не более. Полноценные темы, где менялся набор кнопок, вид элементов интерфейса, в субкультуре *Firefox* теперь именуются «полные темы». Их целая уйма — старых — висит на сайте дополнений *Firefox*, однако почти все они не работают с новыми его версиями. А на странице тем *Palemoon* (https://addons.palemoon.org/themes/complete) — полные темы все рабочие, а других, неполных, и нет вовсе!

Запуск *Palemoon* происходит, кажется, чуточку быстрее, чем *Firefox* — впрочем, памяти оба потребляют почти одинаково. У *Palemoon* быстрее работает интерфейс, это особенно ощутимо на больших списках с прокруткой. Тот же Scrapbook, открытый в режиме боковой панели, в *Firefox* прокручивается мучительно медленно, в то время как в *Palemoon* — с обычной для таких дел скоростью. Но при этом при попытке зайти в подменю Scrapbook в главном меню *Palemoon* браузер глубоко задумывается.

Для большинства современных браузеров, кроме таких, как *Dillo* или *Links*, важна скорость работы встроенного движка JavaScript, который запускает дополнения и различные web-приложения. Чем быстрее работает этот движок и чем больше возможностей разных версий JavaScript он обеспечивает, тем лучше для пользователя. Я не удержался и сделал небольшое сравнение производительности движков JavaScript браузеров, включая *Palemoon*, на различных бенчмарках JavaScript. Это ничуть не лучше и не хуже бенчмарков, допустим, графических карт или памяти. Для кого-то бенчмарки кажутся искусственными и мало имеют общего с реальностью, а кто-то доверяет итогам.

Тестирование движка можно выразить голым общим числом (которое зависит не только от скорости движка, но и от скорости процессора), или расписывать подробно: ведь один движок может превосходить другой, скажем, в обработке строк текста, а другой движок быстрее первого в криптографических задачах.

Но поскольку так или иначе эти подробности дают некоторое итоговое число, я буду, для экономии места, пользоваться им. Мол, в определенном тесте, на моем компьютере с процессором Intel(R) Core(TM) i7-4771 @ 3,50 ГГц и видеокартой AMD Radeon HD 7800 этот браузер получил столько очков, а тот — столько-то. Но это не значит, что такое соотношение производительности будет справедливо в каждом случае — в JavaScript-версии *Quake* или других приложениях.

Сравнивать буду 64-битные линуксовые сборки таких браузеров, как *Chromium 45.0.2454.85* Mageia.Org 4, *Opera 12.16* (последняя из «старых» *Opera*, на движке Presto), *Firefox 42*, *Palemoon 25.8* и *Vivaldi 1.0.303.52* (Beta). У *Vivaldi* и *Chromium* движок браузера по сути один — *Chrome*.

Разработчики *Chromium* придумали сравнительный тест (бенчмарк) Octane (http://chromium.github.io/octane/), который показывает следующее — тут «больше» значит «лучше»:

- Chromium: 35860
- Vivaldi: 35097
- Firefox: 31306
- Palemoon: 20055
- Opera 12.16: 9145

Не знаю, должен ли я изображать удивление, пораженно моргая глазами и оглядываясь по сторонам. Вообще Octane основан на запуске реальных JavaScript приложений из разных ниш применения — например, работа с массивами, математическими функциями и битами тестируется

www.linuxformat.ru/subscribe

на примере PDF-читалки из *Firefox*, эмуляция игровых консолей — на эмуляторе GameBoy и так далее. Из суммы результатов этих разрозненных тестов и составляется общее итоговое число. И некоторые тесты существенно на него влияют.

Рассмотрим подробности. В тесте pdf.js — читалка PDF — *Palemoon* получил результат 13398, *Chromium* — 25683. Эмулятор GameBoy: *Palemoon* — 37807, *Chromium* — 85302. Библиотека сжатия данных *zlib*: *Palemoon* — 67948, *Chromium* — 63806. Я не защищаю *Palemoon*, но стоит подобрать другие «кирпичики» в качестве набора тестов, и общий показатель будет другим.

Посмотрим на другие бенчмарки. Старый добрый SunSpider, разработка которого уж прекратилась, а им всё пользуются. Здесь результат показывает время в миллисекундах, затраченное на тест. Чем меньше, тем быстрее работает движок Java, тем лучше!

- Vivaldi: 154
- Chromium: 136
- Firefox: 158
- Palemoon: 127
- Opera: 141

JetStream — «наследник» SunSpider. Разработчики JetStream пошли путем Octane, используя в качестве набора тестов реальные приложения либо их части. Большее значение — лучшее:

- Vivaldi: 205,44
- Chromium: 205,34
- Firefox: 201,06

Opera с треском провалила ряд тестов данного бенчмарка и не получила итогового числв, а *Palemoon* застрял на тесте cdjs (имитация столкновения самолетов).

Напоследок — Kraken JavaScript Benchmark (http://krakenbenchmark.mozilla.org). Результат — в миллисекундах. Чем меньше, тем лучше:

- Vivaldi: 914,2
- Chromium: 899,1
- Firefox: 995,5
- Palemoon: 1294
- Opera: 6089,4

На форуме *Palemoon* Moonchild лично поместил целую статью о том, как отражается работа *Palemoon* в различных бенчмарках, сделав ряд справедливых замечаний, что такие тесты проверяют скорость одних аспектов браузера и оставляют за кадром другие.

Но вот у меня есть приведенные выше числа; что мне с ними делать? По ним выходит, что моя любимая «старая» *Opera* — тормоз, а *Palemoon* плетется тоже в хвосте, и я понимаю, почему разработчик *Palemoon* посвятил защите своего детища целую статью. Обидно ведь, если обходят по числам.

Но когда я запускаю *Palemoon*, ощущаю ли я разницу в скорости при посещении разных сайтов, в сравнении с другими браузерами? Ну вот карты Яндекс в режиме «панорамы» под *Chromium* работают как-то пошустрее, чем в *Palemoon*. Быстрее не то отрисовка, не то подгрузка графики, когда камеру вращаешь. С картами Google та же картина. Однако тут неясно, на что грешить — на отрисовку, которая связана с графической подсистемой, или на производительность движка JavaScript.

Но что-то мне подсказывает: бенчмарки не так уж пристрастны к определенным браузерам, тем более, что, как видно по результатам выше, пристрастия эти примерно одинаковы.

Я бы отнес *Palemoon* к неторопливым, однако надежным браузерам. Да, карты в режиме панорамы — чуть медлят, зато Flash-видео в Youtube или Контакте у меня не вылетает, как в *Firefox*. А так ли тормозит в *Palemoon* графика? Давайте проверим, пресловутыми синтетическими бенчмарками!

Первый на очереди — CanvasMark 2013 (www.kevs3d.co.uk/dev/canvasmark/). Это HTML5-бенчмарк с уклоном в 2D и отрисовку через Canvas. В результатах, больше — лучше:

- Vivaldi: 22829
- Chrome: 22557
- Firefox: 11704
- Palemoon: 11246
- Opera: 10918

Насколько эта закономерность проявится в другом бенчмарке, BMark (www.wirple.com/bmark)? Он проводит два теста отрисовки через Canvas и два теста отрисовки через WebGL. Больше — лучше.

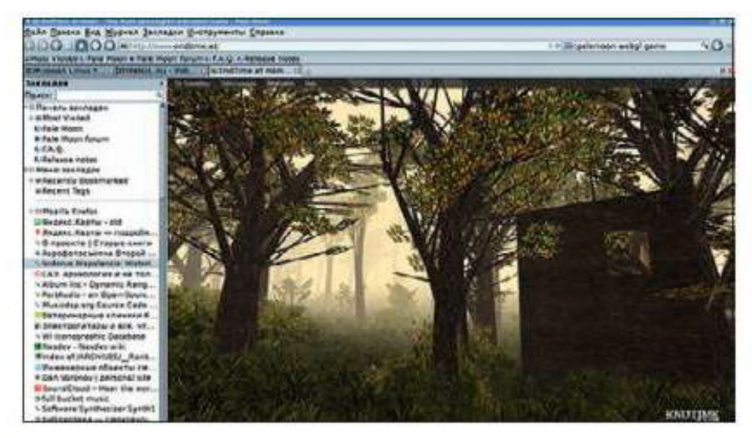
Что же, значительный отрыв браузеров на движке *Chrome* проявился и в этих бенчмарках. «Старая» *Opera* действительно устарела для нагруженных графикой сайтов, и это заметно по тем же Картам Яндекс в режиме панорамы. А браузерам на основе *Firefox* есть повод развивать скорость! Но что ждет их в будущем?

В недрах проекта Mozilla зреют две штуки — язык программирования Rust и разрабатываемый на нем (пока не полностью, частью на C++) движок браузера — движок под названием Servo. На это Servo, когда оно будет готово, и перепрыгнет *Firefox* — иначе зачем разрабатывать новый движок... Однако не новый ли движок на горизонте мешает в полной мере развивать текущий, знаменитый Gecko?

У разработчика *Palemoon* нет нового движка в кармане. У него есть ответвление Gecko, куда время от времени вливается код из того же Gecko. Если (полагаю, вернее — «когда») *Firefox* перейдет на Servo, *Palemoon* останется на Gecko, и еще какое-то время будет запускаться на компьютерах пользователей, которым не нужен современный уровень работы некоторых сайтов — если только Moonchild в одиночку или с группой сподвижников не возьмется за дальнейшее развитие Gecko. Который уже сейчас по скорости отстает от *Chrome*. А что же будет, когда к Gecko совсем потеряют интерес разработчики из Mozilla/*Firefox*?

Однако задолго до этого может случиться неприятность другого плана. И поскольку известно, что в *Firefox* отказались от совместимости со старыми темами оформления, не вижу причин, по которым нельзя отказаться от совместимости с XUL — современным нам нынче API дополнений.

Ведь потихоньку идет введение поддержки WebExtensions — дополнений, написанных под API, используемый в движке *Chrome*. Такая поддержка уже есть в экспериментальных сборках *Firefox* —



HTML5/WebGl: игра *Endtime At Home* идет без тормозов.

Firefox Nightly. Но Moonchild уже объявил, что не будет поддерживать WebExtensions и останется при XUL.

А разработчикам дополнений будет выгодно в будущем, когда *Firefox* обзаведется полноценной поддержкой WebExtensions, перейти на этот API. Разработчикам проще поддерживать единый межбраузерный API, чем несколько таковых. Много ли дополнений от *Firefox* работает в *Seamonkey* — этой ипостаси классической *Mozilla*? Раз-два, и обчелся.

Palemoon с дополнениями от *Firefox* совместим гораздо более. Сейчас в *Palemoon* доступны тысячи современных плагинов от *Firefox*. Но когда *Firefox* «примет» WebExtensions, *Palemoon* останется с этими же тысячами дополнений, более не развиваемых. И хорошо, если *Palemoon* подхватит знамя Gecko, XUL, и снова перетянет часть субкультуры «классического» *Firefox* на себя, как уже случилось в небольшом масштабе — хотя еще жив «подлинник», сам *Firefox*, регулярно подпитывающий *Palemoon*.

И у *Palemoon* будет два варианта будущего. В одном, очень оптимистичном, *Palemoon* полностью станет преемником *Firefox* по духу — как дистрибутив Mageia продолжила дело даже не Mandriva, но ее предшественницы Mandrake. В другом случае *Palemoon* продолжит существовать как устаревающий (относительно развития WWW) браузер.

Palemoon позволит нам определить, насколько были правы разработчики *Opera*. Когда они собирались отказаться от движка Presto для десктоп-версии своего браузера, их просили открыть код Presto, на что был дан ответ: мол, даже если мы откроем код, всё равно он слишком велик и сложен, чтобы его продолжил развивать кто-то посторонний. Код же Gecko открыт, и уже сейчас в какой-то мере (я не сравнивал исходники *Palemoon* и *Firefox*) развивается Moonchild'ом.

Надеюсь, через несколько лет мы вернемся к этой теме и посмотрим, что случилось с *Firefox* и *Palemoon*. А пока я буду пользоваться последним вместо *Firefox*, всё равно производительность примерно одинаковая, при более приятном для меня, более надежном поведении браузера. LXF

Тест	Canvas		WebGL		Всего
Браузер	Тест 1	Тест 2	Тест 1	Тест 2	
Vivaldi	699	1394	590	679	3362
Chromium	681	1206	595	726	3208
Firefox	477	134	137	128	876
Palemoon	331	209	176	159	875
Opera	342	222	49	128	741

По советам м-ра Брауна

Джолион Браун
В свободное от консультаций по Linux/DevOps время **Джолион** обуздывает стартап. Его самая большая амбиция — найти причину пользоваться *Emacs*.

Эзотерическое системное администрирование из таинственных закоулков серверной.

Вспоминая Аду

Тринадцатого октября был День Ады Лавлейс [Ada Lovelace]. Ада — историческая личность, которую я нахожу волнующей. Она не только считается первым в истории программистом, но и именно ей принадлежит концептуальная идея о том, что вычислительные устройства способны не только на обработку чисел. Она прожила интересную (пусть и довольно привилегированную) жизнь — была дочерью лорда Байрона, стала графиней Лавлейс и была знакома с Чарльзом Диккенсом и Майклом Фарадеем.

Но более всего она известна по работе с Чарльзом Бэббиджем. В 1842–43 гг. она перевела с французского на английский запись одной из его лекций о второй теоретической вычислительной машине — Аналитической машине. Она добавила к лекции серию заметок о том, как должна работать машина, и включила метод выполнения конкретных вычислений с помощью Машины (которую, к сожалению, так и не создали). Именно этот метод считается первой компьютерной программой (хотя по этому поводу есть разногласия, как и со всеми историческими событиями). По-моему, это было удивительным достижением, учитывая, за сколько лет до эры компьютеров это произошло.

Так совпало, что в первый раз я столкнулся с Unix в университете (много лет назад), делая упражнения по программированию на языке Ada, названном в ее честь. Подозреваю, что программа Ады была гораздо лучше моих!

Сейчас я планирую посетить местное мероприятие по Ada — его проводит один из моих клиентов, чтобы отметить достижения женщин в науке, технологии, инженерных науках и математике (STEM). Я уже писал в этой колонке о том, что стараюсь помогать женщинам, желающим сделать карьеру в системном администрировании и DevOps — на этом мероприятии я предложу помощь всем, и призываю читателей посетить их местные мероприятия и сделать то же!

jolyon.brown@gmail.com

Microsoft создает себе систему на Linux

Журналисты рвутся проверить, свистел ли на горе рак и не замерз ли ад, а где-то таинственный хакер латает роутеры уязвимости.

Недавно мы наблюдали то, что еще пару лет назад было невозможно и представить — Microsoft в блоге своей платформы Azure сообщила о разработке собственного дистрибутива на базе Linux. Но пока мы не увязли в обсуждении, предпочла ли Microsoft KDE или Gnome, спешим сказать, что в этом случае Linux применяется в качестве основы для так называемого Azure Cloud Switch. Так Microsoft попыталась создать ПО для сетевых устройств.

Благодаря своей гибкости Linux отлично подходит для такого сценария, и его часто можно увидеть в сетевых стойках. Подробностей немного, но это довольно серьезный шаг для Microsoft, которая ранее пыталась впихнуть Windows во все продукты.

› **Даже Microsoft знает, что секрет хорошего облака — это подкладка из Linux.**

ACS вряд ли скоро появится на **LXFDVD**, но это еще один пример того, как гигант из Редмонда постепенно меняет свое мнение о любезной нашим сердцам ОС.

И еще одно необычное событие. Symantec сообщила подробности об обнаруженном вирусе под названием *Linux.Wifatch*. Эта программа ищет домашние роутеры и другие устройства, подключенные к Интернету (например, web-камеры) и, судя по всему, устраняет на найденных устройствах уязвимости. Программа, написанная на Perl, ничуть не прячется, как другие, более опасные программы, оставляя вместо этого отладочные сообщения, достойные серьезного анализа. Среди действий, предпринимаемых *Wifatch* — закрытие открытых портов telnet на домашних роутерах с сообщением для владельца устройства о необходимости обновить прошивку и закрыть открытые порты.

Пока что Symantec не обнаружила вредоносных действий, выполняемых этой программой в Сети, но предупреждает, что ее код использует «лазейки [backdoors]». Хотя намерения данного червя выглядят благородными и возвышенными, всё же не следует забывать, что он внедряется в системы без согласия пользователей, и его таинственному хозяину впоследствии легко будет использовать его с вредоносными целями (или просто-напросто потерять над ним контроль).

Часть 1: Kubernetes

Джолион Браун учит перехватывать контроль над контейнерами, пока те не перехватили контроль над вами, благодаря открытой версии ПО от Google.

На этом уроке я приступлю к рассказу о *Kubernetes*. Начнем с нескольких ключевых концепций *Kubernetes*, а через месяц рассмотрим построение инфраструктуры с его помощью. *Kubernetes*, с тех пор как Google открыл его исходный код, стал очень популярным проектом. Он автоматизирует операции с контейнерами, такие как развертывание и масштабирование, и запускается на кластере хостов. С текущими трендами в отношении контейнеров, быстрым развертыванием и архитектурой микросервисов *Kubernetes* определенно стоит знакомства и изучения.

На мой взгляд, в последние год-полтора весь мир помешался на контейнерх. Здесь, на священных страницах *Linux Format*, я писал о них лишь пару раз (побоявшись вызвать у вас «контейнерную усталость»). Но снаружи почти у всех, с кем я работал, есть либо довольно преуспевший проект оценки контейнеров, либо планы такого проекта. В большинстве случаев это означает работу с Docker, и вокруг этого проекта, а также в более широком пространстве контейнеров сложилась здоровая экосистема. Отчасти на нее влияет постоянно развивающаяся идея «облака» и ожидаемая роль контейнеров в ней. «В этих облаках есть деньги», может выразиться кто-то. Так как многие читатели данной рубрики несомненно сталкиваются с контейнерами, думаю, пора уделять им внимание более регулярно.

С любой технологией инфраструктуры одна из главных причин беспокойства системных администраторов — в том, как они будут ею управлять. Никто не хочет целыми днями сражаться со своими системами, исправляя ошибки вручную. От физических систем до виртуальных машин и контейнеров — количество дискретных элементов, которые нужно поддерживать (и управлять ими), растет. Естественно, стали появляться решения для контроля над этой ситуацией. К тому же, контейнеры классные! Все хотят впечатлить своего босса/клиента/другое важное лицо, представляя автоматически масштабируемую, эластичную и избыточную службу на контейнерах.

Возьмем Google. С точки зрения инфраструктуры, глобальные вычислительные ресурсы «огромного масштаба», созданные интернет-гигантом, получили почти мистический статус. Во многом это обязано скрытному поведению компании, когда дело доходит до обсуждения ее внутренней архитектуры (когда единственное, что достается любопытным — проблески научной документации). С ростом своих систем Google начала решать проблемы, широкомасштабно запуская и управляя виртуальными машинами. Компания решила пересмотреть проблему и гарантировать, что сбои отдельных компонентов не вызовут сбои в других областях, что они могут быть распределены по разным вычислительным ресурсам и эти сбои можно обрабатывать автоматически. Внутренний инструмент оркестрации Google назывался *Borg* (и есть еще один, под названием *Omega*). *Kubernetes* — потомок *Borg*, пакетированный для внешнего и открытого потребления.

› На сайте есть несколько полезных маркетинговых текстов, которыми вы сможете впечатлить своего босса, рассказывая о проекте *Kubernetes*.

Группы контейнеров и узлы

Сейчас *Kubernetes* поддерживает контейнеры Docker и Rocket (из CoreOS); поддержка других обещается в ближайшем будущем. Эти контейнеры запускаются в коллекциях, называемых «группами контейнеров» и представляющих собой строительные блоки для *Kubernetes*. Группа может состоять из одного или нескольких контейнеров. Если группа состоит из нескольких контейнеров, все они будут расположены на одном и том же физическом хосте. *Kubernetes* может разворачивать несколько копий групп (на основе одной и той же конфигурации) и разворачивать замены для групп, уходящих в автономный режим. Именно на уровне групп *Kubernetes* выполняет свои задачи по планированию и оркестрации. Этот отличается, например, от чистой установки Docker, где кирпичиком является сам контейнер. Группе контейнеров назначается один IP-адрес вместо назначения адреса каждому контейнеру, с номерами портов, которые используются для связи между приложениями в группе. Группа обычно состоит из одного полного экземпляра некого микросервиса. *Kubernetes* предоставляет обнаружение служб, поэтому IP-адреса и DNS можно использовать для указания на набор таких сервисов. Группа также может содержать том, видимый всем контейнерам в группе, позволяя обмениваться вре́менными (эфемерными) данными. Для более постоянных данных необходимо использовать ресурсы монтирования NFS.

Узел — это физический компьютер, где находятся одна или несколько групп. Некоторые узлы являются основными, на них работает ПО управления. К ним относится *etcd* — которая может быть вам знакома по проекту CoreOS, о нем мы писали в одной из более ранних статей «Советов». Это хранилище пар «ключ–значение» для общей конфигурации и обнаружения сервисов. Сервер API обрабатывает вызовы, которые все компоненты *Kubernetes* совершают и получают в процессе работы кластера. Планировщик »

Сборка и запуск приложений в Kubernetes

Надеюсь, что сказанное дальше не слишком шокирует вас, но стоит потратить некоторое время на объяснение, как приложения взаимодействуют друг с другом в облачной инфраструктуре вроде *Kubernetes*. Особенно в такой, как *Kubernetes*, учитывая, что в ней мы управляем контейнерами.

К этим сущностям не прилагается большого довеска в виде виртуальной машины, но они и не настолько эластичны. Приложениям, работающим в контейнерах, надо учитывать и планировать элементы инфраструктуры, которые появляются и исчезают рядом с ними. Для статического сайта это не имеет большого значения, но для любого приложения, которое должно хоть как-то сохранять состояние, это чрезвычайно важно. Это состояние нельзя сохранять в самом контейнере энергонезависимым образом, так как сами контейнеры энергозависимы. Данные, требующие сохранения, должны быть записаны в какое-то хранилище данных — это может быть обычная база данных (или облачная версия, такая как RDS Amazon) или хранилище разделяемых данных, такое как Redis, о котором я рассказывал ранее. С такой архитектурой для масштабирования «вверх» достаточно запустить несколько новых копий сервиса. По крайней мере, в теории: у реального мира есть привычка разрушать утопии архитекторов инфраструктуры. Тем не менее, эти идеи принесли нам крупнейшие системы в мире (если Google снова не опередил всех и уже не делает в своих дата-центрах что-то другое).

и контроллер работают с группами обычных узлов: они проверяют, что запущено правильное количество групп, что новые контейнеры размещаются на узлах, имеющих достаточно емкости для них, и т.д. На обычных узлах запускается т.н. *kubelet* (обеспечивающий работу самих групп) и *kube-proxy*, представляющий собой (сюрприз!) простой сетевой прокси-сервер и балансировщик нагрузки, который передает трафик сервисам, запущенным в группах. Все это можно объединить на схеме (см. внизу).

Управляем Kubernetes: kubectl

Простейший способ взаимодействия с *Kubernetes* — через интерфейс команды *kubectl*. С ней можно создавать простые автономные группы однострочной командой (Этот пример взят из документации *Kubernetes*, которую можно найти на http://kubernetes.io).

```
$ kubectl run my-nginx --image=nginx --replicas=2 --port=80
CONTROLLER CONTAINER(S) IMAGE(S) SELECTOR REPLICAS
my-nginx my-nginx nginx run=my-nginx 2
```

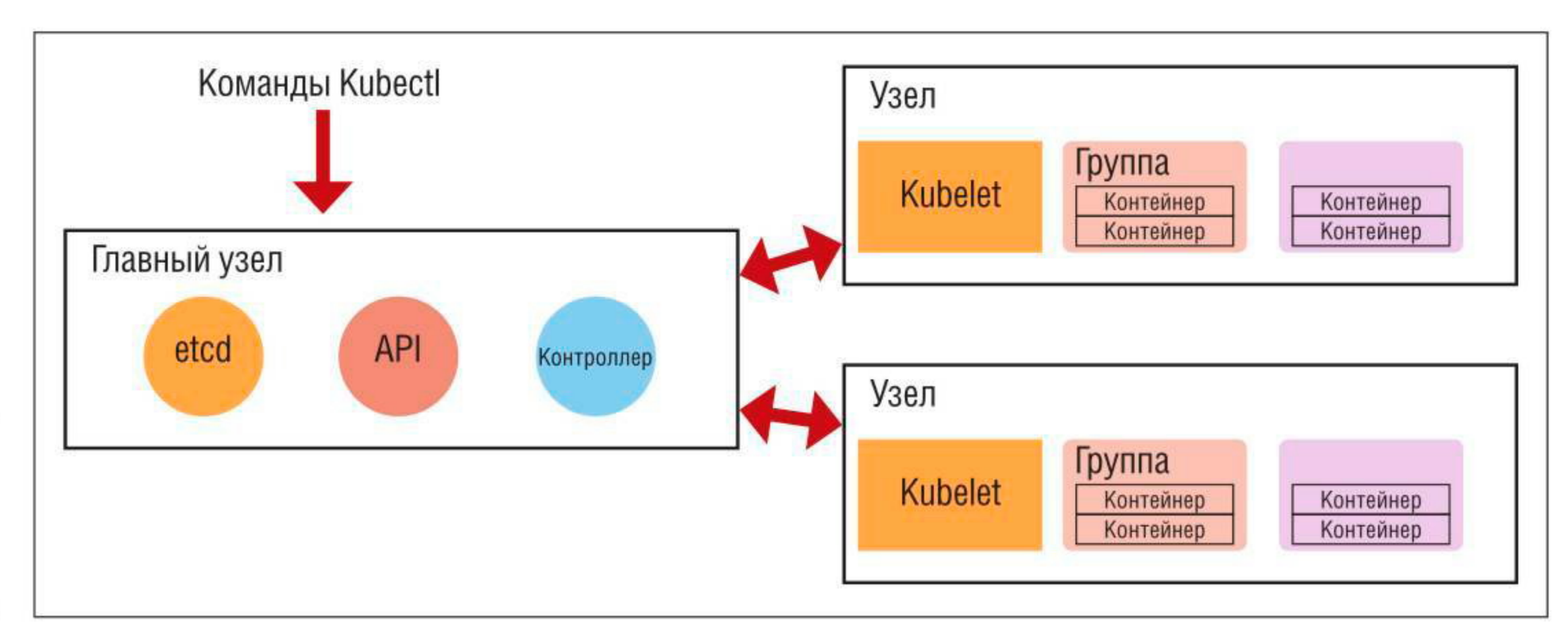
Но для нетривиальных конфигураций удобнее создать файлы YAML или JSON, которые *kubectl* прочтет и будет ими руководствоваться. У них, разумеется, есть и дополнительное преимущество — их можно добавить в систему управления версиями и хранить в репозитории кода (что всегда хорошо). Эти файлы определений сами по себе могут быть довольно просты, но также могут включать проверки работоспособности службы (которая поможет *Kubernetes* решить, находится ли контейнер в рабочем состоянии) и пользовательские поля «имя/значения», называемые метками или аннотациями.

Метки удобно идентифицируют данные в контейнерах для администратора кластера. Обычно это отдельные слова, например, environment: prod или environment: dev. Аннотации обычно длиннее — например, это информация, связанная с загрузкой кода в Git, имя автора конкретной службы или URL-адреса соответствующей документации. Метки, в частности, используются для сортировки и поиска сервисов, и лучше всего пользоваться ими с самого начала.

Контроллеры репликации создаются в *Kubernetes* так же, как и обычные группы, с добавлением ключевых слов типа replicas: 2. В этом случае *Kubernetes* запустит два экземпляра запрошенной группы и восполнит это количество, если они «упадут», будут закрыты и т.д.

Kubernetes работает с сетью немного иначе, чем обычная установка *Docker*. Администратору незачем беспокоиться о выделении портов для обхода приватной сети, которая разрешает взаимодействовать контейнерам только с другими контейнерами того же узла. Все группы могут взаимодействовать друг с другом, даже минуя узлы. Но после перезапуска выделенные IP-адреса изменяются. Требуется определить сервисы (опять же, с помощью конфигурационных файлов), которые автоматически обработают эту ситуацию для администратора. Легко настроить сервис, который работает с портами, помеченными как порты web-серверов *Apache*, и перенаправляет их трафик на порты 80 или 443. По мере того, как группы проходят по своему жизненному циклу, конечная точка сервиса будет обновляться новыми IP-адресами. Чтобы внешние клиенты (например, интернет-клиенты) могли обращаться к сервисам кластера *Kubernetes*, они должны иметь публичные IP-адреса и быть подключенными к тому, что в *Kubernetes* называется NodePorts [Порты узлов] или LoadBalancers [Балансировщики нагрузки]. Эти два похожих метода управляют открытыми Интернету IP-адресами, которые может предоставлить, например, соответствующий облачный провайдер. Другой популярный метод — запустить *HAProxy* (это одна из моих самых любимых открытых программ).

Уфф! Надеюсь, информации было не слишком много сразу. В *Kubernetes* есть и многое другое. Это по-настоящему мощная система с массой практических возможностей. На следующем уроке мы установим *Kubernetes* и перейдем к некоторым рабочим примерам, включая мониторинг и репликацию.



› Упрощенное представление архитектуры *Kubernetes*. Можно видеть, что основным «кирпичиком» архитектуры является группа.

Канарейки и развертывания

«Канарейное развертывание» — это идея, согласно которой приложение можно развернуть рядом с работающим сервисом, и оно может получать рабочий трафик. Если кто не в курсе, отсылка к канарейке — это старая практика, по которой клетки с птицами помещались в шахты, чтобы предупреждать о появлении ядовитого газа. Если птица или новая версия приложения перевернулась лапками кверху, то шахтер или команда разработчиков поймут — что-то не так, и начнут выбираться наружу.

С помощью меток (см. основную статью) легко развернуть и различать различные версии приложения и так же легко вывести их из рабочего режима (в такой ситуации подойдут метки 'version:1.0' и 'version:1.1').

Также удобно, если *kubectl* для *Kubernetes* автоматически развернет приложение до новой версии. При задании соответствующего параметра начнется замена групп (по одной), запускающих старую версию приложения, группами с обновленной версией. Этот процесс можно наблюдать и (что еще более впечатляет) прерывать и откатываться к предыдущей версии одной командой. Это позволяет обновлять сервисы незаметно для пользователя (в случае, если эта функция поддерживается приложением корректно).

Всем, кто прочел слова «восстановить с ленты» в разделе откатывания плана развертывания, можно в этот момент простить навернувшиеся на глаза слезы.

Двухфакторная аутентификация

Вы не параноик — они на самом деле намерены добраться до вас. Улучшите свою защиту, воспользовавшись *Google Authenticator* с SSH.

Думали ли вы, сколько раз в день вы набираете слово “ssh”? Это, несомненно, одна из самых полезных команд в арсенале администратора Linux, с ее встроенным шифрованием (наличие которого я уже начал считать в порядке вещей), а также козырями перенаправления и туннелирования в рукаве. На самом деле, чаще, чем я, SSH пользуются только взломщики систем, автоматические боты и черви, регулярно пытающиеся проникать в системы.

Существует ряд хорошо известных рекомендаций для защиты SSH в Linux. Отключение входа в систему с правами root, использование ключей вместо паролей и (если возможно) сужение доступа к порту SSH (22) через *iptables* или нечто вроде ограниченным набором адресов. Лично я не хлопочу об изменении порта SSH на другой, поскольку утилиты сканирования все равно его найдут, но знаю, что многие администраторы так делают. Я предпочитаю установить *Denyhosts* — пакет, который тупо блокирует мнократные попытки подключения.

Для систем, к которым возможно подключение при поездках (я теперь беспрерывно использую телефон с 4G), определение адресов источников для брандмауэра может быть непростым, даже с динамическим DNS. Проще, на мой взгляд, включить двухфакторную аутентификацию через свободно доступный сервис и приложение *Google Authenticator*. Такой сервис предоставляется и другими провайдерами — взгляните на них, если не доверяете Google — но по моему опыту, предложение Google довольно просто в настройке. Простейший вариант его использования — установить приложение *Google Authenticator* на свой смартфон (это позволяет очень просто сосканировать QR-код). Для его настройки надо выполнить всего несколько команд и изменить несколько параметров. Примеры ниже соответствуют моим системам на базе Ubuntu; в других системах больших отличий быть не должно.

Первым делом установим сам пакет (следуя обычной процедуре apt-get update):

```
$ sudo apt-get install libpam-google-authenticator
```

Затем я запускаю утилиту конфигурацию как обычный пользователя (я никогда не подключаюсь по SSH напрямую от имени root, эта возможность дезактивирована):

```
$ google-authenticator
```

Это порождает несколько вопросов, на которые в каждом случае я отвечаю «да». Они ограничивают повторное использование маркеров; привязывают маркеры ко времени; ограничивают частоту попыток входа и определяют время действия маркеров для решения проблем с временной синхронизацией.

Во время этого процесса будет показан QR-код — просто сфотографируйте его приложением Google, и приложение добавит его в качестве варианта проверки и начнет генерировать коды, которые будут использоваться при входе в систему. Установщик заодно выведет некоторую информацию, которую надо сохранить в безопасном месте — такую как ключ безопасности, код проверки и некоторые «аварийные одноразовые коды», которые используются в непредвиденных ситуациях.

По окончании установки надо изменить несколько параметров в файлах **/etc/ssh/sshd_config** и **/etc/pam.d/sshd**. Убедитесь, что в **sshd_config** имеется следующий набор параметров:

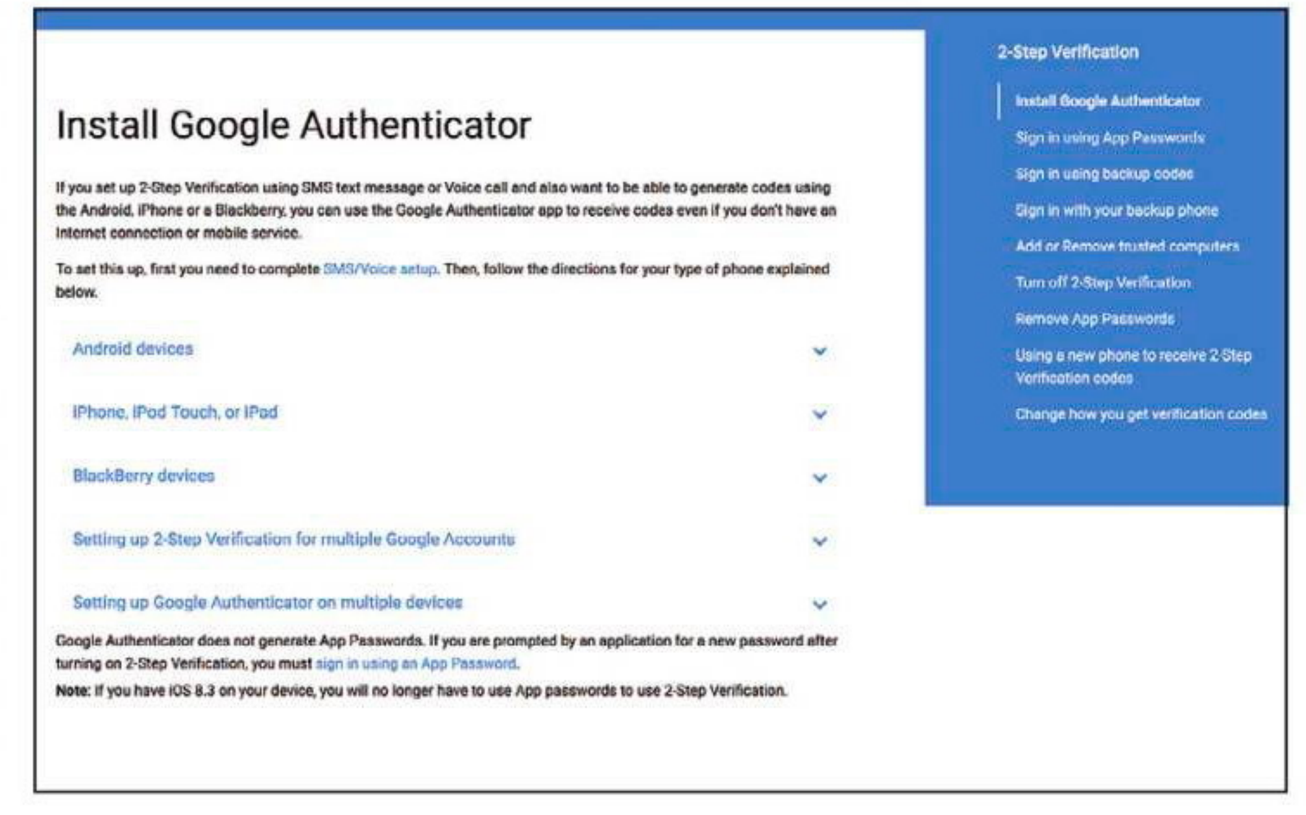
```
# Установить в yes, чтобы включить пароли типа запрос-ответ
# (возможны проблемы с некоторыми модулями и потоками PAM)
ChallengeResponseAuthentication yes
UsePAM yes
AuthenticationMethods publickey,keyboard-interactive
# Установить в no, чтобы отключить текстовые туннельные пароли
PasswordAuthentication no
```

Затем откройте файл **/etc/pam.d/sshd** и убедитесь, что его «шапка» выглядит так:

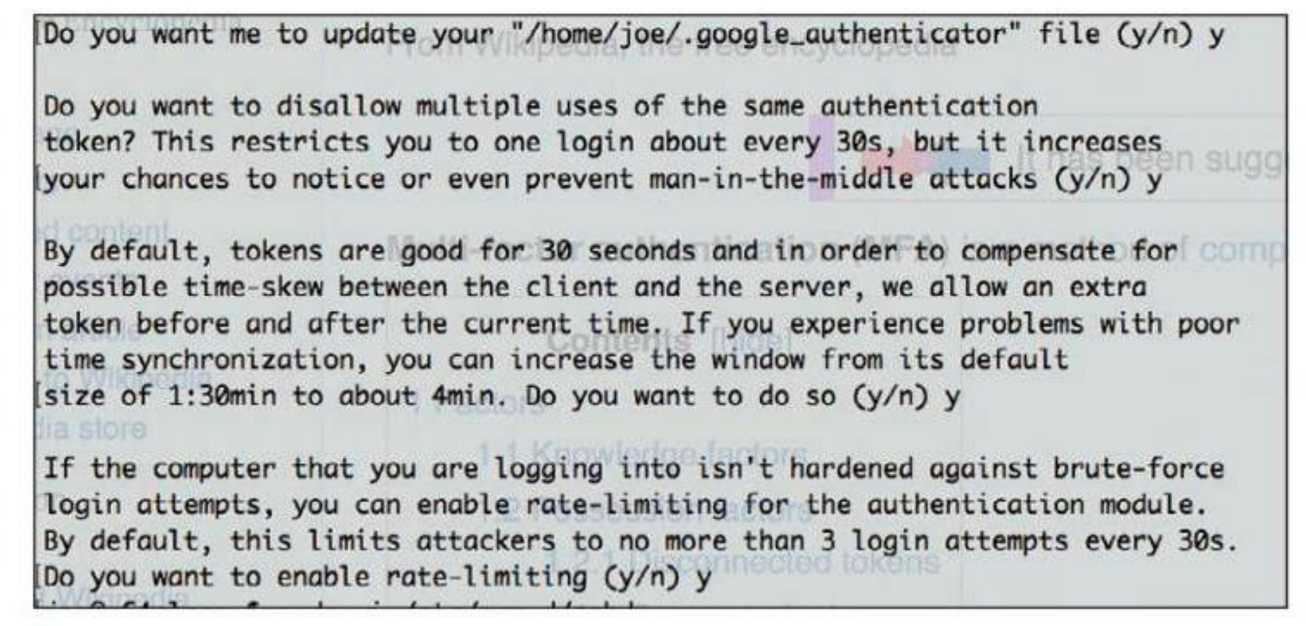
```
# Конфигурация PAM для сервиса Secure Shell
auth required pam_google_authenticator.so
# Стандартная аутентификация Un*x.
#@include common-auth
```

Теперь перезапустите демон SSH. Проверьте, что приложение *Google Authenticator* генерирует коды для вашей системы (код будет снабжен именем пользователя и именем хоста, на котором он создан).

При следующем входе в систему у вас запросят этот код проверки, а затем будет выполнена проверка аутентификации открытого ключа. Это дает вам еще один уровень защиты от злоумышленников, причем такой, который настраивается за несколько минут — дело вполне стоящее. LXF



› На странице *Google Authenticator* есть все подробности об использовании аварийных кодов и руководства для различных смартфонов.



› Каждый знает, что ключ к высокому уровню безопасности — ответить «да ['y']» на все вопросы установщика.

0+

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!

www.hh.ru

hh.ru HeadHunter

*Данная услуга является платной. Подробнее о ее стоимости и условиях предоставления вы можете узнать по адресу hh.ru/applicant/services

LINUX FORMAT Учебники

Наши эксперты помогут вам с любым приложением Linux!

ЕВГЕНИЙ БАЛДИН
Подтвердивший свою квалификацию физик.

Компьютерные удобства

Бабочка: Но ведь есть великие дела, за которые стоит отдать жизнь.
Ринсвинд: Нет, нет таких дел! Потому что жизнь у тебя только одна, а великих дел как собак нерезаных! Единственным критерием истины является опыт.
«Интересные времена». Терри Пратчетт

GNU/Linux можно установить фактически на любой кофемолке с элементами интеллекта. Процесс установки бывает весьма захватывающим, но если интересует результат, а не процесс? Выбор железа, гарантированно предоставляющего нужную функциональность при привычном ПО, даже если стоит оно дороже, чем хочется, и предоставляет меньше дополнительных функций, чем у альтернатив.

Среди приземленного большинства принято посмеиваться над инициативами FSF насчет составления «Путеводителей по технико-этическому выбору подарков [Ethical Tech Giving Guide]». Да, жертвовать функциональностью и дополнительными ресурсами ради максимума открытости железа не всегда оправданно. Но после долгой мороки, скажем, с HP Microserver, где для снижения частоты вращения вентилятора без помощи паяльника надо ставить закрытый драйвер к закрытому RAID-контроллеру, выпущенному только к старой версии Ubuntu, осознаешь ценность доступного железа и рекомендаций FSF. Следовать им вы не обязаны, но присмотреться к ним полезно и сейчас; а в будущем это может гарантировать и выгоду.

E.m.Baldin@inp.nsk.su

В этом месяце вы научитесь...

Выбирать среду 58
Не без помощи **Ника Пирса** определим рабочую среду своей мечты, которой и наличный компьютер тоже обрадуется.

Украшать стол 62
Нейл Ботвик учит делать рабочий стол уникальным и презентабельным, размещая на нем всяческую информацию о системе.

Цеплять принтеры 64
Да не только принтеры, а еще и сканеры — **Ник Пирс** в этом поднаторел и охотно делится ценными знаниями.

Оперировать файлами 68
От **Дмитрия Пантелеичева** вы узнаете, как перемещать туда-сюда файлы и каталоги и как на них ссылаться.

Двигать науку 72
Алексу Кэмпбеллу показалось неправильным оставлять свой ПК в праздности: пусть лучше поспособствует великим научным открытиям!

Ставить Ubuntu 74
Положим, Ubuntu ставить нам уже приходилось, но на сей раз **Нейл Мор** вознамерился водрузить ее на хромбук. Зачем ограничивать функции чисто коммуникаторскими?

Управлять IT-сервисами 76
Сергей Коклеев и **Юрий Малолетко** автоматизируют процессы управления разработками — в частности, управление инцидентами.

АКАДЕМИЯ КОДИНГА

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

Программируем сеть 80
Д-р **Крис Браун** отвешивает свой прощальный поклон, рассказывая про сетевое программирование с использованием UDP.

Создаем интерфейсы 84
Эстетическим воззрениям **Джонни Бидвелла** претят разношерстные интерфейсы, и он их причесывает с помощью *GTK* и *Glade*.

Рабочие столы: Ищем идеал

Ник Пирс тестирует семь окружений рабочего стола, выявляя, какой из них является идеальной средой для ваших нужд.

Наш эксперт

Ник Пирс имеет более чем 20-летний писательский опыт для различных платформ и балуется Linux добрую часть десятилетия.

Среда рабочего стола — важнейший аспект вашего опыта пользователя Linux, как наиболее удобный способ взаимодействия с ПК. В отличие от Windows или Mac, Linux не приковывает вас к единственному рабочему столу, и, как мы выяснили в прошлом месяце, переключать их невероятно просто. Если у вас трудности с рабочим столом по умолчанию в вашем дистрибутиве, то пришло время перемен. Но вместо того, чтобы просто экспериментировать с различными рабочими столами, пробуя их все, пока не найдете подходящий, стоит составить представление о том, чего конкретно вы ожидаете от рабочего стола, затем сузить выбор до двух или трех альтернатив и, наконец, остановиться на одном (а то и двух) — наиболее предпочтительном.

Для данного урока мы подобрали семь самых популярных рабочих окружений, выделив хорошее, плохое и неприятное в каждом. Но прежде чем начать: а на что надо смотреть при оценке рабочих столов? Конечно, первое впечатление — выглядит ли рабочий стол привлекательно (или, по крайней мере, не отталкивающе)? Помните, что большинство окружений можно настроить и оформить на очень разные виды, и если вам нравится внешность текущего рабочего стола, весьма вероятно, что другой рабочий стол можно настроить (или даже установить особую версию), сохранив этот привычный вид.

Функциональность, очевидно, также является ключевым критерием: какие функции предлагаются и самим рабочим столом, и его базовыми приложениями, такими как файловый менеджер или текстовый редактор? Удобство — насколько просто его использовать? Логично ли расположены элементы, на ваш взгляд? Не придется ли делать больше щелчков, чтобы получить доступ к ключевым возможностям системы? Также важно подумать о гибкости рабочего окружения — какие аспекты могут быть настроены на ваш вкус, и легко ли это сделать? Последнее, но не по значимости — вопрос эффективности. Насколько отзывчивой и быстрой будет замена рабочего стола при вашей конфигурации?

При оценке рабочих столов мы задали себе все эти и другие вопросы, но в конечном итоге это является делом вашего личного вкуса и потребностей.

› "Heads up Display" Unity — это удобная, экономящая время функция рабочего стола по умолчанию в Ubuntu.

Unity

https://unity.ubuntu.com

Unity — это собственный эксклюзивный рабочий стол Ubuntu, и для многих — исходный эталон сравнения с другими окружениями. Unity стоит особняком — нравится вам это или нет — несмотря на внешнее сходство с Gnome 3. Он не столь минималистичен, как новый Gnome, но все же относительно скуден — просто пускач-launcher, закрепленный в левой стороне экрана, и рабочая панель вверху.

Одной из сильных сторон Unity является Dash, инструмент для быстрого поиска буквально всего — от мультимедиа и файлов до программ и web-ссылок; он также создает удобные ярлыки для часто используемых элементов. Еще есть HUD — нажмите Alt, чтобы вызвать его. Он обеспечивает альтернативу меню, позволяя вводить, что вы хотите сделать, например, сохранить или распечатать. По мере ввода всплывают предложения, и полностью поддерживается нечеткий (интеллектуальный) поиск.

Имеются и недостатки. Во-первых, launcher нельзя переместить, хотя и можно скрыть. Кнопки окна размещаются в верхнем левом, а не верхнем правом углу окна. Система уведомлений тоже плохо реализована и, как и многое в Unity, может быть изменена только с помощью сторонних средств.

Unity — хороший выбор для тех, кто не поленится изучить его, и будет особенно привлекателен для тех, кто экспериментирует с Ubuntu на небольших дисплеях — благодаря удобному сосредоточению функций; но если вы попытаетесь его адаптировать или разочаруетесь отсутствием конфигурирования, не паникуйте. Альтернатив хватает.

Gnome

www.gnome.org

Дни доминирования Gnome на настольных ПК с Linux прошли — отчасти из-за отталкивающего эффекта минималистичной оболочки Gnome 3. Она задумана, чтобы сосредоточить вас на выполняемой задаче, устранив потенциально отвлекающие элементы рабочего стола, но те, кто привык к навигации через серию панелей, меню и прочим атрибутам, возможно, не захотят такое видеть или в крайнем случае вернут оболочку Gnome Classic, обеспечивающую рабочему столу больше общего с Gnome 2.

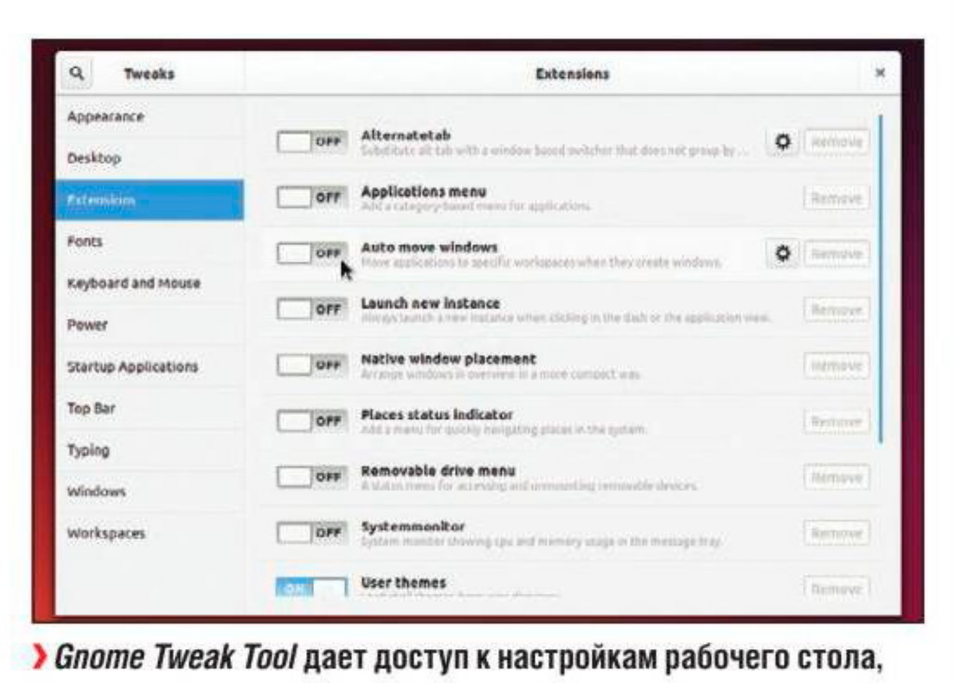
Попав на рабочий стол Gnome, вы увидите почти пустую верхнюю панель — нажмите Activities, чтобы вызвать launcher с ярлыками всех приложений в системе и полем поиска. Освоение не займет у вас много времени — в конце концов, одной из сильных сторон Gnome является его удобство. С другой стороны, ему не хватает некоторых распространенных функций; стремление к минимализму привело к потере ключевых функций основных приложений, таких как двухпанельный вид файлового менеджера для удобной переброски файлов.

Гибкие рабочие столы

Одним из ключевых критериев в наших тестах было определение настраиваемости рабочих столов. По этой части KDE Plasma наиболее гибок: почти каждый элемент можно настроить прямо с рабочего стола. Cinnamon также легко настроить с помощью средств управления рабочим столом, а также его «специй», «запеченных» на рабочем столе.

Большинство рабочих столов предлагают инструменты базовой настройки, доступные через меню настроек. Потратьте время, чтобы изучить, как быстро настроить вид рабочего стола: от обоев до фона окон. Обычно панели настраиваются щелчком правой кнопкой мыши — сразу видно, насколько они настраиваемы.

Изучив эти основы, вы нередко обнаружите широкий спектр плагинов, тем и дополнений, доступных в Интернете: например, пользователи Xfce должны посмотреть Xfce Goodies (http://goodies.xfce.org). Другие окружения рабочего стола, такие как Unity и Gnome, можно в дальнейшем изменить с помощью сторонних утилит. Пользователи Unity должны установить *Unity Tweak Tool*, а *Gnome Tweak Tool* настраивает Gnome. Настраиваемость Mate более ограничена, но можно разблокировать несколько опций, установив *Mate Tweak* из репозитория **ubuntu-mate-dev/trusty-mate** (в Ubuntu Mate этот репозиторий включен по умолчанию).



› *Gnome Tweak Tool* дает доступ к настройкам рабочего стола, которые обычно скрыты.

› Минималистский подход Gnome кое-что прячет — его лучшие функции скрыты за кнопкой Activity.

› Cinnamon дает хорошее сочетание функциональности и производительности, упакованное в более традиционный рабочий стол.

Gnome более настраиваемый, чем Unity, хотя опять же вам понадобятся сторонние приложения, чтобы добиться от него максимальной отдачи. Он тоже поддерживает расширения оболочки, но их можно установить только через браузер, и они часто ломаются при крупных обновлениях Gnome; другие рабочие столы поддерживают такие дополнения гораздо лучше. Тем, кто не проводит много времени на рабочем столе, сосредотачиваясь на приложениях, Gnome подойдет, но если нужен более традиционный интерфейс, попробуйте лучше Gnome Classic. Или читайте дальше...

Cinnamon

http://cinnamon.linuxmint.com

Cinnamon — официальный рабочий стол для Linux Mint; это ответвление Gnome 3 с возвратом акцента на пользователя настольного ПК. Он использует базовые технологии Gnome, включая ветки его основных приложений, и гарантируется, что они остаются более функциональны, чем в самом Gnome; элегантный рабочий стол, легко узнаваемый, особенно для пользователей, пришедших с Windows, с кнопкой меню, ярлыками приложений и системным лотком, упакованными в панель в низу экрана.

Он также легко настраивается посредством ряда «специй»: темы, расширения, апплеты и плавающие десклеты (аналог виджетов KDE в Cinnamon), которые управляются непосредственно через собственный инструмент System Settings [Параметры системы].

Хотя Cinnamon добился больших успехов в повышении производительности (например, последний выпуск использует механизм предзагрузки для быстрого старта после запуска), его зависимость от Gnome 3 означает, что он еще относительно «тяжел»; притом он приобрел репутацию слегка глючного, хотя последние релизы улучшили его надежность.

Хотя, на наши деньги, если вы ищете элегантный, мгновенно доступный рабочий стол, который берет лучшие черты Gnome и представляет их в традиционном виде, Cinnamon — рабочий стол, достойный ознакомления.

Mate

http://mate-desktop.org

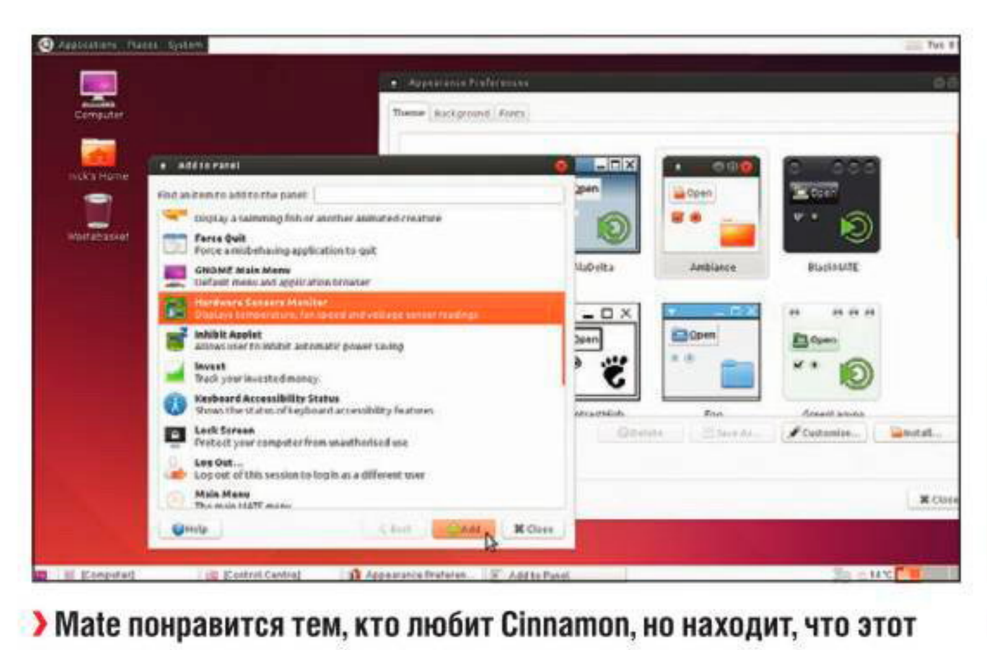
Если вам нравится идея традиционного рабочего стола, но надо, чтобы он работал на слабых или старых машинах, в качестве возможной альтернативы Cinnamon рассмотрите Mate. Это тоже ответвление Gnome, но в данном случае основанное на старшем релизе Gnome 2. Cinnamon поможет снизить потребность в ресурсах, тем не менее предоставив широкий выбор установок, включая базовую — без сверхизбыточности, тормозящей вашу машину, но все основные функции по-прежнему имеются (в том числе *Caja* — двухпанельный файловый менеджер, ответвленный от *Nautilus* из Gnome).

Mate имеет две панели — верхнюю и нижнюю, и можно добавить еще, плюс разместить их по сторонам экрана. По умолчанию в нижней панели отображаются открытые окна, а ряд меню в верхнем левом углу обеспечивает удобные ярлыки для ключевых частей вашей системы. Панели могут также быть расширены рядом »

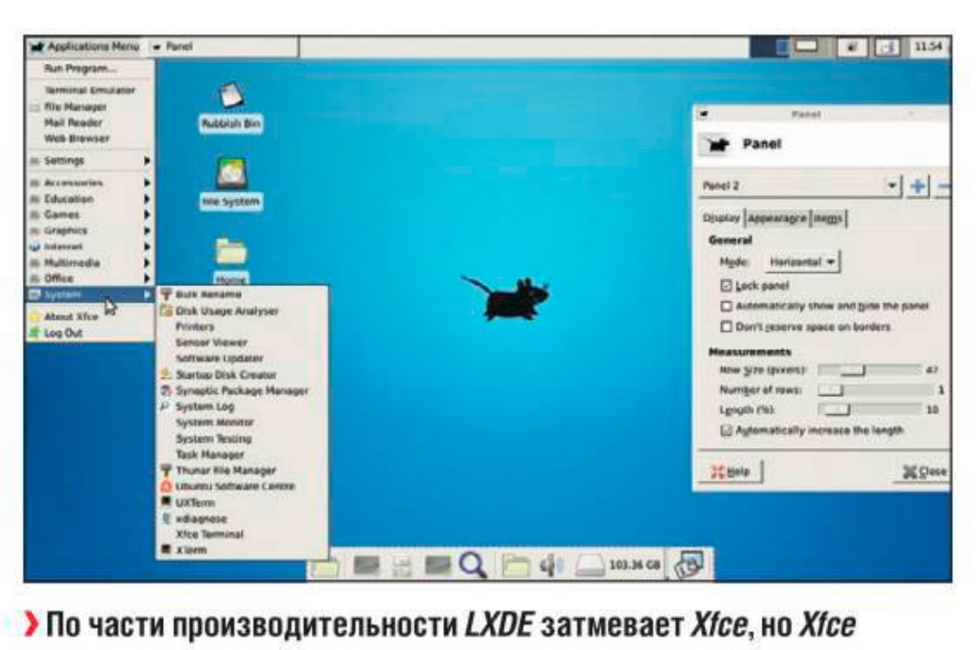
Скорая помощь

Можно установить несколько рабочих столов бок о бок, чтобы сравнить их — закончив с одним, просто выйдите. Нажмите на значок рядом с именем пользователя и выберите другой рабочий стол, потом зайдите в него, чтобы протестировать.

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



› Mate понравится тем, кто любит Cinnamon, но находит, что этот второй слишком алчный.



› По части производительности *LXDE* затмевает *Xfce*, но *Xfce* выделяется некоторыми грамотными функциями.

дополнительных апплетов, например, запуска задач, кнопкой питания и погоды и т. д.

Удобный *Центр управления* дает доступ к большинству системных настроек, в том числе нескольким легко настраиваемым частям самого Mate (см в разделе Appearance [Внешний вид] опции переключения и настройки темы). Для максимальной адаптации пользователи Ubuntu могут даже захотеть попробовать Ubuntu Mate — специально созданную версию, предназначенную для более плавной интеграции с Ubuntu.

В конечном счете Mate предлагает разумный компромисс между настраиваемостью Cinnamon и подходом без излишеств более легких рабочих столов *LXDE* и *Xfce*; кстати говоря, если Mate все же слишком роскошен для вашего ПК, читайте дальше.

LXDE/Xfce

http://lxde.org, www.xfce.org

Эта пара твердо ориентирована на тех, кто ищет простой, минимальный рабочий стол, с потреблением памяти втрое меньше, чем у тяжеловесов вроде KDE Plasma и Unity. Оба выглядят скучноватыми и устаревшими по сравнению с конкурентами, но это по сути лучший выбор для медленных компьютеров, таких как Raspberry Pi.

LXDE имеет ту же базовую компоновку, что и Cinnamon — всего одна панель внизу экрана, что напоминает Windows 95. Это не особо привлекательно, но вполне функционально и помогает свести беспорядок на рабочем столе к минимуму; как Mate, *LXDE* можно расширить лишь несколькими плагинами. Основные приложения тоже легковесные, но все же есть симпатичные мелкие штрихи, такие как поддержка двойной панели в файловом менеджере.

Xfce был первым легковесным рабочим столом, отгородившемся от современной отделки Gnome 3, но с тех пор его затмили окружения, подобные *LXDE*: чуть скромнее по системным ресурсам и при этом чуть современнее. Тем не менее, *Xfce* есть чем блеснуть: его главная панель по умолчанию находится вверху экрана, и нам нравится принцип размещения его меню приложений. Здесь мало возможностей настройки параметров, но *Xfce* может завоевать фанатов своим launcher'ом (программа называется **xfce4-panel**), который сидит в низу экрана и предлагает удобные ярлыки для различных элементов. Также следите за *Xfce Goodies* (http://goodies.xfce.org), который позволит вам сделать с ним еще больше.

Скорая помощь

LXDE — самый нетребовательный рабочий стол из рассмотренных нами: в тестах он запросил всего 87 МБ памяти, а Unity использует 312 МБ. Подробности см. http://bit.ly/deskbenchlinux.

KDE Plasma 5

www.kde.org

Если вам нужен полный контроль над рабочим столом, то KDE Plasma 5 — то, что надо. В некоторых отношениях он больше похож на каркас для постройки собственного рабочего стола, хотя версия 5 поставляется с разумными умолчаниями, которые позволяют вам с чего-то начать. Учтите, некоторые дистрибутивы все еще предлагают по умолчанию версию 4, так что будьте готовы подключить его репозиторий сами (например, через **kubuntu-ppa/backports**).

После запуска все остальное в ваших руках — предоставляется одна панель внизу экрана и удобная кнопка для вызова инструментов, расположенная в правом верхнем углу.

Здесь свершаются великие дела. KDE во многом основан на виджетах, которые можно прикрепить к панелям или оставить плавающими на рабочем столе. Их имеется много, но легко загрузить еще больше, прямо с рабочего стола. KDE также позволяет использовать Комнаты [Activities], которые напоминают виртуальные рабочие столы и позволяют настраивать ваш рабочий стол для конкретных целей, например для просмотра web-страниц или редактирования изображений.

KDE уникален среди рабочих окружений в этом мини-обзоре, будучи построенным на инструментарии *Qt*, а не *GTK*, что означает немного бóльшую ресурсоемкость, особенно при обновлении; может также оказаться, что имеющиеся приложения не вписываются в элегантный вид KDE. В результате, если вам нужен рабочий стол без лишней суеты, лучше поискать в другом месте; но если вы горите желанием построить рабочий стол с нуля, то KDE должен быть первым в списке. LXF

› *LXDE* — наш выбор для маломощных ПК; ему не хватает настраиваемости, но он отзывчив и быстр.

› KDE Plasma 5 не для слабых духом, но если вы хотите построить уникальный рабочий стол, его стоит попробовать.

III Бизнес-форум

19 февраля 2016

отель Azimut Санкт-Петербург
Санкт-Петербург, Лермонтовский пр., д. 43/1

Телеком двух столиц 2016

Санкт-Петербург

Эффективные пути повышения конкурентоспособности операторов связи в мегаполисах

Более **200** участников

Более **40** докладчиков – операторов, регуляторов и поставщиков оборудования и сервисов

Более **40** инфопартнеров

Поддержка ведущих отраслевых ассоциаций и объединений деятелей рынка

Расширенная выставка производителей и интеграторов телекоммуникационных решений

Ключевые темы форума:

- Переформатирование рынка услуг связи: новые вызовы и тренды
- Стратегии и возможности развития операторов связи в текущих экономических условиях
- Эффективные способы оптимизации телеком-инфраструктуры «большого города»
- Трансформация информационных технологий для оптимизации и развития телеком-бизнеса
- Информационные технологии на службе операторов связи: взгляд в будущее
- Воздушные линии связи в Москве и Санкт-Петербурге: перспективы инвентаризации и переноса в кабельную канализацию
- Импортозамещение как ключевой вектор развития телеком-рынка, заданный государством: планы операторов

Организатор:

COMNEWS CONFERENCES

Генеральный партнер:

МегаФон

При поддержке:

НАДИКС

АКПП

Информационные партнеры:

COMNEWS СТАНДАРТ BROADBAND & SATELLITE NEWSLETTER RUSSIA

Для регистрации: +7 495 933 5483, conf@[redacted], www.comnews-conferences.ru/tds2016

Мониторинг: Запряжем Conky

Хотите, чтобы ваш рабочий стол был уникальным и презентабельным? **Нейл Ботвик** покажет, как добавить системный монитор *Conky*.

Наш эксперт

У **Нейла Ботвика** богатый опыт загрузок, ведь у него по компьютеру в каждой комнате, а вот с перезагрузками похуже — с тех пор, как он перешел на Linux.

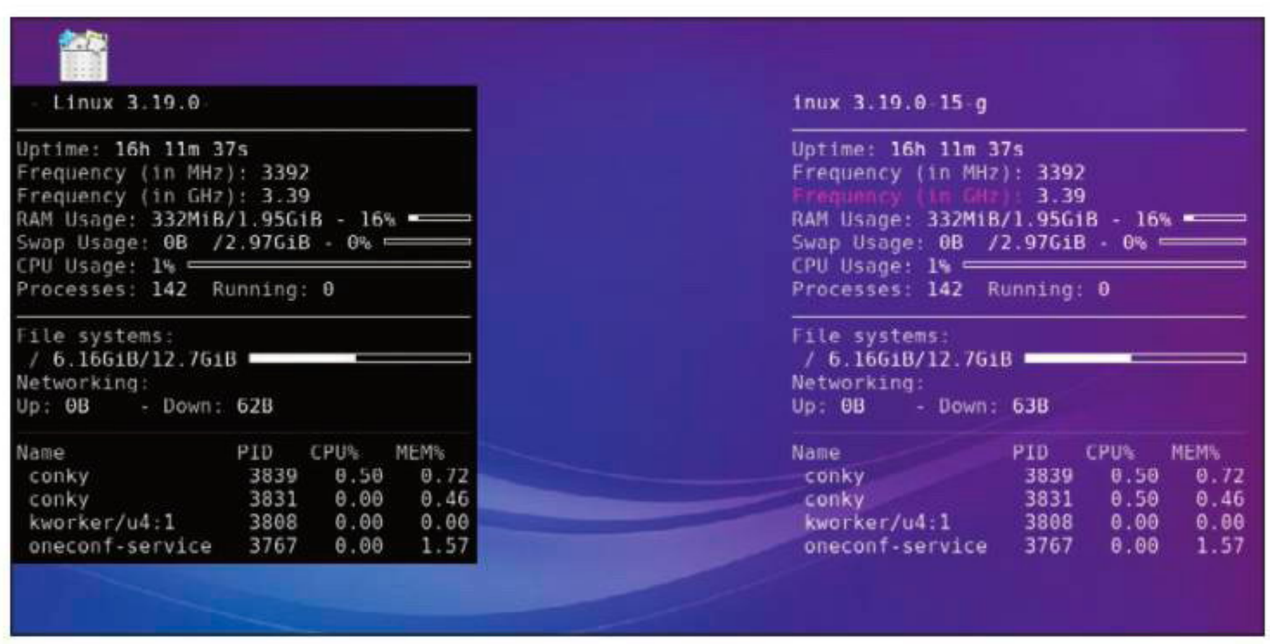
Системных мониторов нынче полно. В большинстве окружений рабочего стола имеется программа, сообщающая различные статистические данные о системе: время работы, уровень загрузки системы, свободной памяти, температуру и т. д. Как правило, эти данные отображаются в отдельном окне или в виде апплета в системном лотке. Тем не менее, пробуя различные live-дистрибутивы с **LXFDVD**, вы, возможно, заметили, что иногда такая информация представлена прямо на рабочем столе, и это выглядит более презентабельно. Также вы могли отметить, что способ отображения в каждом дистрибутиве свой. Делается это с помощью программы *Conky*, которая, благодаря гибкой настройке, позволяет выбрать, что именно отображается и где.

В менеджере пакетов вашего дистрибутива *Conky* должен быть, так что установите его обычным способом. Если вы пользователь Ubuntu, установите пакет *conky-all*, там есть кое-какие полезные дополнения. Затем запустите в терминале conky. Если вы ожидали, что перед вами мгновенно появится красивый дисплей, то извините: это будет суховатое черное окно с белым текстом, которое нам сейчас предстоит настраивать. Убейте его с помощью Ctrl+c — вот почему мы начали с терминала, а кто не начал, откройте его и выполните killall conky. Границы окна с кнопкой Закрыть здесь нет. *Conky* считывает свои настройки из **~/.conkyrc**, а поскольку такого файла нет, он загружает параметры по умолчанию из **/etc/conky/conky.conf**. Можно начать с создания нового файла настройки, но давайте упростим себе задачу, скопировав **/etc/conky/conky.conf** в **~/.conkyrc** и открыв его в вашем любимом редакторе. Файл содержит два раздела: первый определяет свойства окна *Conky*, а второй, после строки TEXT, определяет содержимое окна.

Давайте внесем несколько изменений: для начала переместим окно, найдем линию выравнивания и изменим ее на alignment top_right, выбрав верхнюю, нижнюю, среднюю, а затем слева, справа, посередине. Можно еще настроить gap_x и gap_y, изменив расположение слева/справа и вверху/внизу экрана. Теперь избавимся от этого черного фона: найдите настройки own_window (порядок не важен, но связанные настройки удобнее задавать вместе) и добавьте own_window_transparent yes.

Заодно можете изменить шрифт и размер. Чтобы не перезапускать *Conky* после изменения настроек, отправьте ему сигнал HUP из терминала, чтобы он перечитал свой конфигурационный файл, вот так: $killall -HUP conky.

Если вы используете Gnome или KDE, то можете не получить ожидаемого результата, поскольку в корневом окне рабочего стола они устанавливают свои обои. (Для решения, см. врезку *Почему нет прозрачности?* на стр. 63.)



Сменили в conkyrc всего один параметр, а разница налицо; вывод по умолчанию — слева.

Добудем побольше информации

Наладив способ отображения, перейдем к самому важному: определим, что именно будет отображаться. В окне *Conky* отображаются строки, идущие после TEXT. Они содержат простой текст, переменные и параметры; перед двумя последними есть префикс $, например, конфигурация по умолчанию выглядит как ${color grey} Uptime:$color $uptime.

Она устанавливает серый цвет текста, выводит слово Uptime, задает цвет по умолчанию, а затем отображает содержимое переменной $uptime, то есть информацию о времени работы системы. В верхней части **.conkyrc** есть две строки, которые влияют на это; одна устанавливает цвет по умолчанию, другая — частоту обновления содержимого переменных:

Запуск нескольких Conky

Conky может присутствовать не в единственном экземпляре; возможно, вы захотите, чтобы один отображал информацию о системе, а другой — о погоде или о ленте RSS. Для этого создайте несколько конфигурационных файлов, например, **~/.conkyrc_system** и **~/.conkyrc_rss**, а затем добавьте их в скрипт, подобный приведенному ниже, к которому будет обращаться система автозапуска:

```
#!/bin/sh
sleep 10
conky --config=~/.conkyrc_system
conky --config=~/.conkyrc_rss
```

Нечего и говорить, что в этом случае вам нужно будет также настроить разные позиции в отдельных файлах rc.

Почему нет прозрачности?

В некоторых рабочих столах обои отображаются поверх экрана, и чтобы их стало можно увидеть сквозь *Conky*, придется немного поколдовать. Хотя фон *Conky* прозрачный, за ним ничего не просматривается. Установите программу для просмотра изображений *feh* и укажите файл, используемый в качестве обоев, а затем создайте скрипт оболочки, вот таким образом:

```
#!/bin/sh
conky
feh --bg-fill /path/to/your/wallpaper
sleep 2
killall -HUP conky
```

Тогда *feh* будет отображать те же обои в качестве «реального» фона рабочего стола. Запустите этот скрипт вместо *Conky* и убедитесь, что в **.conkyrc** есть строка background yes, иначе команду feh он не увидит.

```
default_color white
update_interval 1.0
```

В **conkyrc** много переменных, отображающих время работы системы, уровень загрузки процессора, свободной памяти, скорости передачи данных, температуры и прочее. Все это перечислено на man-странице *Conky*, а список настроек и переменных можно найти на http://conky.sourceforge.net/config_settings.html и http://conky.sourceforge.net/variables.html. Переменные имеют префикс $; если вы хотите добавить к какой-то из них опцию, например $color, как указано выше, заключите имя переменной и опцию в фигурные скобки. Если вам нужен текстовый вывод, просто добавьте в **conkyrc** строку для каждого параметра, вместе с Label $variable. Можно также указать выравнивание. Чтобы текст был по левому краю, а значения — по правому, используйте следующие строки:

```
RAM:$alignr$mem/$memmax
Disk usage:$alignr${fs_used /home}/${fs_size /home}
```

Они показывают объем оперативной памяти и дискового пространства, а заодно и сколько их всего. Вывод в виде текста очень информативен и смотрится неплохо, если подстроить прозрачность и шрифты, но *Conky* умеет отображать объекты не только построчно. Во-первых, используя offset и voffset, вы можете менять расположение, а также шрифт, множество раз внутри одной строки; Gotham на **LXFDVD** — довольно радикальный тому пример. Кроме того, с помощью переменной image, имеющей синтаксис ${image path/to/image -p X,Y -s WxH}, можно добавлять изображения.

Настройки положения и размера менять в общем не обязательно; по умолчанию они расположены в левом верхнем углу окна в размере оригинала. Если изображение больше окна, оно обрезается; чтобы масштабировать его, укажите -s.

Пусть работают другие

Экспериментировать с файлом конфигурации **conkyrc** можно долго, но делать это не обязательно: есть готовые варианты, доступные на http://custom-linux.deviantart.com или на wiki *Conky* по адресу https://github.com/brndnmtthws/conky/wiki и можно даже их попробовать: *Conky Manager* (https://launchpad.net/conky-manager).

Большинство из тем, имеющихся на DeviantArt, подходят для использования с *conky-manager*, надо просто распаковать каждый архив в ваш каталог **~/.conky** (если такого нет, создайте его) и запустить *Conky Manager*. Появится список всех доступных тем, любую из которых вы можете просмотреть, отметив флажок рядом с ее именем. Количество просматриваемых тем не ограничено, а кнопка Настройки позволяет внести простые изменения в каждую из них, в частности, выравнивание и расположение: несколько параллельно запущенных вариантов *Conky* смотрятся так себе. *Conky Manager* содержит скрипт запуска — он будет вызывать экземпляры *Conky*, которые вы выбрали в *Conky Manager*; вы можете пользоваться им вместо запуска вручную.

Скорее всего, вы выберете, чтобы *Conky* запускался сам, с помощью функции автозапуска вашего рабочего стола. Однако это может создать проблемы, особенно с прозрачными окнами, поскольку *Conky* может запуститься еще до полной инициализации рабочего стола. Лучше всего запускать *Conky* скриптом (как показано во врезке *Почему нет прозрачности?* вверху), ставя пред началом работы *Conky* команду sleep, чтобы рабочий стол успел загрузиться — например, создать файл с таким содержанием и сделать его исполняемым, чтобы ваш рабочий стол запускал этот сценарий вместо *Conky*.

```
#!/bin/sh
sleep 10
conky
```

Если вы используете *Conky Manager*, функция sleep в скрипте запуска там уже есть.

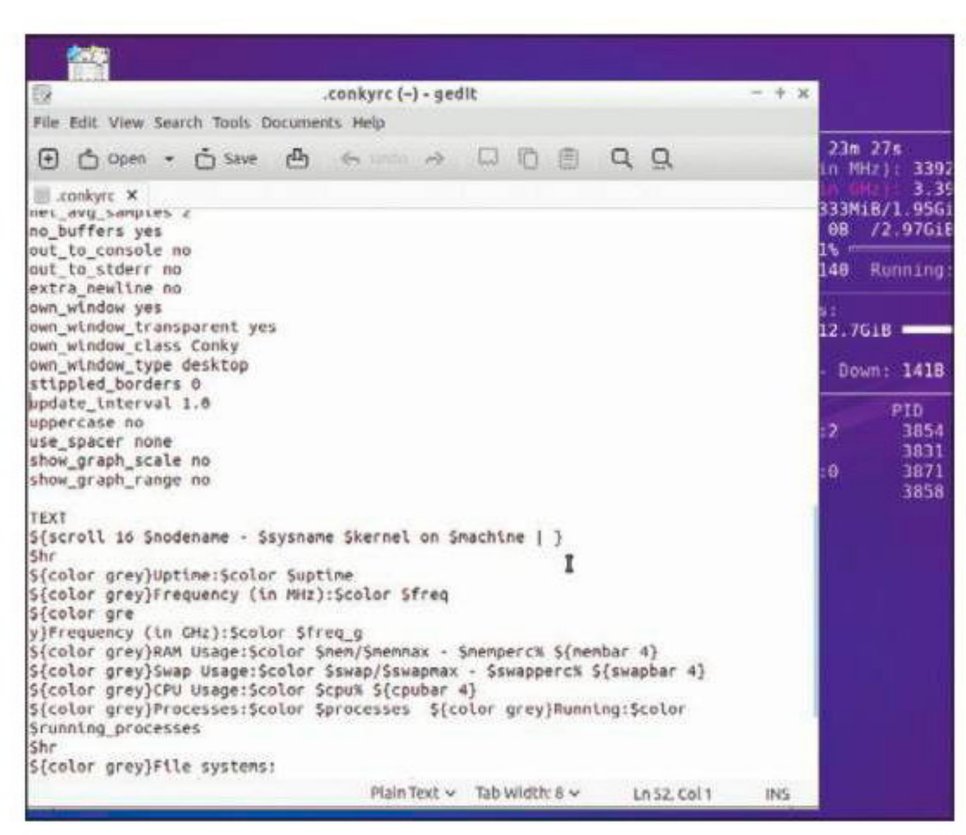
Трюки Conky

Как видите, *Conky* умеет отображать статическую и динамическую информацию о системе, но в запасе у него есть и другие трюки. Например, он может подключаться к *Audacious* или Music Player Daemon и отображать информацию о воспроизводимой в данный момент дорожке. Есть также различные варианты для отображения RSS-каналов. Еще одна полезная опция — отображение количества непрочитанных писем:

```
Unread Mail:$alignr${imap_unseen mail.example.com username
password}
```

Дополнительные опции позволяют задать интервал проверки (по умолчанию — пять минут), порт подключения и почтовый ящик для проверки. Подобная опция доступна для POP3. Температура в системе определяется через ACPI, а если вы установите *hddtemp*, *Conky* будет сообщать и о температуре жестких дисков.

Разобравшись с основными опциями, вы откроете для себя множество более продвинутых, в том числе скриптинг; но и основных вам будет более чем достаточно, чтобы приступить к мониторингу своей системы. LXF



› Поначалу файл конфигурации conkyrc кажется мудреным, но это просто перечень настроек и отображаемых параметров.

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

Ubuntu: Ставим принтер/сканер

Ник Пирс делится ценными советами о подключении и настройке ваших принтеров и сканеров системами и инструментами типа *CUPS*, *SANE* и *XSane*.

Наш эксперт

Ник Пирс подсел на Linux — сначала на Ubuntu, теперь на Minibian — вот уже как 10 лет. Он предпочитает инструменты с графическим интерфейсом, но не прочь и повозиться с терминалом.

Принтер и сканер ныне стали практически обязательным атрибутом компьютера, но вот их настройка в Linux может оказаться делом довольно хитрым. Впрочем, беспокоиться не стоит: при наличии правильных драйверов и инструментов у вас всё получится легко и просто. В большинстве дистрибутивов Linux есть два основных инструментария, облегчающих этот процесс: *CUPS* для принтеров и *SANE* для сканеров. *CUPS* — это централизованный инструмент для добавления и настройки принтеров, а *SANE*, соответственно, предоставляет интерфейс для управления сканером, также включающий средства просмотра и сканирования изображений. В данном учебнике мы рассмотрим оба, вкупе с полезными функциями, позволяющими осуществлять доступ к ним и тонкую настройку, не обращаясь к терминалу.

В других операционных системах, как правило, принтер и сканер подключаются довольно легко — драйверы либо предустановлены, либо прилагаются на диске производителя, либо вам подскажут, откуда их скачать. В Linux все не столь просто, но если вы знаете, как надо действовать, вам не придется тратить на это долгие часы.

Хорошо уже то, что при установке принтера не нужно сразу искать на сайте производителя. Если повезет, драйверы будут уже на месте, особенно если ваше устройство сетевое: в этом случае надо просто открыть приложение *CUPS* и проверить, обнаружит ли оно ваш принтер.

В последних версиях Ubuntu, включая 14.04.3. LTS, *CUPS* должен быть предустановлен, и вы найдете его в меню Параметры системы, щелкнув на Принтеры. Если его там нет, можно установить через *Software Centre* (введите в поиске «принтер» и выберите Принтеры) или через терминал: sudo apt-get install cups.

После установки нажмите кнопку Добавить и проверьте, видит ли система ваш принтер и поддерживается ли он. Если ваш принтер подключен по USB, но не обнаруживается — скорее всего, так и будет — надо сначала добавить подходящие драйверы. Сперва поищите на сайте производителя драйверы для Linux и конкретно для вашего дистрибутива (вероятно, это будет универсальный пакет). Есть надежда, что будет поддерживаться ваш дистрибутив (вплоть до номера версии), но если это не так, паниковать не надо: мы успешно установили МФУ B1265nf от Dell с помощью универсального пакета, обещавшего только поддержку версий ранее 11.04.

Если повезет, драйвер будет иметь формат Deb-пакета, тогда нужно будет только дважды щелкнуть по нему, чтобы запустить мастер установки. Если драйвер в виде tar-архива (с расширением **.tar.gz**), то там, вероятно, будет инсталлятор в виде скрипта оболочки (**.sh**); тогда следуйте пошаговым инструкциям на стр. 65. По завершении, вновь откройте приложение *CUPS* в Параметрах системы и убедитесь, что принтер обнаружен.

Установка через сеть

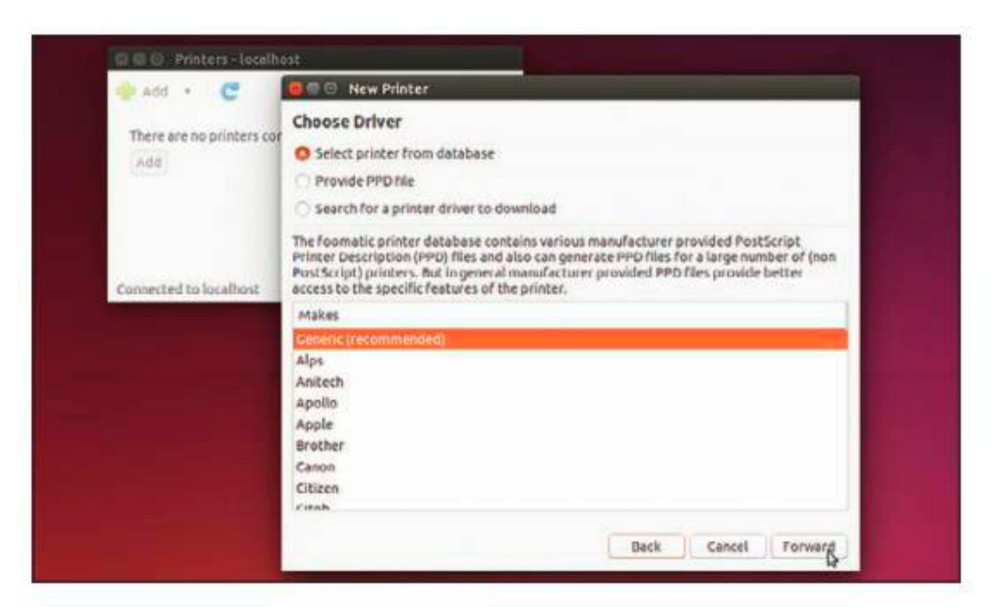
Если вы добавляете сетевой принтер, он должен обнаруживаться и без предварительной установки драйверов. Нажмите кнопку Добавить, выберите Сетевой принтер и подождите, или нажмите кнопку Найти сетевой принтер. После короткой паузы все обнаруженные принтеры должны появиться на экране. Выберите из них свой, и увидите доступные протоколы связи. (Если их несколько, смотрите подробнее в разделе *Какой тип подключения?* на стр. 66) Выберите IPP, если он доступен; в противном случае выберите LPD или посмотрите, нет ли на сайте производителя драйвера с возможностью IPP-подключения.

Если варианты, предложенные *CUPS*, вас устраивают, нажмите Вперед, и программа автоматически найдет и установит доступные драйверы. Если таковых не обнаружится, вам предложат три варианта: выбрать из базы данных, предоставить файл PPD или искать драйвер для загрузки. Начните с выбора производителя принтера, чтобы увидеть, есть ли точное совпадение. Если нет, попробуйте поискать — возможно, при помощи вашего web-браузера, тогда поле поиска будет шире.

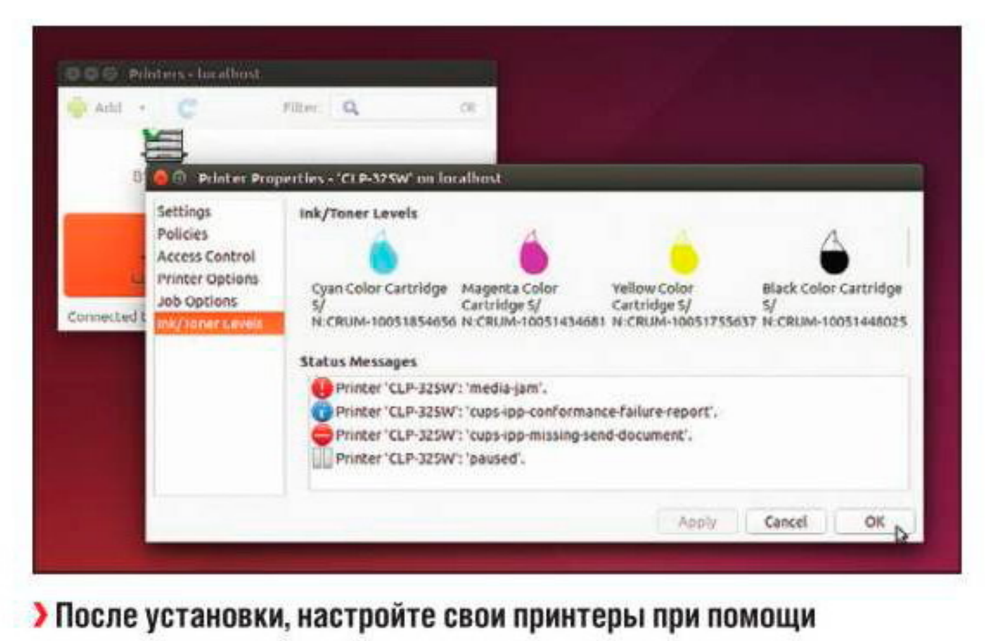
Если у вас принтер модели PostScript, можете также использовать файл PostScript Printer Description (PPD), если он имеется (введите в поиске название и модель принтера и слово “PPD”). Таким образом *CUPS* получает описание его возможностей, которые он может использовать при выводе файлов.

Если ничего не выходит, не волнуйтесь: снова нажмите Выбрать принтер из базы данных, но на сей раз выберите Универсальный (рекомендуемый) и нажмите Вперед. Универсальный драйвер PCL является предустановленным, так что попробуйте сначала либо pxlmono, либо pxlcolor, в зависимости от того, черно-белый у вас принтер или цветной. Чтобы установить его, нажмите кнопку Вперед — в большинстве случаев это обеспечит вам необходимую базовую функциональность.

Какой бы способ добавления принтера вы ни использовали, вам предложат ввести название, описание и расположение, а затем



Благодаря *CUPs* сетевые принтеры зачастую можно настроить даже без драйверов.



› После установки, настройте свои принтеры при помощи соответствующего инструмента в Параметрах системы.

напечатать тестовую страницу — рекомендуем так и сделать, чтобы проверить, работает ли подключение. После этого появится экран настройки принтера, где вы можете изучить его параметры и посмотреть, что можно изменить.

Тонкая настройка принтера

В основной вкладке Настройки можно изменить описание и расположение вашего принтера, а также его URL. Нажав Изменить... рядом с URL устройства, вы можете попробовать выбрать другой протокол. Также вы найдете здесь параметры печати пробных страниц и поддержки принтера.

Вкладка Правила позволяет изменить состояние принтера, определив, является ли он включенным, готовым к работе или настроенным для общего доступа. Вы также можете понять, что происходит, если принтер выдает ошибку, и включить аутентификацию в «Правилах использования», если эта функция поддерживается и требуется (при выборе этого варианта настройте принтер через web-интерфейс *CUPS*).

Контроль доступа позволяет запретить (или разрешить) доступ к принтеру на уровне пользователя. Придется покопаться в Параметрах принтера и Параметрах задания, задать ключевые настройки по умолчанию. Вкладка Параметры принтера в основном касается настроек качества печати и формата бумаги по умолчанию, тогда как Параметры задания позволят настроить такие вещи, как количество печатных копий, ориентацию страницы и отдельных элементов, таких как текст и изображения.

Наконец, вкладка Уровень чернил/тонера, если таковая имеется в вашем принтере, позволяет отслеживать состояние основных компонентов вашего принтера, в том числе таких, как термофиксатор.

Теперь ваш принтер настроен, и пришло время его использовать. Откройте приложение и, когда будете готовы, выберите Файл > Печать. Принтер должен быть в списке, так что остается только подправить параметры, имеющиеся в диалоговом окне, и нажать Печать. Если у вас установлено несколько принтеров, один из них всегда будет задан по умолчанию; изменить его можно, используя диалог в меню Настройки системы > Принтеры, щелкнув правой кнопкой на нужный принтер и выбрав Установить по умолчанию.

В большинстве случаев инструмент *CUPS*, находящийся в Параметрах системы, включает все необходимые опции, но *CUPS* можно также настроить и через браузер: для доступа к нему, введите localhost:631. Там вы найдете удобный обзор и ссылки на форум *CUPS*, если он вам понадобится; для управления web-утилитой используйте вкладки в верхней части страницы. Перейдите в Принтеры и выберите свой, для получения доступа к его параметрам — навигация здесь осуществляется с помощью выпадающих меню. Если вы установили специальный драйвер, привязанный к определенному пользователю, вам на данном этапе предложат ввести имя пользователя и пароль для доступа к требуемым страницам.

Если вы настроили сетевой принтер, у вас также должен быть прямой доступ к настройкам самого устройства — для этого надо ввести его IP-адрес. Чтобы его узнать, загляните в раздел Настройки принтера — нажмите кнопку Изменить рядом с URI устройства и откройте Сетевой принтер. IP-адрес будет указан в скобках после названия вашего принтера. Введите адрес URL, и у вас будет доступ к самому принтеру и даже больше возможностей управления им (например, для создания дуплексных вариантов двусторонней печати).

Классы принтеров

CUPS позволяет группировать принтеры в классы. Главным образом это предназначено для работы в сетях, где множество »

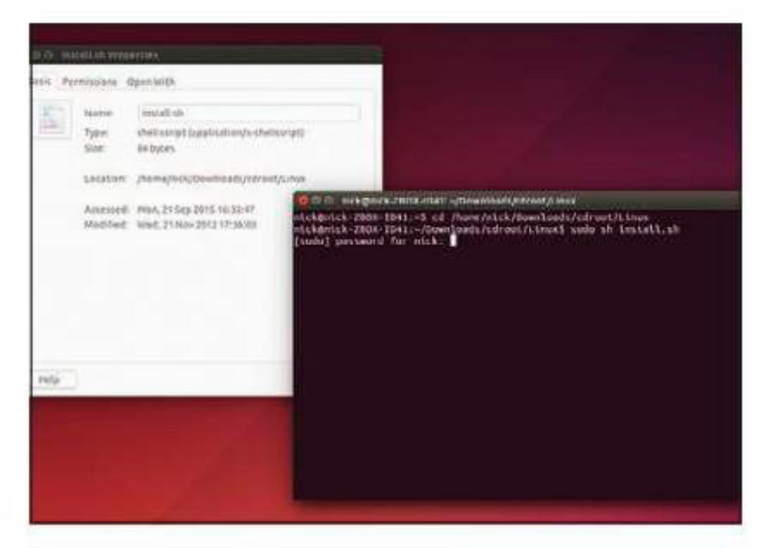
Скорая помощь

Никак не наладить правильную работу принтера? Загляните на http://openprinting.org/printers и поищите там описание своей модели. Если оно есть, пролистайте комментарии — не исключено, что кто-нибудь вам поможет. Или задайте свой вопрос на форумах.

Устанавливаем драйвер принтера из tar-архива

1 Добыть и распаковать установщик

Зайдите на сайт производителя и найдите нужный драйвер для вашего дистрибутива Linux; возможно, это будет универсальный драйвер. Сохраните файл **.tar.gz** в папку **Загрузки**, щелкните на ней правой кнопкой и выберите Распаковать сюда. После этого откройте разархивированную папку и найдите файл **install.sh.** Правым щелчком выберите Свойства, чтобы запомнить, где он находится.



2 Запустить установку

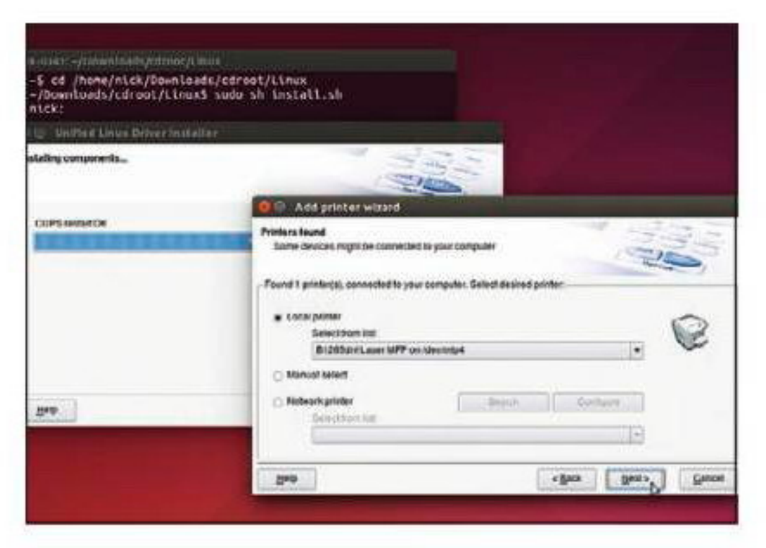
Теперь откройте терминал и введите

```
cd <path>
```

— вместо <path> подставьте путь, указанный в строке Расположение окна Свойства файла **install.sh.** Затем введите команду

```
sudo sh install.sh
```

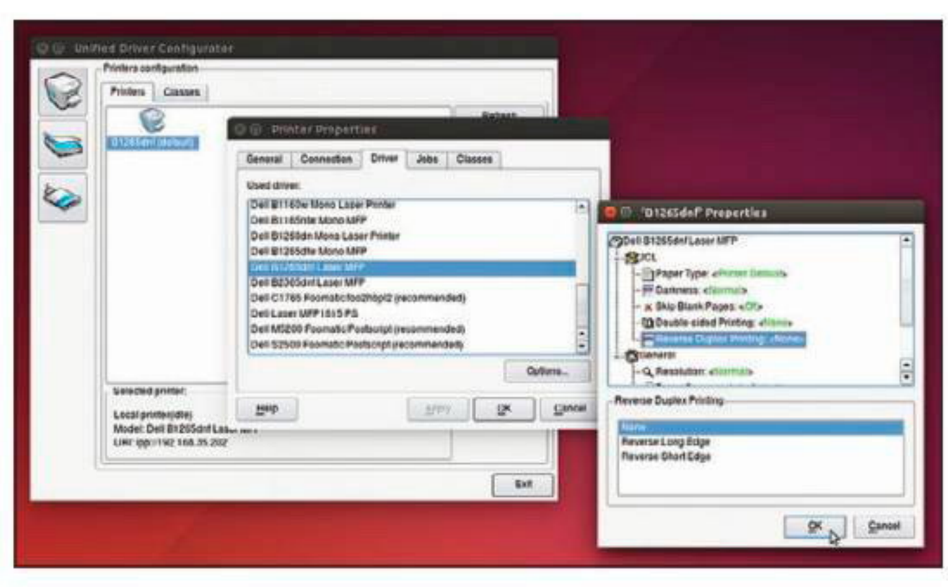
и, при необходимости, пароль пользователя, и нажмите Enter. После небольшой паузы должна открыться программа установки.



3 Следовать инструкциям

Следите за работой программы установки, да не забудьте указать свое имя пользователя, чтобы иметь доступ к принтеру.

По завершении установки вам, скорее всего, будет предложено настроить ваш принтер для *CUPS*, и, возможно, потребуется ввести название, описание и расположение принтера, прежде чем вы сможете перейти собственно к его использованию по прямому назначению.



› В некоторых принтерах есть свои программы настройки, дающие больше возможностей контроля.

› Успех — или провал — OCR в основном зависит от качества документа-оригинала.

пользователей хотят иметь доступ к малому количеству принтеров. Каждый класс содержит иерархию принтеров, где добавленные в первую очередь предпочтительнее, чем те, что добавлены позже. При печати с использованием класса принтер назначается по принципу доступности, то есть если печатает один человек, он всегда получает доступ к первому принтеру, а если два человека одновременно, то первый получает доступ к приоритетному варианту, а второй — к следующему доступному.

Исходя из этого, необходимо определить принцип классификации ваших принтеров: будут ли первыми в очереди те, что печатают быстрее, или вы хотите направить людей к тем, у которых больший объем печати, как у лазерных принтеров? После этого откройте Параметры системы > Принтеры, нажмите кнопку "˅", рядом с кнопкой Добавить, и выберите Класс.

Вам предложат ввести название, описание и расположение для одного принтера, но лучше будет думать в терминах «класса», например, «учебные принтеры» или нечто подобное. После этого нажмите Вперед, и вам предложат добавить к классу существующие принтеры — не забудьте сделать это в порядке предпочтения. Далее, нажмите кнопку Применить. Потом вы можете установить этот класс в качестве используемого по умолчанию вместо одного принтера — щелкните правой кнопкой на нужной записи и выберите Установить по умолчанию.

В Linux есть также хорошие встроенные инструменты для доступа и использования сканера. Во многих случаях — в частности, с более новыми моделями — можно просто подключить его к компьютеру, и велика вероятность, что он просто будет работать. Отчасти это возможно благодаря встроенной поддержке *SANE* (Scanner Access Now Easy), которая обеспечивает все внутренние функции сканирования в Linux.

Чтобы посмотреть, повезло ли вам, подключите сканер к свободному порту USB и запустите *Simple Scan*, чтобы проверить, обнаружит ли его система. Если да, то дело в шляпе — переходите к следующему разделу; если нет — не паникуйте. Во-первых, проверьте наличие драйверов для Linux у своего производителя. Следуйте инструкциям, чтобы загрузить и установить их; опять же, если имеется скрипт **install.sh**, следуйте пошаговому руководству (см. стр. 65).

Если ваш сканер является частью многофункционального устройства, должны быть общие драйверы для сканера и принтера;

Скорая помощь

Если вы пытаетесь распознать текст из документа с несколькими колонками, непременно используйте инструменты *YAGF* с функциями коррекции перекоса страницы и выбора нескольких столбцов, это увеличит ваши шансы на успех.

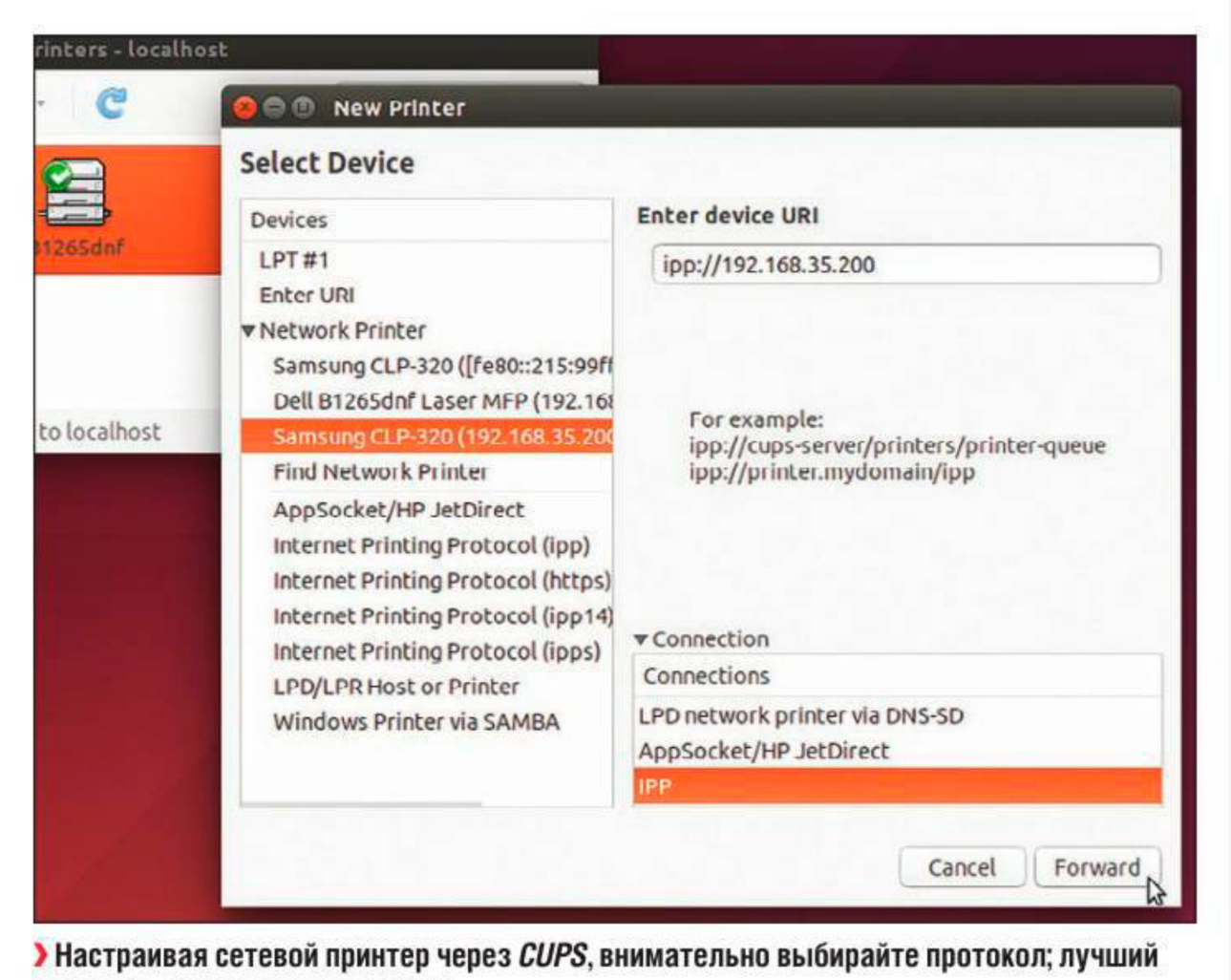
Какой тип подключения?

Если вы подключаете сетевой принтер, то при настройке вам предложат выбрать соединение для его настройки. Но что это за протоколы, и какой нужно выбрать, если выбор у вас есть?

Самый простой — протокол AppSocket (также известный как HP JetDirect). Как правило, он используется в отсутствие других альтернатив, но простота также означает, что никаких тонкостей управления в нем нет: например, нет управления очередью, так что вы не сможете прервать задания, отправленные на печать. Также нельзя узнать состояние принтера, а значит, к примеру, проверить уровни чернил и тонера.

Далее идет LPD, он же Line Printer Daemon (он же — LPR, или Line Printer Remote). Здесь уже есть управление очередью, а также распознавание типов данных, таких как неформатированный текст или информация PostScript. Но нет возможностей удаленного подключения или настройки, и опять-таки нельзя посмотреть уровень чернил или тонера (хотя иногда бывает, что и можно), и ваши задания на печать не могут быть зашифрованы.

Последний протокол — IPP — самый современный и наиболее функциональный. Его полное название — Internet Printing Protocol, и именно его *CUPS* берет за основу для управления заданиями и очередями. Это наиболее предпочтительный способ подключения, поскольку вы получаете больше обратной связи от принтера и можете лучше контролировать его, благодаря его двунаправленной природе.



› Настраивая сетевой принтер через *CUPS*, внимательно выбирайте протокол; лучший вариант — IPP, если он доступен.

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

Подключаем неподдерживаемые сканеры

Если, несмотря на все ваши усилия, вы все же не сумели подключить свой сканер через *SANE*, обратитесь на www.hamrick.com, там вы найдете программу под названием *VueScan*, с поддержкой более 2500 моделей.

Нажмите на ссылку Поддерживаемые сканеры и посмотрите по производителю — если ваша модель есть, скачайте tar-архив и распакуйте папку **VueScan**, а затем двойным щелчком по *vuescan* внутри нее запустите установку самой программы. Она автоматически обнаружит ваш сканер и предоставит пошаговые инструкции для просмотра и сканирования. *VueScan* содержит множество полезных инструментов для достижения безупречного результата и работает со многими моделями, уже поддерживаемыми в Linux, так что если с *XSane* у вас не клеится, попробуйте это. Единственный недостаток в том, что она платная — Стандартная версия обойдется вам в $39,95, а Профессиональная стоит $89,95 (чтобы почувствовать разницу, можно в любой момент обновиться со Standard до Pro). Хорошая новость в том, что *VueScan* распознает всё, что пожелаете (кроме разве что водяного знака) на всех ваших сканах, так что с покупкой вы не прогадаете.

если вы добавили компонент принтера из МФУ без установки отдельных драйверов, нужно будет сделать то же и для сканера. Если вы устанавливали драйверы, но сканер не обнаружен, попробуйте установить их снова поверх исходных драйверов — с нашим Dell B1265dnf это сработало. После установки драйвер может предоставить собственную утилиту для проверки обнаружения сканера — например, пользователи Dell должны увидеть на рабочем столе ярлык DELL Unified Driver Configurator; можно опять же запустить *Simple Scan*, теперь сканер должен распознаваться.

Для получения дополнительной консультации по устранению неполадок в *SANE* при обнаружении сканера обратитесь на https://help.ubuntu.com/community/Scanners или просмотрите соответствующий раздел в *VueScan*.

Как видите, в Ubuntu есть оболочка *Simple Scan* для *SANE*, предоставляющая простой, без излишеств, интерфейс для сканера. Он хорошо подходит для проверки работоспособности сканера, но ему не хватает ключевых инструментов, таких как функция предпросмотра или возможность тонкой настройки параметров вашего сканера.

Пора сканировать

А если вы хотите, чтобы все было по полной программе, откройте *Ubuntu Software Center* и введите в поиске "xsane", чтобы найти и установить *XSane*, более мощный интерфейс для сканирования. После запуска он будет отображать четыре окна — для начала, нажмите Добавить предпросмотр, чтобы создать предпросмотр изображения, которое вы пытаетесь сканировать. После этого используйте окно Основные параметры, чтобы выбрать изображение — выберите Стандартный под Форматом страницы, а затем обрежьте до нужного размера при помощи четырех измерителей под ним.

Затем нажмите кнопку лупы 1:1 в главном окне, чтобы увеличить изображение в предпросмотре. Инструментом Пипетка выделите белые, черные и серые части изображения для коррекции цвета, а затем используйте ползунки гистограммы в верхнем левом окне для тонкой настройки цветового баланса. В правом окне, где вы выбираете, куда сохранить файл, какой формат использовать и так далее — выберите 16 миллионов цветов для фотографий и убедитесь, что разрешение сканирования увеличено от 75 dpi до 300 dpi или 600 dpi, если вам нужно сканирование в высоком разрешении. Есть также элементы управления для регулировки яркости, контрастности и гаммы.

По умолчанию, при нажатии Сканировать ваше изображение откроется в окне просмотра в *XSane* — это хорошо для первого сканирования, поскольку вы сможете просмотреть изображение и выполнить дальнейшую оптимизацию. Удовлетворившись результатом, нажмите кнопку Просмотр и измените на Сохранить, а затем нажмите Сканировать еще раз.

Еще одно преимущество *XSane* — интеграция с *GIMP*: это позволяет сканировать прямо в приложение, чтобы затем довести ваше изображение до совершенства. Запустите *GIMP* и пройдите в Файл > Создать > XSane, чтобы выбрать свой сканер или выполнить поиск доступных устройств. Делается все точно так же, кроме одного существенного момента: не будет опции вывода, поэтому при нажатии Сканировать изображение будет автоматически передано в *GIMP*.

Через инструмент *goocr* можно также связать *XSane* со встроенным движком оптического распознавания (OCR). Тогда можно сканировать печатные документы и конвертировать их в редактируемый текст. Это гибкое решение, но если вы хотите попробовать что-то еще, то установите Cuneiform и *YAGF* из *Ubuntu Software Center*. Cuneiform представляет собой альтернативный движок OCR, а *YAGF* обеспечивает опрятный интерфейс как для Cuneiform, так и для Tesseract — еще одного OCR-движка, славного своей точностью. *YAGF* также прекрасно интегрируется с *XSane*: запустите *YAGF*, нажмите кнопку Сканировать, и откроется *XSane*. Установите цвет Черный и белый — Line Art и разрешение 600 dpi. Нажмите Сканировать, и документ будет отсканирован и отправлен обратно в *YAGF*. Осмотрите качество сканирования, а затем используйте элементы управления, расположенные выше сканированного изображения, чтобы подготовить его для распознавания. После этого нажмите на красную кнопку OCR рядом с выпадающим меню языка. *YAGF* по умолчанию использует для декодирования страниц Tesseract, а редактируемый текст будет помещен в правой панели, где его можно просматривать и править. Не нравится результат? Выберите Настройки > Настройки распознавания, затем Cuneiform, и нажмите OK, чтобы попробовать его — в наших тестах Tesseract проявил себя лучше, хотя и не намного. Будьте готовы, при необходимости, подправить и подстроить параметры исходного сканирования, чтобы текст был максимально ярким и четким.

Завершив просмотр и редактирование текста, можете сохранить его в виде файла TXT или скопировать в буфер обмена, чтобы вставить в *Writer* или другой редактор. Поздравляем, теперь ваш сканер настроен для выполнения стандартного и текстового сканирования. LXF

Скорая помощь

С помощью *CUPS* можно также «распечатать» документы в файл PDF, если отдельно таковая функция отсутствует. Установите *cups-pdf* из *Ubuntu Software Center* — и при следующей печати появится новый PDF-принтер.

› *XSane* содержит полный набор инструментов для сканирования.

GNU Core Utilities: часть 3

Под руководством **Дмитрия Пантелеичева** путешествуем дальше по базовому программному пакету из окружения GNU.

Наш эксперт

Дмитрий Пантелеичев считает, что любую технологию надо изучать от простого к сложному, и каждый шаг закреплять практическими примерами.

Итак, начало изучения пакета GNU Core Utilities положено (см. Учебники **LXF202** и **LXF203**). Делаем глубокий вдох и идем дальше: дальше еще интереснее.

Операции над файлами

Для работы с уже созданными файлами служат утилиты *cp*, *install*, *dd*, *mv*, *rm*, *shred*.

Первая утилита, которую мы рассмотрим, *cp*, копирует файлы. По умолчанию, получившаяся копия является совершенно независимым экземпляром, никак не связанным с исходным файлом.

Создадим в текущем каталоге новый файл с именем **source**.

```
cat > source << EOF
Первая строчка
EOF
```

Скопировать его командой cp можно тремя способами — указав как параметры: 1) имя файла и целевой каталог — туда файл скопируется с прежнем именем; 2) старое и новое имя файла — копия файла с новым именем появится рядом со старым файлом; 3) имя файла, целевой каталог и новое имя — файл скопируется в целевой каталог с новым именем.

Скопируем наш файл в каталог **Документы** с именем **target**: cp source Документы/target. Затем сделаем копию с именем **target** в текущем каталоге: cp source target. И, наконец, копию с именем **target** — в каталоге **Документы**: cp source Документы/target. Проверьте, что в обоих каталогах появилось по два файла: **source** и **target**.

Теперь к исходному файлу добавим вторую строчку:

```
cat >> source << EOF
Вторая строчка
EOF
```

Просмотрим исходный файл: cat < source. Мы должны получить на экране две строки:

```
Первая строчка
Вторая строчка
```

Теперь проверим все скопированные файлы.

```
cat < Документы/target
cat < Документы/source
cat < target
```

Появилась только первая строка. Т.е. изменения в исходном файле никак не влияют на файл конечный.

Теперь проделаем ту же операцию, добавляя ключи -l и -s.

```
cp -l source target_hard
cp -s source target_symbolic
```

У нас появились две новые копии файла, который, как вы помните, в своей последней версии состоял из двух строк. Соответственно, обе его новые копии также должны состоять из двух строк. Можете проверить это командой cat.

Добавим к исходному файлу третью строку:

```
cat >> source << EOF
Третья строчка
EOF
```

Проверим обе новых копии нашего файла:

```
cat < target_hard
cat < target_symbolic
```

К обеим этим копиям тоже добавилась третья строчка! Итак, эти файлы не независимы, а как-то связаны с исходным файлом.

Выведем полное содержимое текущего каталога с помощью команды vdir (она выводит содержимое в расширенном формате). Фрагмент получившегося результата выглядеть будет так.

```
-rw-r--r--. 2 dima students 84 сен 5 22:06 source
-rw-r--r--. 1 dima students 28 сен 5 22:00 target
-rw-r--r--. 2 dima students 84 сен 5 22:06 target_hard
lrwxrwxrwx. 1 dima students 6 сен 5 22:04 target_symbolic -> source
```

Начнем с первого поля формата. У копии **target_hard** это -, у копии **target_symbolic** — l. Мы знаем, что значения первого поля следующие: файл (-), каталог (d), ссылка (l). Выходит, одна из двух наших копий, связанных с исходным файлом, является ссылкой (что понятно), другая — настоящим файлом (что не очень понятно).

Второе, что вы заметите — размер файла. Копия **target_hard** имеет точно такой же размер (84), как и исходный файл. Вторая же копия, **target_symbolic**, имеет значительно меньший размер (6).

И, наконец, третье — это запись имени элемента. В первом случая имя записано просто (target_hard), во втором — более сложно (target_symbolic -> source).

Разгадка тут вот в чем. Здесь мы имеем дело с двумя разными видами ссылок. Ссылки бывают жесткими и символическими. Символическая ссылка содержит только указание на местоположение исходного файла. При обращении к этой ссылке система просто переходит к исходному файлу. А с жесткими ссылками ситуация такова. Любой файл имеет три характеристики: 1) занятое место на диске, где и записана информация; 2) индексный дескриптор, уникальный внутри дискового раздела и содержащий атрибуты файла; 3) связанный с файлом элемент каталога. Так вот, элементов каталога у одного и того же файла может быть несколько. Поскольку все они по структуре одинаковы, система воспринимает каждый из них как отдельный файл. Удаление всех жестких ссылок на данный файл означает удаление самого файла. Место на диске, занимаемое файлом, становится доступным и впоследствии может перезаписываться другими файлами. Поскольку индексный дескриптор уникален внутри раздела диска, то создание жесткой ссылки на файл на другом разделе диска невозможно: можно создать только символическую ссылку.

По утилите *cat* добавим, что по умолчанию она копирует только файлы, пропуская каталоги. Если же вызвать ее с ключом -r (--recursive), она скопирует также каталоги со всем содержимым.

Создадим каталог **mail**, а в нем файл **friend**.

```
mkdir mail
cat > mail/friend << EOF
```

```
Письмо от друга
EOF
```

Попробуем скопировать весь каталог в **Документы**:

```
cp mail Документы
```

Провал! На экране появляется следующее:

```
cp: пропускается каталог "mail"
```

А с ключом -r (--recursive) наша цель будет достигнута.

```
cp -r mail Документы
```

Следующая утилита, *install*, копирует файлы и позволяет одновременно с копированием задавать атрибуты конечного файла. Владелец файла задается с помощью ключа -o (--owner), группа — с помощью ключа -g (--group), а режим — -m (--mode).

Давайте создадим таким образом еще одну копию файла **source**, но владельцем ее установим пользователя root; поместим его в группу students (если группу не укажем, то по умолчанию файл будет помещен в группу root, к которой принадлежит владелец), а режим установим такой: владелец имеет право читать и писать, члены группы — только читать, остальные не могут ничего. Операцию эту необходимо выполнять с правами суперпользователя.

```
install -o root -g students -m u=rw,g=r,o= source target_root
```

Выведем список файлов в расширенном формате (vdir), чтобы посмотреть, что у нас вышло. Видно, что новая копия принадлежит суперпользователю-root, группе students, и все права установлены в соответствии с нашими пожеланиями.

```
-rw-r--r--. 2 dima students 84 сен 5 22:06 source
-rw-r--r--. 1 dima students 28 сен 5 22:00 target
-rw-r--r--. 2 dima students 84 сен 5 22:06 target_hard
-rw-r-----. 1 root students 84 сен 5 22:32 target_root
lrwxrwxrwx. 1 dima students 6 сен 5 22:04 target_symbolic -> source
```

Не ленитесь смотреть справку по каждой утилите. Как вы помните, она вызывается командой с ключом --help.

Утилита *dd* предназначена для преобразования информации. Исходные данные она берет или из файла, или из стандартного ввода. Результат помещает также или в файл, или в стандартный вывод. Ключей у нее нет (кроме обычных --help или --version), зато много параметров. Познакомимся с первым из них: conv=<значение>. Значения, принимаемые этим параметром, покажет --help. Но в качестве самых наглядных приведем lcase (перевод в нижний регистр) и ucase (перевод в верхний регистр).

```
dd conv=ucase << EOF
Hello!
EOF
```

Поскольку мы не указали ни исходный, ни конечный файл, информация будет считана со стандартного ввода, а результат отправлен в стандартный вывод.

```
HELLO!
0+1 записей получено
0+1 записей отправлено
скопировано 7 байт (7 B), 3,5478e-05 с, 197 kB/c
```

Параметр if=<имя файла> показывает, из какого файла читать. Параметр of=<имя файла> показывает, в какой файл писать. Например, вышеописанную задачу по переводу текста в верхний регистр можно выполнить, сохранив результат в файл.

```
dd conv=ucase of=helloworld << EOF
Hello, world!
EOF
```

Откроем полученный файл — dd if=helloworld — и убедимся, что в файле текст сохранился с переводом в верхний регистр.

```
HELLO, WORLD!
0+1 записей получено
0+1 записей отправлено
скопировано 13 байт (13 B), 3,3852e-05 с, 384 kB/c
```

Параметр skip=n указывает, какое количество блоков надо пропустить при преобразовании файла. Параметр count=n указывает, какое количество блоков надо прочитать. Параметр bs=bytes задает количество байтов в блоке.

Попробуем пропустить первые 7 символов в нашем текстовом файле и вывести следующие 2. Поскольку символ латинского алфавита соответствует одному байту, надо установить размер блока в один байт.

```
dd if=helloworld bs=1 skip=7 count=2
```

В результате должны прочитаться только два символа.

Много и других параметров имеет эта команда. Найдите их в справочной системе (--help) и поэкспериментируйте с ними.

Утилита *mv* предназначена для перемещения или переименования файлов. Чтобы переименовать файл, достаточно указать новое имя как параметр: mv helloworld hello. Чтобы переместить файл, надо указать путь к новому месту без имени файла:

```
mv hello Документы
```

А можно и указать имя, и при этом его изменить:

```
mv Документы/hello ./greeting
```

Ключ -v (--verbose) выдает в консоли информацию о том, какое действие программа сейчас произвела. Этот ключ используется во многих программах, выполняющих какое-либо действие, не связанное с выводом текста на экран. Использование ключа -v во многих случаях позволяет сразу увидеть ошибку и ее исправить.

```
mv -v greeting hello
```

На экране должно появиться следующее.

```
"greeting" -> "hello"
```

Еще один интересный ключ — -b (--backup) — защитит от случайного стирания каких-то важных файлов. Давайте сделаем копию файла **hello** в папку **Документы** — cp -v hello Документы —и перенесем туда же исходный файл. При этом возникнет опасность, что мы сотрем в папке **Документы** старый файл **hello**, в который мы могли внести важные изменения, а потом забыть об этом. Ключ -b создаст копию старой версии файла, которую мы сможем потом просмотреть и, если ничего важного там нет, окончательно удалить. Учтите, что сокращенные версии ключей можно объединять в один. Так, -v в сочетании с -b можно объединять в одну запись: -vb. Команда mv -vb hello Документы выведет на экран следующий комментарий.

```
"hello" -> "Документы/hello" (резервная копия: "Документы/hello~")
```

Просмотрев каталог **Документы** (ls Документы), мы увидим, что там появился файл **hello~**, содержимое которого — старая версия файла **hello**.

Можно перемещать и несколько файлов одновременно. Скажем, в нашем текущем каталоге осталось много файлов, начинающихся на **target**. (Помните, мы их понаделали,разбирая разные виды копирования?) Все эти файлы можно разом переместить в каталог **Документы**. Дело в том, что многие программы, работающие с файлами, при задании имени файла в качестве аргумента допускают символ подстановки (он же — символ-джокер или wildcard character). Символ подстановки позволяет указать на несколько файлов одновременно, объединив их по некому признаку. Наиболее распространенным таким символом является звездочка, позволяющая объединить файлы по одинаковому фрагменту их названия: mv -v target* Документы.

Наконец, можно перенести все файлы и каталоги, если в качестве их названия указать символ подстановки *:

```
mv -v Документы/* Общедоступные
```

— все файлы из **Документы** перешли в каталог **Общедоступные**.

Утилита *rm* уже знакома вам по примерам (**LXF203**). Добавим, что, как и другим утилитам по работе с файлами, можно удалить сразу несколько файлов, если их названия имеют одинаковые последовательности символов, по которым их можно объединить. Например, удалим из каталога **Общедоступные** все файлы с именами, начинающимися на **target**: rm -v Общедоступные/target*. Либо можно вообще удалить все файлы в данном каталоге, задав * в качестве параметра: rm -v Общедоступные/*.

Ссылки бывают символическими и жесткими. Символическая ссылка содержит только путь к исходному файлу. И при обращении к ссылке система переходит к исходному файлу. Жесткая ссылка является элементом каталога и связана с индексным дескриптором файла. На каждый файл имеется как минимум одна жесткая ссылка, но их может быть и несколько. Все они воспринимаются системой как самостоятельные файлы, хотя ведут на одно и то же место на диске.

»

› Рис. 1. Результат работы утилиты *shred*.

```
Terminal
dima@dima-System-Product-Name ~ $ shred -v phone
shred: phone: проход 1/3 (random)…
shred: phone: проход 2/3 (random)…
shred: phone: проход 3/3 (random)…
dima@dima-System-Product-Name ~ $ cat < phone
```

Но, как вы уже наверняка подумали, последняя команда весьма опасна. Среди ненужных файлов может затесаться пара-тройка нужных, и они удалятся вместе со всеми. Чтобы избежать этой ситуации, надо, чтобы каждый раз перед удалением запрашивалось подтверждение. Эта опция включается ключом -i: rm -i *. Тогда перед удалением каждого файла будет задаваться такой вопрос.

```
rm: удалить обычный файл “source”?
```

Ответ надо дать одним символом: y (да), n (нет).

Последняя утилита этого раздела, *shred*, выполняет функцию шредера. Как уже говорилось ранее, удаление файла заключается в удалении последней жесткой ссылки на него. Сама же информация на диске, которую содержал удаленный файл, хотя и становится недоступной, но сохраняется до тех пор, пока новый файл не запишет на это место другую информацию. Но невозможно точно предсказать, когда это произойдет. Как известно, существуют различные технологии, с помощью которых можно попытаться прочитать с диска недоступную информацию. Если она содержала коммерческую или частную тайну, то ее восстановление чревато неприятностями. Такой файл желательно не просто удалить, а измельчить на мелкие кусочки, чтобы исключить всякую возможность восстановления. Суть работы программы *shred* в том, что она перезаписывает файл несколько раз, чтобы место на диске, которое он занимал, гарантировано заполнилось мусорной информацией. По умолчанию программа перезаписывает файл три раза.

Создадим файл с какой-нибудь сугубо частной информацией.

```
cat > phone << EOF
Мой телефон +7 920 *** ** 87
EOF
```

Пройдемся по нему программой *shred* с ключом -v, чтобы программа показала нам, что делает: shred -v phone. Сообщения на экране (рис. 1) покажут, что программа совершила три прохода. Откроем теперь файл и посмотрим, что в нем теперь находится.

```
cat < phone
```

Мы видим нагромождение всевозможных читаемых и нечитаемых символов. Если же информация в файле была настолько чувствительная для вас, что вам показалось мало трех проходов, задайте большее количество проходов с помощью ключа -n:

```
shred -vn 10 phone
```

Теперь файл можно спокойно удалять: rm -v phone.

Усложним задачу

Более сложные операции над ссылками, каталогами, межпотоковыми каналами и специальными файлами выполняют утилиты *link*, *ln*, *readlink*, *unlink*, *mkdir*, *rmdir*, *mkfifo*, *mknod*.

Утилита *link* — простейший способ создать жесткую ссылку. Она не имеет ключей (кроме двух стандартных, знаете каких), и аналогична команде cp -l. Создадим файл —

```
cat > source << EOF
Первая строчка
EOF
```

и жесткую ссылку на него: link source hardlink.

Можете с помощью добавления новых строчек и вывода списка командой vdir убедиться, что получилась жесткая ссылка.

Утилита *ln* несколько сложнее. По умолчанию она также создает жесткую ссылку, а с ключом -s — символическую ссылку. Есть и другие ключи; просмотреть их поможет --help. Например, здесь присутствует уже известный нам ключ -v, дающий указание комментировать в консоли все, что происходит при работе утилиты.

```
ln -v source hardlink2
ln -sv source symlink
```

Чтобы узнать, к какому же файлу ведет данная символическая ссылка, используется третья утилита этого раздела: *readlink*.

```
readlink symlink
```

Следующая программа, *unlink*, удаляет файлы и ссылки. Она аналогична программе *rm*, но проще и не имеет ключей.

```
unlink symlink
unlink hardlink
unlink hardlink2
```

Если вас смущает название *unlink* для утилиты, удаляющей файл, то зря. Ничего удивительного здесь нет. Ведь удаление файла — это и есть удаление последней жесткой ссылки на него.

Следующая утилита, *mkdir*, позволяет создать каталог:

```
mkdir work
```

В данном случае создался каталог **work**.

Интересно, что одной командой mkdir можно сразу создать сложную структуру каталогов. В этом нам помогут символы подстановки. Несколько каталогов одного уровня перечисляются через запятую и помещаются в фигурные скобки. Если какая-то часть слова выносится за скобки, она приписывается к названию всех каталогов данного уровня. Вложенность каталогов обозначается через косую черту. Если, прежде чем создавать вложенные каталоги, нужно создать родительский, то список вложенных каталогов должен начинаться с запятой. Если название или его фрагмент содержит пробелы, оно должно заключаться в кавычки. Все это звучит, конечно, совсем сложно и непонятно. Лучше продемонстрировать это на практике. Давайте создадим сложную структуру каталогов. Предположим, что данная структура будет нужна студенту технического университета, чтобы помещать туда учебные материалы.

Сначала войдем в каталог **Документы** (чтобы не устраивать неразбериху в домашнем каталоге): cd Документы. Затем создадим структуру каталогов в соответствии со схемой рис. 2:

```
mkdir {Книги/{,Учебники,"Классика Computer Science","Ино-
странные языки"},Лекции/{,"Дискретная математика","Тео
рия автоматов",Программирование/{,"1 курс","2 курс","3
курс","4 курс»},"Теория вероятностей"},Программы/{,"Turbo
"{Pascal,C,Basic},Linux/{,Mint,Ubuntu,Fedora},ReactOS}}
```

Да, команда получилась слишком длинной, и вводить ее неудобно. Но можно такие длинные команды вводить по частям. Если ввести часть команды, а затем обратный слэш, командная оболочка будет ждать продолжения этой команды на следующей строке.

```
mkdir {Книги/ \
{,Учебники,"Классика Computer Science","Иностранные языки"}\
,Лекции/{,"Дискретная математика","Теория автоматов"\
,Программирование/{,"1 курс","2 курс","3 курс","4 курс"}\
,"Теория вероятностей"},Программы/{,"Turbo "{Pascal,C,Basic}\
,Linux/{,Mint,Ubuntu,Fedora},ReactOS}}
```

При создании каталога можно также сразу задавать доступ к нему с помощью ключа -m (или --mode). Создадим каталог **Развлечение**, для которого владельцу будут даны права на просмотр, редактирование, переименование и удаление содержимого. А членам группы и всем прочим пользователям будет отказано во всех правах.

```
mkdir -m u=rwx,g=,o= Развлечение
```

Можете проверить, что получилось, командой vdir.

Скорая помощь

Можно копировать, перемещать и удалять все файлы в данном каталоге одновременно, если в качестве параметра указать звездочку. Можно также применить операцию к нескольким файлам с одинаковым фрагментом названия, используя этот фрагмент в сочетании со звездочкой. Звездочка является в данном случае символом подстановки.

Следующая программа, *rmdir*, удаляет каталоги:

```
rmdir Развлечение
```

Но если каталог не пустой, она выведет сообщение об ошибке.

```
rmdir Книги
rmdir: не удалось удалить "Книги": каталог не пустой.
```

В этих случаях следует воспользоваться уже изученной нами утилитой *rm* с ключами -r. По умолчанию rm не удаляет каталоги, а удаляет только файлы. Но если вызвать ее с ключом -r, она осуществляет рекурсивное удаление, то есть удаляет каталоги вместе с содержимым (см. справочную систему --help).

```
rm -r Книги
```

Подобное рекурсивное удаление опасно тем, что можно случайно удалить какой-нибудь нужный файл. Поэтому лучше добавлять ключ -i, чтобы перед удалением каждого элемента запрашивалось подтверждение. Отвечать на подтверждение надо y (да) или n (нет).

```
rm -ri Лекции
rm -ri Программы
```

Две последние утилиты этого раздела, *mkfifo* и *mknod*, имеют дело с файлами других типов. В частности, рассмотрим межпотоковый канал. Межпотоковый канал связывает два потока, один из которых пишет что-то в него, другой — читает из него. Нам пока знакомы только два потока: стандартный ввод и стандартный вывод. Но уже пользуясь этими потоками, можно продемонстрировать, как работает межпотоковый канал.

Создадим межпотоковый канал **fifo1**, что и делает программа *mkfifo*:

```
mkfifo fifo1
```

Теперь вызовем команду для чтения из него в стандартный поток.

```
cat < fifo1
```

И... программа зависла. Ничего не выводится, а программа остается активной. То есть информации нет, и конца файла нет. Но это неудивительно: межпотоковый канал не хранит информацию, как файл, а ждет, когда другой поток в него что-нибудь запишет.

Откроем вторую консоль. Наберем уже известную нам команду отправки информации в файл из стандартного ввода и начнем вводить туда какие-нибудь строчки.

```
cat > fifo1
```

И что же мы видим? Мы видим (рис. 3), что в другой консоли начал появляться текст, который мы вводим! Программа *cat* ждала-ждала информации из файла, и дождалась! Теперь давайте прекратим вводить текстовые строчки, и введем сочетание Ctrl+C, с которым мы уже встречались — оно прерывает работу консольных программ. В соседней консоли сразу появилось приглашение командной оболочки. То есть прекращение отправки информации было воспринято как конец файла.

И, наконец, утилита *mknod*. Ее вызов требует одного из трех параметров: p, b, c. Вызов ее с параметром p создает межпотоковый канал, то есть аналогичен команде mkfifo.

```
mknod fifo2 p
```

А вызов с остальными двумя параметрами создает файлы другого типа — специальные файлы. Специальный файл — это интерфейс, который выглядит, как файл, но на самом деле является связью с драйвером аппаратного устройства.

Аппаратные устройства бывают символьные и блоковые. Символьные устройства обеспечивают прямую связь с потоком, вводя информацию по мере поступления и выводя ее по мере необходимости, минуя всякие буферы. Блоковые устройства сначала буферизуют информацию. Вызов утилиты *mknod* с параметром b создает специальный файл блокового устройства; вызов с параметром c создает специальный файл символьного устройства.

Кроме параметра b или c, команда mknod требует указания еще двух параметров: старшего и младшего номера устройства.

Старший номер — это номер драйвера устройства. Один драйвер может оперировать несколькими устройствами (например, в системе может быть два одинаковых жестких диска). Младший номер как раз служит для того, чтобы уникально идентифицировать конкретное устройство среди них.

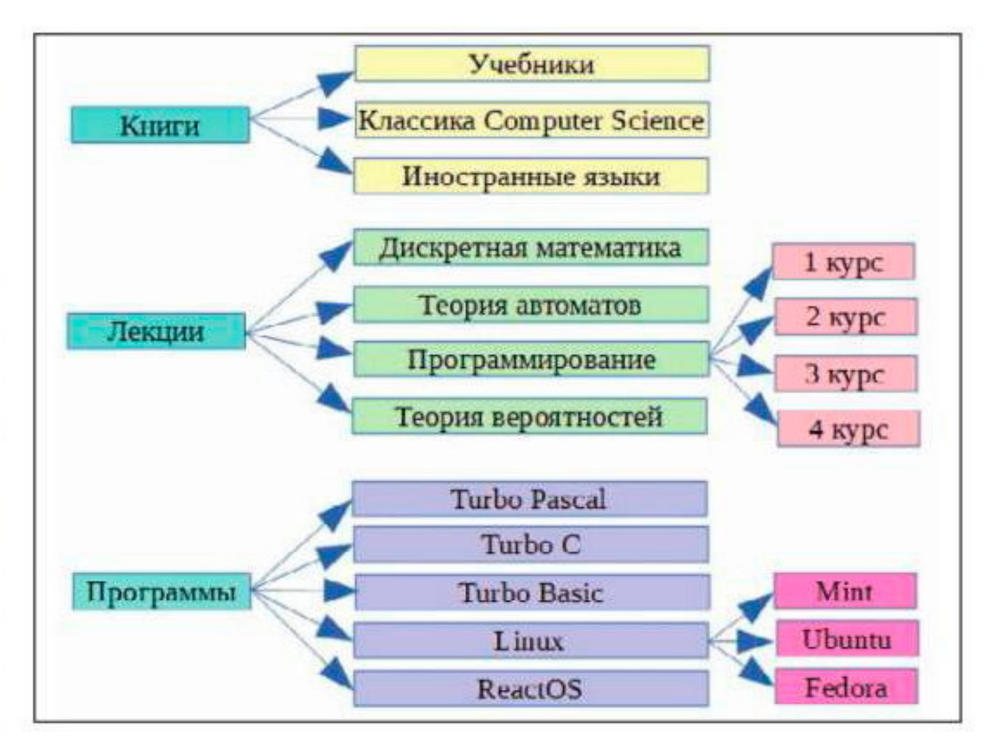


Рис. 2. Сложная структура каталогов, создаваемая одной командой mkdir.

Чтобы просмотреть, какие драйвера загружены в системе, нужно прочитать информацию из файла **/proc/devices**:

```
cat < /proc/devices
```

На экране появится список всех драйверов устройств, с указанием их номеров и типов устройств. Список разделен на две части: список символьных устройств (Character devices) и блоковых устройств (Block devices).

С двумя типами устройств мы уже встречались раньше: tty (эмулятор терминала), pts (псевдотерминал ведомый). Весь список типов устройств требует отдельного подробного рассмотрения. Пока ограничимся лишь типом sr, означающим привод CD-ROM. Давайте создадим специальный файл привода CD-ROM.

Наш список показал, что sr-устройство относится к категории блоковых, и его драйвер зарегистрирован под номером 11. Значит, в качестве типа файла надо указать b; старший номер — 11, младший номер — 0 (чтобы получить первый привод CD-ROM, имеющийся в системе). Операция выполняется под учеткой суперпользователя.

```
mknod cdrom b 11 0
```

Теперь попробуем считать информацию с этого устройства с помощью команды od. (Как вы, наверное, помните, утилита *od* предназначена для получения дампа файла.)

```
od -t x1 cdrom
```

Но у нас в приводе диск отсутствует. Поэтому ничего из специального файла не будет читаться. Должно появиться следующее сообщение:

```
od: cdrom: Носитель не найден
```

Теперь вставим диск в привод и повторим нашу команду с od. Привод зашуршал, а по экрану побежали строчки байтов. Ура! Мы получили доступ к диску. А то ли еще будет! LXF

Скорая помощь

Одной командой mkdir можно сразу создать сложную структуру каталогов, если ввести их название с применением символов подстановки.

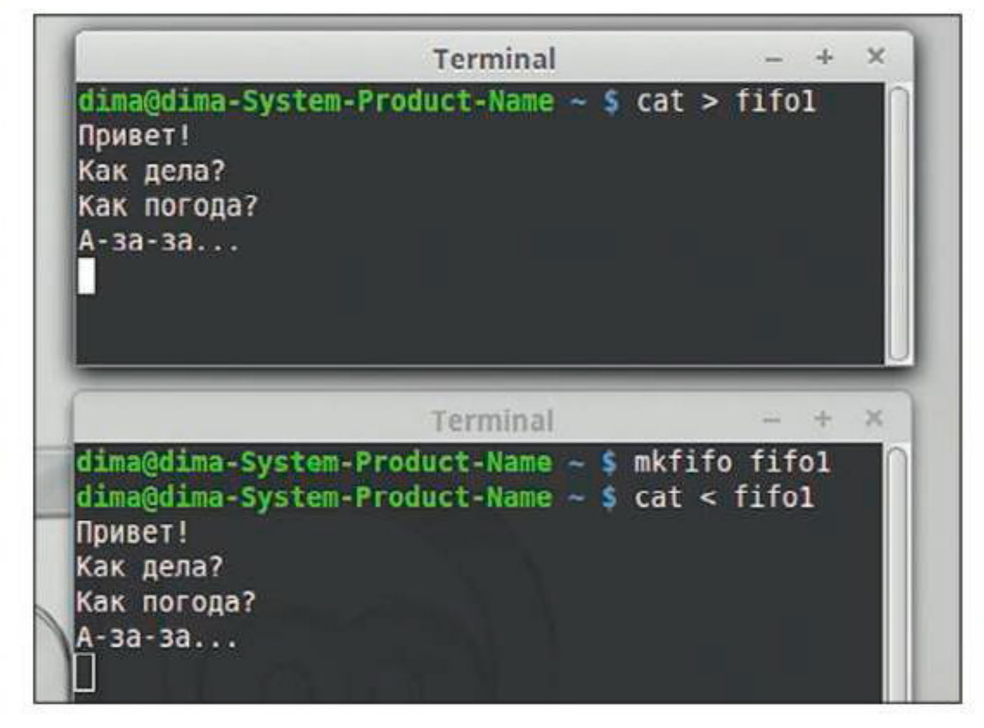


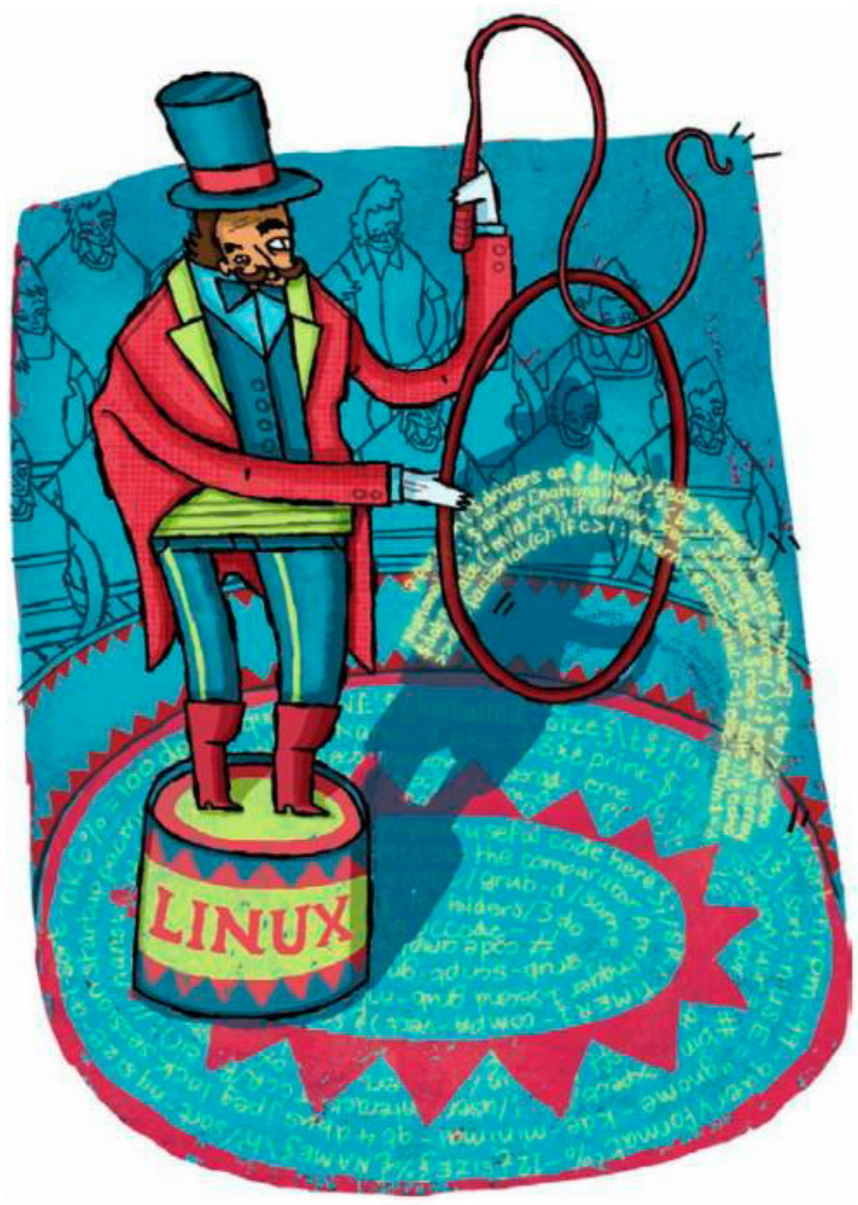
Рис. 3. Пример работы межпотокового канала.

BOINC: Мощь проектов науки

Алекс Кэмпбелл знает, как употребить ресурсы CPU и GPU вашего ПК на слежку за маленькими зелеными человечками или поиск лекарства от рака.

Наш эксперт

Алекс Кэмпбелл — технический журналист, потративший много времени на сборку компьютеров и поломку вещей. Он всё еще любит узнавать новое о Linux, и всё еще многое ломает.



Linux прекрасен в том числе и тем фактом, что люди вкладывают свое время, энтузиазм и деньги в разные проекты. Именно это придает свободному ПО с открытым кодом его мощь. Дух участия в чем-то большем, нежели сугубо личные интересы, всегда был в сердце Linux и свободного ПО.

К сожалению, не каждый может пожертвовать денег на программные проекты, и не каждый умеет кодировать, тогда как другие напридумывали всякие проекты, на которые они хотели бы получить ресурсы. К счастью, личности с благотворительными наклонностями могут сделать пожертвование, не делая абсолютно ничего. Ну, почти.

Berkeley Open Infrastructure for Network Computing (BOINC) — это ПО университета Беркли, позволяющее пожертвовать свои ресурсы CPU и GPU на такие проекты, как Поиск внеземного разума [Search for Extraterrestrial Intelligence, SETI]. Мы проведем вас по требуемым шагам, чтобы вы тоже помогали искать ответы на величайшие вопросы науки, пока вы спите. Да, это серьезно. Наш урок будет базироваться на Ubuntu Gnome 14.04. BOINC доступен в репозиториях Ubuntu, так что добавлять PPA не придется.

Графический интерфейс пользователя построен на *GTK 2*, и если вы работаете с чистой установкой KDE, вам придется скачать библиотеки *GTK*. Можно использовать BOINC без GUI, если вы хотите запустить его на сервере, не требующем X (такого, как вездесущий Raspberry Pi). Вам также захочется проверить, много ли ресурсов вы готовы передать и сколько это потребует электроэнергии. Если вы обнаружите, что высокотехнологичный CPU, на котором работает ваш компьютер Linux, потребляет слишком много и встает проблема охлаждения, можете ограничить для BOINC использование CPU (и GPU) и расписание его запуска (см. на сайте проекта BOINC, https://boinc.berkeley.edu).

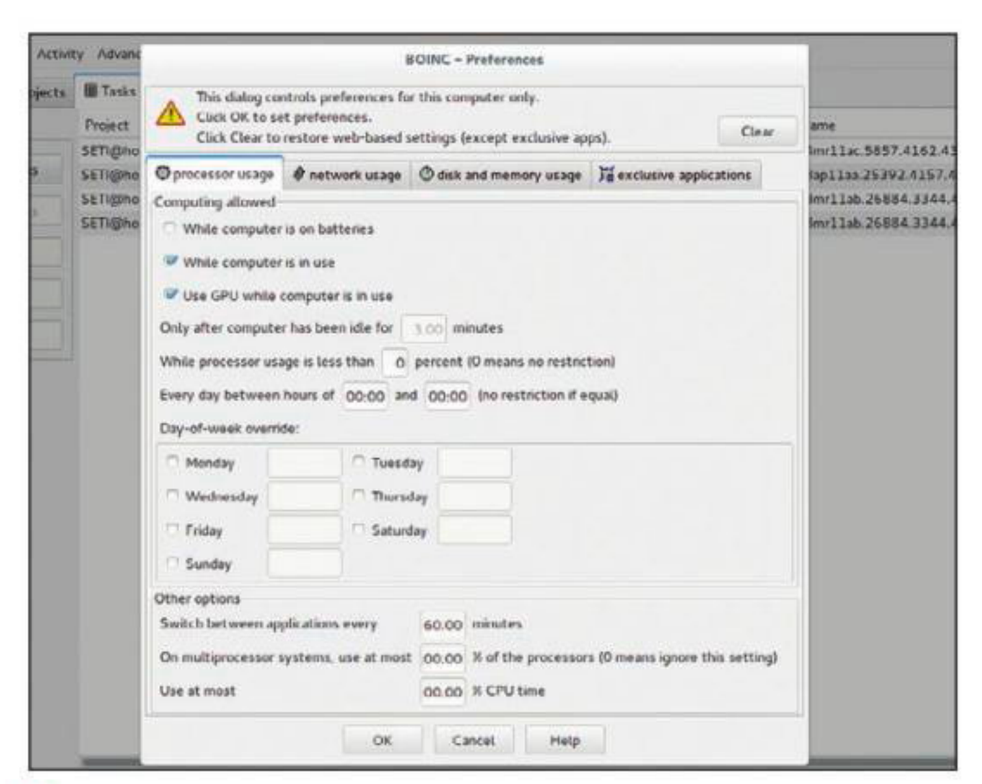
Наконец, у некоторые проектов требования выше, чем у самого BOINC. Например, некоторым может понадобиться обработка видео с возможностями OpenGL.

BOINC на Linux

Первым делом надо скачать два пакета BOINC из репозиториев Ubuntu. Просто наберите `sudo apt-get install boinccliente boinc-manager`. Отсюда Ubuntu возьмет зависимости и установит клиент и GUI. Раз плюнуть.

Если по какой-либо причине вы не можете или не хотите идти путем менеджера пакетов, можете скачать с сайта (http://boinc.berkeley.edu/download.php) скрипт оболочки, который установит вам BOINC. Проект заявляет, что программа установки скриптов не может использовать встроенные функции безопасности Linux и устанавливать BOINC в качестве демона, причем потребляет больше памяти. Короче, лучше используйте свой менеджер пакетов, если у вас нет по-настоящему серьезной причины его избегать.

Чтобы запустить менеджер BOINC, просто нажмите клавишу Super и запустите поиск по 'boinc'. Если вам нужно запустить его



› Продвинутые настройки BOINC позволяют определить расписание и распределение ресурсов.

Блестящие проекты BOINC

Самым известный проект BOINC — Berkeley Space Sciences Laboratory SETI@home, но есть множество и других прекрасных проектов:

» **ATLAS@home** от **CERN** Обсчитывает данные эксперимента ATLAS на Большом адронном коллайдере.

» **Rosetta@home Вашингтонского университета** Вычисляет структуру белка, чтобы бороться с такими страшными заболеваниями, как ВИЧ, рак и малярия.

» **Оксфордский университет** Запускает проект www.climateprediction.net, который занимается расчетом глобальных климатических моделей.

Настройки BOINC избавят вас от тягостной проблемы выбора между проектами. BOINC умеет подключаться к проектам на равные интервалы, гарантируя, что каждый из них получит от вас одинаково труда и заботы.

Любой исследователь, которому требуется много компьютерных ресурсов для решения проблемы, может создать сервер BOINC и разместить проект без особых затрат. Однако требуется наличие на своем компьютере пакета LAMP и некоторое знание технологии XML для настройки сервера и размещения проекта (подробности см. на http://boinc.berkeley.edu).

из командной строки, воспользуйтесь командой boincmgr. Она сгенерирует всплывающее окно с предложением добавить новый проект или использовать менеджер учетных записей.

Быстрый способ — выбрать опцию Add Project. Если вы пойдете этим путем, на следующем экране вам предложат выбрать проект для работы. Когда вы его выберете, BOINC обратится к серверу проекта, чтобы понять, что делать дальше. Некоторые проекты покажут экран правил использования, список требований или другую информацию. Когда мы попытались добавить эксперимент ATLAS из Большого адронного коллайдера [Large Hadron Collider] (ATLAS@home), правила использования предупредили нас, что для продолжения необходим *VirtualBox*.

Если вы согласны с условиями использования или у проекта их нет, вы увидите окно входа проекта. Каждый проект требует собственного логина и пароля. Когда вы введете имя пользователя и пароль, BOINC загрузит данные и приступит к работе.

Добавлять каждый проект вручную довольно нудно. С помощью менеджера учетных записей вы можете использовать единственный логин при старте BOINC и отслеживать прогресс нескольких проектов.

Чтобы выбрать менеджер учетных записей, просто щелкните по одной из опций в диалоге или введите URL. BOINC предлагает два сайта менеджеров учетных записей, BAM! (BOINC Account Manager) и GridRepublic. Вы можете пооткрывать оба URL в браузере, чтобы посмотреть, который из сервисов прельстит ваше воображение. Здесь мы продолжим работу с BAM!.

BOINC? BAM!

Как и индивидуальные проекты, BAM! потребует входа с именем пользователя и паролем. Чтобы создать их, посетите сайт BAM! и создайте учетную запись. Все, что нужно указать на сайте — имя пользователя, e-mail и пароль. Потрудитесь указать реальный адрес, потому что вам понадобится активировать вашу учетную запись с помощью URL, отправленного по e-mail.

Войдя с вашим именем пользователя и паролем, используйте сайт, чтобы добавлять проекты, нажав Signup for projects и затем по иконке скрепки справа от названия проекта, к которому вы хотите присоединиться. Проектов можно выбрать сколько угодно. Однако стоит знать, что в добавлении проектов есть один подводный камень: если вы добавляете проекты вручную через менеджер BOINC, ваш прогресс и учетные данные не будут впоследствии доступны в вашем менеджере учетных записей.

Все это проделав, вы можете войти в BAM! из BOINC. Вместо имени пользователя вы также можете использовать то, что в BOINC называется «слабый аутентификатор». При этом вам надо лишь ввести в качестве пароля случайную последовательность. Это позволяет другим компьютерам или людям выполнять вычисления от вашего имени, не вводя пароль. Но будьте внимательны, меняя свой пароль BAM!: слабый аутентификатор изменяется вместе с паролем. После того, как BOINC соединится с вашей учетной записью BAM!, откройте web-страницу BAM! снова.

Вам понадобится прикрепить компьютер к проекту BAM!, чтобы дать ему работу. Чтобы сделать это, нажмите на My projects в левой столбце сайта BAM!. Затем щелкните по иконке компьютера под столбцом Show hosts. Здесь выберите ПК, который вы собираетесь использовать для проекта, и нажмите Attach. При желании можно запретить проекту использовать GPU.

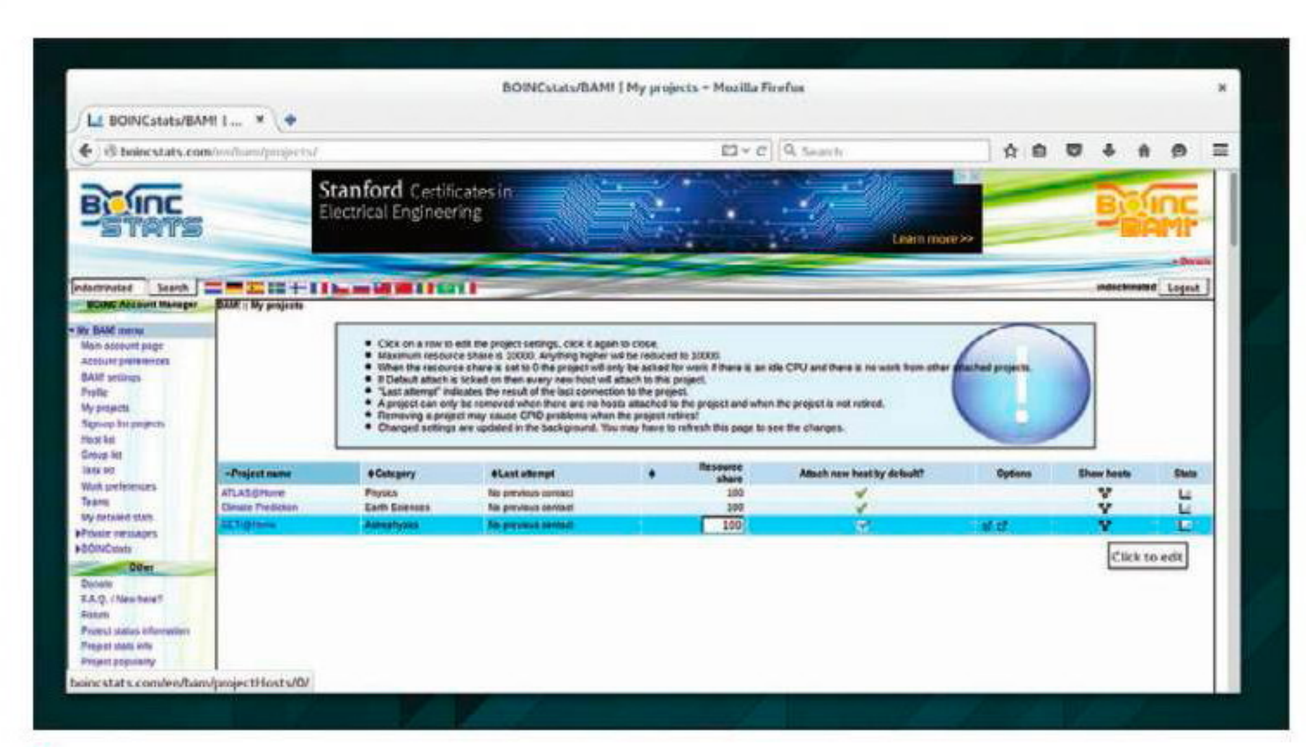
Проделав всё это, снова вернитесь к менеджеру BOINC. Нажмите кнопку Synchronise, и BOINC начнет работу в соответствии с вашими предпочтениями, которые вы сейчас установите.

Автоматизация чудес

Установка ваших предпочтений в BOINC довольно простая, но мощная. Откройте продвинутый вид в менеджере BOINC, перейдя во View > Advanced View. Здесь надо выбрать Tools > Computing Preferences.

Диалог Computing Preferences позволяет устанавливать потребление CPU, GPU, сети, диска и памяти. Он заодно позволяет устанавливать расписание работы и определять, как скоро компьютер из режима ожидания (когда вы прекращаете вводить информацию) перейдет к работе. Вы также можете установить эти значения на сайте проекта или на сайте вашего менеджера учетных записей. Любые настройки, установленные в BOINC, будут отменять настройки из Интернета.

Когда вы установите новое рабочее расписание своего ПК, ваше участие закончено; ваш ПК будет заниматься наукой в ваше отсутствие. Возможно, вам не дадут Нобелевскую премию за жертвование ресурсами вашего ПК, но, оставляя устройство включенным на всю ночь науки ради, вы вправе будете ощутить законную гордость. LXF



› **Web-интерфейс BAM! позволяет устанавливать опции для множества разных компьютеров или проектов из одной центральной панели управления.**

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

Crouton: Linux на хромбуке

Не знаете, что бы такого интересного сделать со своим модным новеньким хромбуком? **Нейл Мор** придает последнему Asus Flip облик настоящего Linux.

Наш эксперт

Нейл Мор добрую неделю пугал редакцию журнала *T3*, запуская на своем хромбуке терминалы Linux и говоря, что тьма скоро поглотит все их MacBook'и — эта публика ведь всему верит.

В последнее время на страницах *Linux Format* побывало множество хромбуков, и как бы высоко мы ни ценили основанную на Linux Chrome OS, зачастую хочется чего-то более мощного по части ПО, в духе полнофункционального Linux-дистрибутива. Если вы подумываете о покупке нового хромбука, возьмите на заметку: с помощью новой системы виртуального изменения корневого каталога [chroot] можно легко запустить Ubuntu поверх Chrome OS. Мы уже рассказывали об этом в **LXF183** [Учебники, стр. 80] и **LXF185** [стр. 50], и, честно говоря, с тех пор мало что изменилось, но теперь у нас есть последняя версия, на базе Ubuntu 14.04 (вместо устаревающей 12.04 LTS), и мы рассмотрим, как запустить ее с любой SD-карты (или USB-брелка) — дабы не занимать места на диске — а также опробуем функции двойной загрузки при запуске Arch.

Да и кто сказал, что хромбуки — штука примитивная? Ничего себе примитивная, при таком-то количестве систем GNU/Linux, доступных для установки на них!

Мы решили взять новый Asus Flip C100P: это интересный хромбук, на базе процессора ARM Rockchip, с сенсорным экраном и по розничной цене всего £250 за версию с 4 ГБ памяти — в самый раз подойдет для установки полноценного дистрибутива. Эта модель нам приглянулась потому, что с компонентом ARM случаются проблемы, которые мы рассмотрим во врезке, посвященной Arch (*см. стр. 75*); в остальном же, для Crouton это удачный вариант.

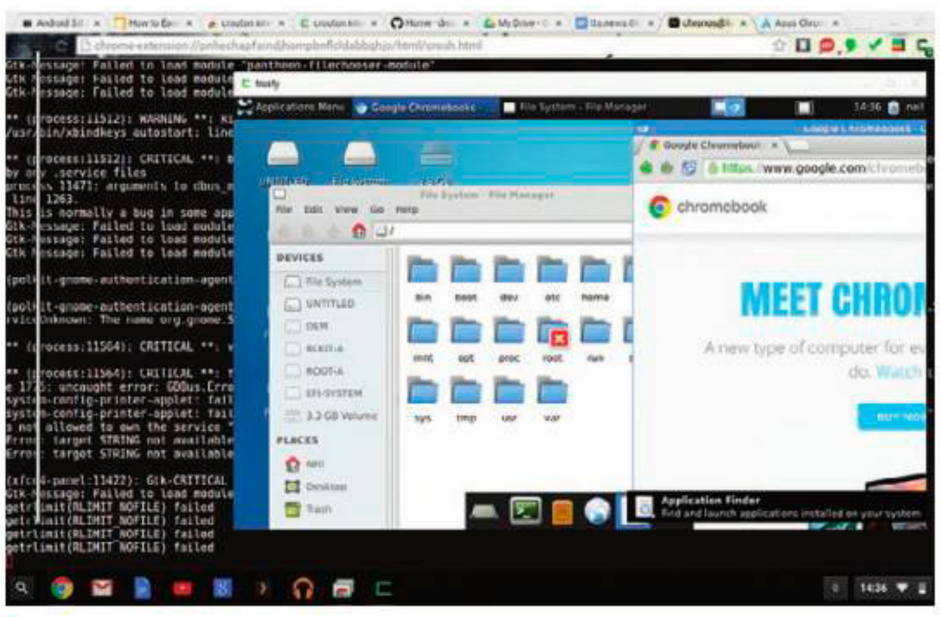
В большинстве хромбуков предпочтение отдано новомодным процессорам Intel Celeron, сейчас это Celeron 3205U. Это беспроигрышный вариант с хорошей производительностью, внушительным сроком службы аккумулятора, проверенной и стабильной архитектурой Intel, и можно не сомневаться, что программы x86-64 будут работать и поддерживаться без проблем. Стало быть, выбирать хромбук на базе ARM — сущее безумие, так? Ан нет! Давайте быстренько разберемся, почему...

Во-первых, у Ubuntu и Debian есть ARM-сборки. В репозиториях их не так много, но зато в них есть большинство самых именитых программ и доступны все привычные инструменты. Процессор ARM действительно менее мощный: наши тесты в Сравнении хромбуков [см. стр. 24 **LXF202**] показали, что ARM Asus Flip работает почти вдвое медленнее, чем Intel Celeron Acer Chromebook 15. Но так ли вам критична большая скорость? Главная проблема процессоров ARM скорее в том, что драйверы с открытым кодом порой ужасны, особенно при отсутствии 3D-ускорения.

Как мы выяснили по нашей сборке Arch, наладить нормальный рабочий стол на поставляемых драйверах практически невозможно. Даже давнему пользователю и фанату Arch Джонни не удалось запустить на нем хотя бы *startx*, ну да ладно. Запуск через Crouton — совсем другая история, хотя с некоторыми ограничениями вы всё же столкнетесь. Конечно, если вы хотите, чтобы аппаратная поддержка была на высоте, разумнее сделать выбор в пользу Intel Chromebook, однако через Crouton мы установили Ubuntu 14.04 *Xfce* с поддержкой сенсорного ввода без всяких проблем. Повезло!

Скорая помощь

Переключаться между Chrome OS и Ubuntu на ARM можно с помощью Ctrl+Alt+Shift+Back и Ctrl+Alt+Shift+Forward (на хромбуках с архитектурой x86, клавиши быстрого доступа — Ctrl+Alt+Back или Ctrl+Alt+Forward и далее Ctrl+Alt+Refresh).



› С помощью дополнения Chrome OS, Linux можно запускать в отдельном окне — буквально, что доктор прописал.

Пробуем Crouton на вкус

Прежде всего надо включить режим разработчика (если он еще не включен), который сотрет все с вашего Chromebook. Так что сделайте резервную копию, или будьте готовы к принудительной очистке системы.

Для начала отключите питание своего хромбука, а затем включите его и нажмите и удерживайте клавиши Esc и Refresh в верхнем ряду клавиатуры. Появится устрашающее уведомление; проигнорируйте его и нажмите Ctrl+d, а затем Enter, чтобы позволить ему уничтожить и переустановить Chrome OS — это занимает минут десять. Затем надо загрузить хромбук нажатием Ctrl+d (он должен загрузиться самостоятельно после 30 секунд, но кто же вытерпит *такое* долгое ожидание); как мы увидим, можно также нажать Ctrl+u, чтобы загрузиться с внешнего USB-накопителя.

Вернувшись в Chrome OS, скачайте скрипт Crouton в папку **Загрузки**, используя https://goo.gl/fd3zc. Нажмите Ctrl+Alt+t, чтобы открыть основной терминал, и с помощью команды `shell` перейдите в полную оболочку.

Crouton обычно хранит chroots в папке **/usr/local**, а нам теперь надо создать символическую ссылку отсюда в папку на SD-карте, создав измененный корневой каталог для Crouton как обычно, но всё будет храниться на внешнем накопителе. Нам повезло урвать карту MicroSD на 32 ГБ; такого объема хватит для установки любой ОС. Давайте изменим каталог, где находится chroot, используя `cd /usr/local`, и с помощью `lsblk` просмотрим список устройств и разделов на вашем устройстве. На USB ищите раздел **sda1**, а на SD-карте —

Прямо в Arch

Arch Linux можно установить на систему с возможностью двойной загрузки, хотя пока, из-за проблем с драйверами, без графического интерфейса. Включив режим разработчика, откройте терминал и введите sudo su, а затем активируйте загрузку с USB:

```
crossystem dev_boot_usb=1 dev_boot_signed_only=0
```

Перезагрузитесь, и теперь вы сможете форматировать, создать разделы и установить Arch на SD-карту. Можно и на USB-накопитель, просто замените **mmcblk1** на **sda.**

Теперь нам надо размонтировать, форматировать, добавить разделы, загрузить, извлечь, настроить и запустить Arch. Введите unmount mmcblk1, чтобы полностью очистить диск. Для создания нового раздела GPT выполните

```
fdisk /dev/mmcblk1
```

и введите g и w для создания и записи таблицы GPT. Для создания разделов используйте

```
cgpt create /dev/mmcblk1
cgpt add -i 1 -t kernel -b 8192 -s 32768 -l Kernel -S 1 -T 5 -P 10 /dev/mmcblk1
```

Чтобы добавить корневой раздел, нужно определить его объем — для этого узнайте объем всех устройств с помощью cgpt show /dev/mmcblk1. Для первого столбца укажите второе с конца число (самое большое) и вычтите 40960, а затем подставьте результат вместо XXXX:

```
cgpt add -i 2 -t data -b 40960 -s XXXX -l Root /dev/mmcblk1
```

Необходимо обновить, а затем форматировать новый раздел с помощью

```
sfdisk -R /dev/mmcblk1
mkfs.ext4 /dev/mmcblk1p2
```

Затем надо скачать и распаковать сборку Arch, выполнить cd /tmp и скачать с помощью wget http://archlinuxarm.org/os/ArchLinuxARMveyron-latest.tar.gz (если загрузка оборвется или архив будет поврежден, выполните ее заново, добавив переменную -c), выполните mkdir root и mount /dev/mmcblk1p2 root, а затем распакуйте архив, используя:

```
tar -xf ArchLinuxARM-veyron-latest.tar.gz -C root
```

Наконец, мы добавим это в раздел, демонтируем его и перезагрузимся:

```
dd if=root/boot/vmlinux.kpart of=/dev/mmcblk1p1
umount root
sync
```

Перезагрузитесь и запуститесь с флешки, используя Ctrl+u. Войдите как root, введите пароль и выйдите в Интернет с помощью wifi-menu. Теперь, имея базовую установку Arch Linux, вы можете обновить ее, используя pacman -Syu; правда, получить графический рабочий стол вам пока не удастся из-за проблем с драйверами, а значит, придется установить драйвер *X.org* и Mali EGL с помощью pacman -S xf86-video-armsocrockchipveyron-libgl. Зато дальше вы сможете делать, что пожелаете!

mmcblk1p1, и обратите внимание на имя после **/media/removable/**. Создайте новую корневую папку с помощью

```
$ sudo mkdir /media/removable/<NAME>/chroots
```

Затем создайте символическую ссылку, используя

```
$ sudo ln -s /media/removable/<NAME>/chroots/ chroots
```

Теперь вы можете посредством Crouton установить Ubuntu в папку с новой символической ссылкой. Мы могли бы использовать Ubuntu Trusty Tar 14.04, но там нет 3D-ускорения, поэтому понадобился интерфейс *Xfce*, чтобы испытать сенсорные возможности Asus Flip. Вы можете добавить шифрование, вставив переменную -e сразу после sh, но возможно, стоит сначала протестировать все без него:

```
$ sudo sh ~/Downloads/crouton -r trusty -t touch,xfce
```

Скачав Ubuntu, вы можете войти в chroot и запустить его с помощью этих двух команд:

```
$ sudo enter-chroot
$ startxfce4
```

Затем выйдите оттуда и вернитесь в Chrome OS. Можно переключаться между Ubuntu и Chrome OS, используя ярлыки, описанные во врезке *Скорая помощь* (слева); мы также рассмотрим, как расширить возможности chroot, чтобы запускать его в окне Chrome OS, а также переключаться между полноэкранным и оконным режимом.

Удалить chroot можно с помощью sudo delete-chroot<имя chroot>, имя по умолчанию — это имя релиза, т.е. trusty. Посмотреть список доступных релизов (разных дистрибутивов), а также список доступных целей (рабочих столов) можно с помощью приведенных ниже команд. Следует помнить: с устройствами ARM, наподобие Flip Unity, 2D доступен только в Ubuntu 12.04, поскольку в процессоре нет поддержки EGL.

```
$ sh ~/Downloads/crouton -r list
$ sh ~/Downloads/crouton -t help
```

Как поддерживать ваш Crouton и chroot в актуальном состоянии? Чтобы проверить обновления, скачайте последнюю версию и посмотрите, что в ней нового, с помощью croutonversion -u -d -c из chroot (выполните croutonversion -h, чтобы узнать, за что конкретно отвечают данные параметры). Выйдите из chroot и выполните sudo sh ~/Downloads/crouton -u -n <chrootname>, чтобы обновить все установленные цели.

Заметили, что часто используете chroot? Вероятно, не мешает сделать резервную копию; легко это сделать, а также восстановить данные поможет функция *edit-chroot*. По умолчанию она находится в папке **~/Download**, но используя переменную -f, вы можете добавить путь, вот так:

```
$ sudo edit-chroot -f /media/removable/<имя карты> -b <имя chroot>
```

Для восстановления просто замените -b на -r.

Наконец, можно установить с сайта https://goo.gl/OVQOEt дополнение, позволяющее использовать общий буфер обмена и расширение окна. Добавив его в самоё Chrome OS, используйте эту команду, чтобы обновить chroot Trusty для целей extension и xiwi:

```
$ sudo sh ~/Downloads/crouton -r trusty -u -t extension,xiwi
```

Чтобы сделать общим буфер обмена, установите в chroot-версию Ubuntu браузер *Chromium* и используйте вышеупомянутую ссылку, чтобы добавить в нее то же дополнение. После этого перезапустите хромбук — и вы увидите, что chroot запускается в окне Chrome OS и у них теперь единый буфер обмена. Очень удобно! LXF

› Возможно, вам удастся запустить только *Xfce*, но мы и тому были рады: ведь так вы получаете доступ ко всему богатству программ Linux, которых нет на Chrome OS.

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

ITSM-процессы: Платформа Linux

Сергей Коклеев и **Юрий Малолетко** оценивают использование Linux как платформы поддержки и управления для процессов, протекающих в ИТ.

Наш эксперт

Сергей Коклеев — генеральный директор «Мегаполис Профи». Почти 20 лет занимается укрощением и приручением ИТ-процессов. Знает о них все, чем и делится с коллегами и слушателями.

Наш эксперт

Юрий Малолетко руководит отделом внедрения «Мегаполис Профи» и разговаривает с «железом» на одном языке. Умеет заставить железку нести пользу людям.

Мы рассмотрим реализацию управления ИТ-сервисами (ITSM, IT Service Management). В терминологии ITIL (IT Infrastructure Library — библиотека инфраструктуры информационных технологий, описывающая лучшие практики организации работы подразделений или компаний), нас интересуют Управление инцидентами с функцией Сервис Деск [Service Desk], управление событиями и запросами на обслуживание.

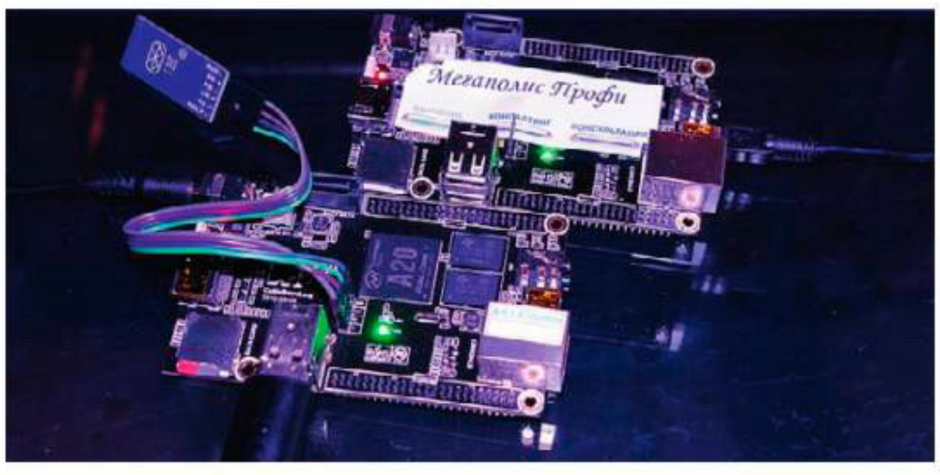
Настроенные платы компании «Мегаполис Профи» с предустановленным ПО готовы для коммерческого применения и уже более года работают в ИТ-компаниях России и Казахстана. Отзывы пока только положительные. Тех, кому интересна тема управленческих процессов в ИТ, описанных в библиотеке ITIL, и методологии аудита COBIT5, построения бизнес-диаграмм OBASHI и управления проектами PRINCE2, мы приглашаем на курсы: см. megapolis-profi.ru.

Система поддержки пользователей

Итак, для сегодняшнего постного блюда — «Система Поддержки Пользователей (СПП)» — понадобятся: ваш любимый дистрибутив Linux; *GLPI* — Система регистрации обращений пользователей; *OCS Inventory NG* — Система управления конфигурациями. Дополнительно — почтовый сервер *Postfix*, сервер баз данных *MySQL*, web-сервер *Apache2*; сгодится любое другое ПО со сходной функциональностью, на ваш выбор. Мы расскажем, как задокументировать деятельность ИТ, а листинги командной строки и споры о том, кто сильнее — Слон или Кит, оставим за кадром. Естественно, все программы должны иметь свободную лицензию, быть достаточно свежими и распространяться с открытым исходным кодом — иначе оценить безопасность применения ПО весьма сложно, тем более, когда клиентскую часть, как в нашем случае системы сбора конфигураций, предполагается ставить на весь наличный парк ПК.

Разбор корзины продуктов

Дистрибутив — это скомпилированный под ARM-архитектуру (т.к. установка будет проводиться на плату CubieBoard2) серверный Debian GNU/Linux 7 (Wheezy). Дистрибутивы с графическим интерфейсом тоже прекрасно работают на этой плате, имеющей встроенный ускоритель графики. Выбор платы обусловлен высокими требованиями к производительности при низком потреблении электроэнергии. Нашему учебному центру понадобилось компактное устройство, работающее «из коробки», и CubieBoard2 выиграл по всем параметрам. Устройство эксплуатируется уже более двух лет и очень востребовано преподавателями, далеко не линуксоидами. Особенно легко, по их оценкам, проходит сохранение, восстановление и «откат» к начальным установкам: надо всего лишь вставить карту памяти в ноутбук, запустить программу копирования ISO-образов и выбрать нужный файл образа и диск. С буквой диска, бывает, ошибаются при выборе, так что лучше отключить все другие съемные носители, кроме необходимого.



› Рис. 1. Так выглядит наше устройство без корпуса.

На рис. 1 представлены наши платы, извлеченные из корпуса. В кадре виден некий адаптер — это BT-HR6, отличное решение взамен провода COM-порта при начальной конфигурации, пока не настроен SSH-сервер. Устройство настраивается AT-командами, далее идет передача данных прямо с COM-порта, к которому подключен адаптер. Программ BT-терминалов под Android масса. В настольных ОС при соединении с бортовым адаптером устройство видно как обычный BT COM-порт. Очень удобно для любых устройств, конфигурируемых по UART, включая коммутаторы и другое активное оборудование. Конечно, помните про безопасность: использовав устройство, отключите его. Список команд настройки BT-HR6 есть на wiki.pinguino.cc/index.php/SPP_Bluetooth_Modules.

Система регистрации обращений

Решив автоматизировать процессы ITSM на основе методологии ITIL, начнем с самого актуального: регистрации обращений пользователей. Управление инцидентами — самый востребованный ITIL-процесс, интересный всем, кто сталкивается с поддержкой ИТ.

Система регистрации обращений — *GLPI*, про нее см. на www.glpi-project.org. Сейчас доступна для загрузки версия 0.85.4. «Падений и зависаний» либо ошибок в работе с базой данных не замечено. Проект активно развивается, в списке корпоративных пользователей на сайте — крупнейшие организации. *GLPI* был выбран в т.ч. и благодаря русскому интерфейсу «из коробки» —это важно для нашего конечного пользователя. Ну и приятно удивили легкость и беспроблемность подключения к MS Active Directory по протоколу Kerberos. Возможна авторизация пользователей с любой реализацией LDAP-каталога, внутренней базы, встроенного механизма PAN Linux. А авторизация почтового сервера IMAP через защищенный протокол — идеальное решение для организаций, не использующих доменную авторизацию рабочих станций. Покажем и другую полезную функциональность. Делаем настоящий программно-аппаратный комплекс!

Система управления конфигурациями

При регистрации запросов на изменения или связанных с инцидентами идеально знать оборудование, установленное у пользователя.

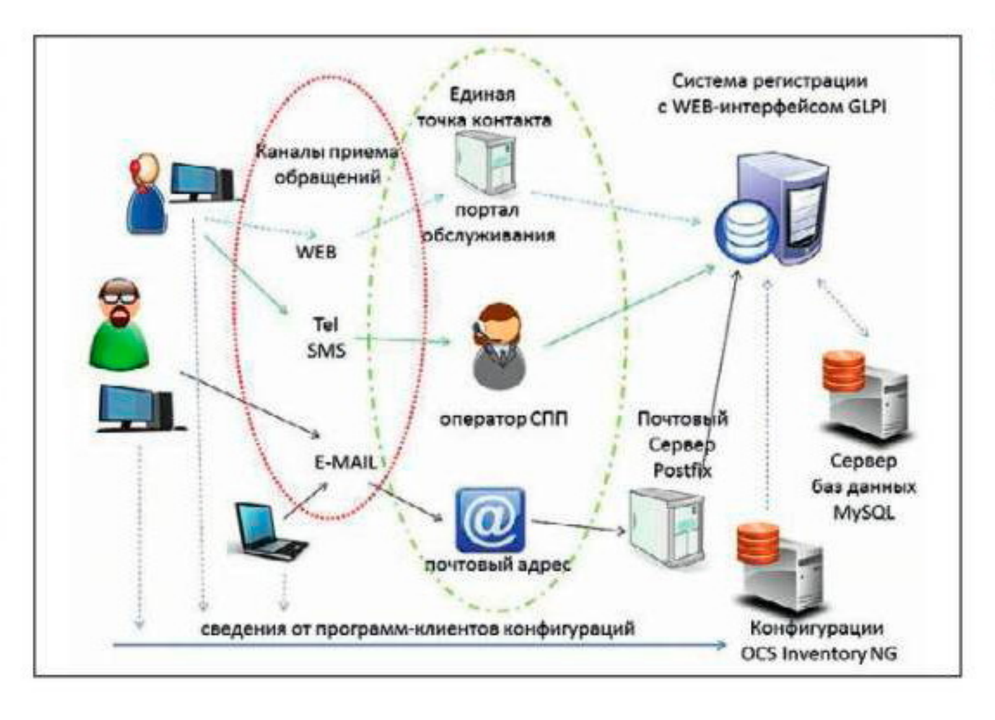
Да и всегда хочется контролировать конфигурацию своей инфраструктуры. Для работы в связке с *GLPI* выбрана система управления конфигурациями *OCS Inventory NG* (www.ocsinventory-ng.org/en). Легкая в работе и инсталляции система нетребовательна к ресурсам и построена на клиент–серверной архитектуре; клиенты есть для всех распространенных систем, в т.ч. мобильных. Также налицо русский язык интерфейса «из коробки». Двумя щелчками мыши в web-интерфейсе настраивается передача и синхронизация отслеживаемых данных о конфигурациях с *GLPI*, т.е. система получит все требуемые данные о рабочем месте пользователя, включая перечень его ПО, уже при регистрации обращения. Это часто экономит время при решении различного рода инцидентов.

Дополнительные приложения

Попробуем дополнить функциональность нашего устройства. Дополнительные приложения: почтовый сервер *Postfix* (www.postfix.org) в конфигурации с виртуальными доменами, *Dovecot* (www.dovecot.org) как IMAP-клиент, Roundcube (roundcube.net) как web-интерфейс к почтовому серверу. Администрировать виртуальные домены мы будем с помощью PostfixAdmin (postfixadmin.sourceforge.net). В качестве базы данных настроим *MySQL* (www.mysql.com) как систему, поддерживаемую всеми задействованными приложениями.

Рисуем в DIA

Начнем со схемы работы нашей системы. Мы взяли редактор диаграмм *Dia* (http://dia-installer.de), но подойдет любой другой со схожей функциональностью. Русский язык тут готов «из коробки». Если взглянуть на ценники платных аналогов, из огромных меню которых используется 10% и то, наверное, только «художниками»-профессионалами, *Dia* вам понравится — даже при том, что для редактирования ранее введенного текста придется жать клавишу F2 и запускать *Dia* в режиме совместимости Windows XP. Для ясности начнем с визуализации желаний. Общая схема процессов СПП — на рис. 2. Идея ясна из рисунка; приступим к реализации.



› Рис. 2. Схема работы СПП.

Система

Коротко, по шагам, опишем схему установки Debian на CubieBoard2. Для других дистрибутивов и «железа» установка может значительно отличаться, но вы, как профессионалы, точно справитесь. Скачаем дистрибутив, с помощью утилиты *Win32DiskImager* в Windows или *dd* в Linux «зальем» образ системы на карту памяти или USB-флешку. Учтите, что современная карта памяти работает в разы быстрее USB-флешки; надежность для наших задач также достаточная. Главное — не забывать вовремя делать резервные копии, про эту процедуру расскажем отдельно. После успешного старта — обновим систему, производем необходимые настройки сети.

Инсталляция GLPI

Мы устанавливали стандартный пакет из дистрибутива, с последующим обновлением до последней стабильной версии с сайта разработчика. Обновление сводится к перезаписи каталога установки файлами, полученными из распакованного дистрибутива.

При установке в Debian система устанавливает все зависимые пакеты самостоятельно, включая PHP, на котором написан интерфейс *GLPI*, *Apache2*, *MySQL*. Для других дистрибутивов могут быть отличия, но мы уверены: профессионалы легко разберутся с установкой. Для дистрибутивов, куда не входит пакет, на сайте разработчика представлена отличная документация. О огромное количество информации имеется во Всемирной сети.

Инвентаризация и контроль оборудования

Для корректной работы по предоставлению качественных услуг поддержки необходимо иметь четкое понимание и представление о том, что конкретно находится в нашей инфраструктуре. Это относится в равной степени и к оборудованию, и к программному обеспечению. Нами был выбрано ПО *OCS Inventory NG*, по следующим положительным характеристикам: мультиплатформенность; открытый код; интеграция на уровне БД с другим ПО; возможность поддержки производителем; нетребовательность к ресурсам.

Из отрицательных сторон можем указать только то, что нет документации на русском языке и надо устанавливать клиент, что не всегда допустимо в компаниях с точки зрения информационной безопасности. Но давайте настроим эту систему.

OCS Inventory NG — сервер

Следующим этапом устанавливается *OCS Inventory NG*. Обычно проблем с этим не возникает. Возможно, придется установить некоторые модули для *Apache2* в «ручном» режиме. Это зависит от дистрибутива и комплекта установки web-сервера.

Почтовые адреса

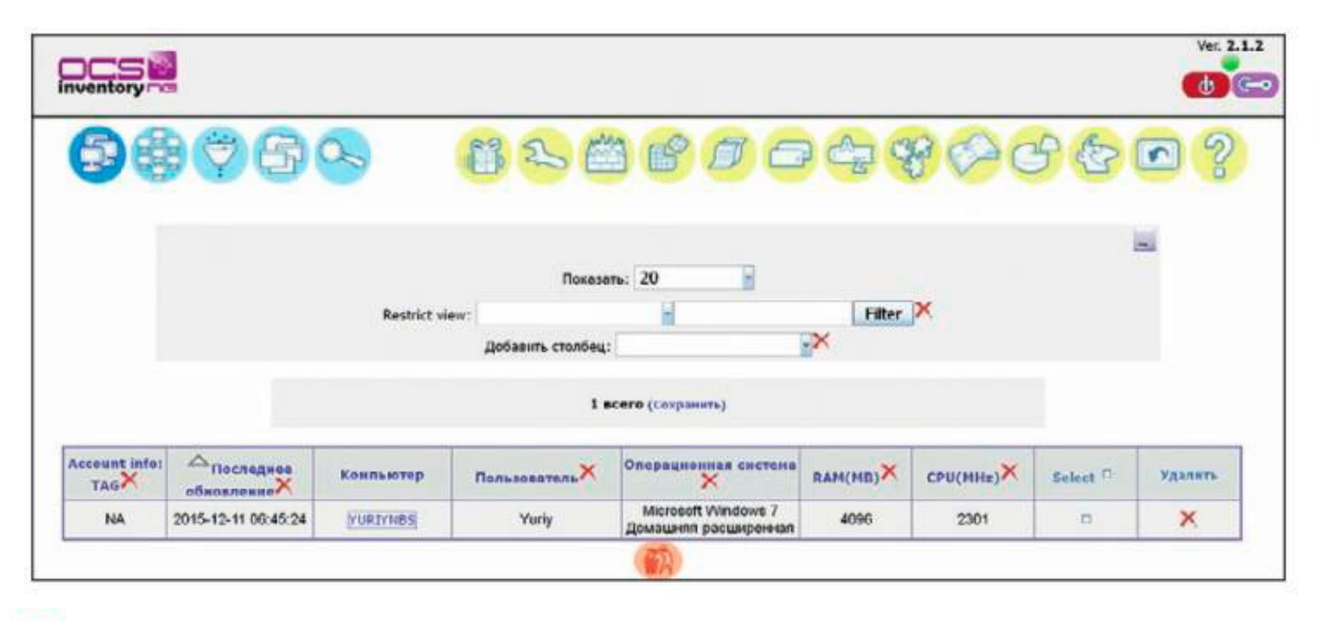
Далее, создаются почтовые адреса — приемники почты для *GLPI*, нечто вида sd@test.local, mng@test.local, supp1@test.local. Желаете использовать собственный почтовый сервер — устанавливается *Postfix* или любой другой почтовый агент, лучше с поддержкой виртуальных доменов и web-интерфейсом. Подробно процедуру установки почтовика расписывать незачем: в Сети это всё есть.

OCS Inventory NG — агент

Ура! Все запускается. Далее, расставляем клиент *OCS Inventory NG* на устройства «жертв» — пользователей. Это нам даст, после первого получения отчетности, полное представление о используемом оборудовании и ПО. Для Win-клиентов были протестированы все способы установки клиента, описанные на сайте производителя ПО. Надо отметить, что все они оказались рабочими. Нами был выбран способ установки путем «сборки» установочного пакета .msi, как наиболее управляемый и позволяющий устанавливать обновлять и удалять приложение-клиент стандартными средствами MS Active Directory. Плюс к этому, при ручной установке на компьютеры, не входящие в домен, либо самостоятельной установке пользователем файла клиента, скачанного с сайта, не понадобится указывать адрес и порт сервера *OCS Inventory NG*. Так что рекомендуем именно этот способ, как наиболее подходящий. Подробное описание процедуры инсталляции с помощью пакета *.msi, описанное разработчиками ПО, доступно по ссылке http://wiki.ocsinventory-ng.org/index.php/Howtos:CreateMsiOCSAgent.

На рис. 3 — пример системы, которая показывает основные характеристики рабочего места пользователя. Более подробные сведения по каждому разделу разнесены по вкладкам (рис. 4).

Итак, информация об инфраструктуре вполне достаточная. Практика показывает, что это очень полезное приложение: оно следит не только за ситуацией с серверным оборудованием, но и за рабочими местами пользователей, где иногда «случайно» пропадают линейка памяти или хорошая видеокарта. »



› **Рис. 3. Окно *OCS Inventory* с общими параметрами пользователя.**

GLPI

Проверив правильность собираемых данных, прямо в web-интерфейсе *OCS Inventory NG* можно настроить нашу основную программу — *GLPI*. Для запуска и начала работы достаточно указать единственный адрес электронной почты, куда будут приходить запросы пользователей. Он должен быть заранее оттестирован в почтовике или средствами web-интерфейса почтового сервера.

Пропишем e-mail-адрес единой точки контакта и настройки почтового протокола в меню *GLPI* Настройки > Приемники и протестируем. Убедившись в работоспособности почты у оператора, создаем требуемых пользователей *GLPI*, назначаем им почтовые адреса. Пробуем создать в системе документ — Заявка. Если все сделано правильно, учетная запись, указанная в поле Назначено, и лицо, указанное в поле Заказчик, получат соответствующее сообщение. Так же, по e-mail, высылается оповещение о любом изменении статуса вашей Заявки. Если это произошло — настройка выполнена успешно. Настроим связь между *GLPI* и *OCS Inventory NG*. Для этого надо установить плагин, с одноименным названием, доступный на сайте *GLPI* в разделе Дополнения. Установка плагина сводится к распаковке скачанного с сайта архива в каталог установки **GLPI\plugins**. Во избежание бессонных ночей, следует обращать внимание на совместимость версии плагина и версии GLPI.

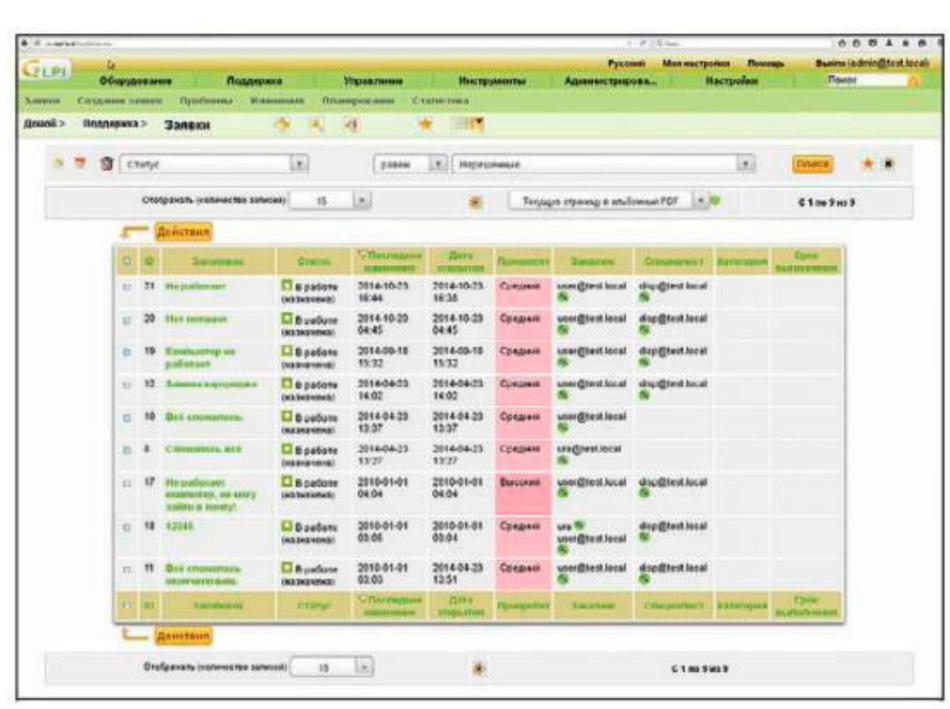
Если все заработало, почта принимается и отправляется, заявки создаются, оборудование инвентаризируется — отлично! Система, называемая в терминах ITIL «Сервис-Деск», уже работает. Добавим пару инженеров и обзаведемся пользователями.

Процесс

Под «процессом»мы понимаем процесс управления, имеющий вход, выход и набор управляющих воздействий — триггеров. В ходе выполнения процесс взаимодействует с другими процессами в заранее определенном формате.

» **Управление Инцидентами** — здесь собираются все обращения, связанные со сбоями в работе всего, что может поддерживать ИТ, и не только. Классифицируются обращения данного типа по простоиу алгоритму: работало и перестало — регистрируем инцидент и идем к вам. Внешне это выглядит, как показано на рис. 5.

» **Управление Запросами обслуживания** — процедура классификации этого типа обращений обычно начинается словами пользователя «Мне бы хотелось...» или просто «Срочно придите убрать эту радиоактивную пыль с моей клавиатуры». Ключевые фразы помогают определить тип входящего запроса. Это все обращения, по которым что-то надо или, напротив, не надо делать, не связанные со сбоем в работе сервисов.



› **Рис. 5. Окно контроля заявок в системе GLPI.**

» **Запросы на изменение** — тут все проще. Если вас что-то просят поменять — работу, например, или «постоянно зависающий компьютер» — смело классифицируйте как запрос на изменение. Обычно такие запросы связаны с управлением конфигурациями и релизами; мы, возможно, рассмотрим их в другой статье.

Рисуем процессы

Справившись с настройкой ПО, немного отдохнем, т.е. сменим деятельность: порисуем диаграммы, пользуясь программой *Dia*. Для примера приведем готовую схему наших процессов, в «слегка» упрощенном виде (рис. 6). Она напоминает задание «пройди лабиринт» из популярного советского детского журнала 1980-х гг., но на самом деле это очень эффективный бизнес-инструмент, пришедший к нам с западной управленческой литературой 1990-х. Читатели журнала без затруднений составят и опишут подобную схему. Это реальный, только очень сокращенный и сжатый пример схемы процесса. Если справиться с написанием процесса не удается, рекомендуем книги ITIL, а именно том *Service Operation* или *Service Support* (есть перевод на русский язык); еще один вариант — заглянуть в учебный центр «Мегаполис Профи», где можно подробно узнать про процессы и их реализацию на практике.

Таблица процедур

Схема процесса составлена, надо двигаться дальше. Любой процесс, как мы уже говорили ранее, состоит из процедур и функций. Заполним таблицу, описывающую процедуры. Это удобно делать, пронумеровав схему процесса, как на рисунке.

Отчетность

Любой управленческий процесс немыслим без красивых графиков, тянущихся вверх, и мелких, как бы незначительных, циферок под ними. Наши процессы не должны быть исключением. Очень эффективно, после немного нудных объяснений руководителю или заказчику о том, как у нас все хорошо или, напротив, плохо, показать красивые картинки графиков. Обычно на их лицах появляется серьезность, и можно услышать что-то типа «ну вот, другое дело, а я уж подумал, вы тут мой бюджет только пилите». Причем — независимо от показателей, к которым относятся графики. И этот мощный управленческий инструмент есть в *GLPI*: вы сможете строить вполне функциональные и показательные отчеты.

Контрольные точки

Часто можно встретить такую модную аббревиатуру, как KPI (Key Performance Indicators, ключевые показатели эффективности).

› **Рис. 4. Окно *OCS Inventory* с расширенной информацией о пользователе.**

Перечень процедур

№ п.п	Наименование процедуры	Описание
1	Вход процесса	На вход процесса, по одному из поддерживаемых каналов приема обращений, поступает запрос пользователя. Это может быть, к примеру: телефон, e-mail или любой другой удобный нам и пользователю способ.
2	Обращение	В программе GLPI формируется документ — «обращение», заполняются обязательные поля и другие атрибуты.
3-4-5	Классификация	Процедуры классификации обращения.
6-7-8	Тип	Присвоение типа входящего обращения, в зависимости от результатов выполнения процедуры классификации.
9	Выполнение работ	Этап выполнения работ.
10	Внешние процессы	Взаимодействие с внешними процессами.
11	Выполнено	Присвоение обращению статуса «Выполнено».
12	Запрос подтверждения	Запрос подтверждения о выполнении работ по обращению от пользователя.
13	Закрытие	Процедура закрытия обращения, после получения подтверждения о выполнении от пользователя.
14	Закрытое обращение	
15	Выход	Окончание работы процесса, выход.

На самом деле, это контрольные точки — просто некие величины, обычно измеримые численно, по которым можно судить о деятельности процесса. Например, «количество зарегистрированных инцидентов в день» — это просто число для статистики, а вот «количество зарегистрированных инцидентов в день — не менее 98% от общего количества обращений» — это уже и есть контрольная точка. Если больше — разбираемся, почему. Часто бывают ситуации, когда неправильно заданная контрольная точка «срабатывает» очень редко или вообще никогда. Это является поводом к пересмотру контрольной точки или доработкам внутри процесса для достижения заданного параметра.

Триггеры

В определении процесса упоминалась некая сущность под названием триггер. Возникает резонный вопрос: а где он на картинке? В данном случае, например, триггером является подтверждение пользователем выполнения поданной им заявки. Пока пользователь не подтвердит выполнение, заявка находится в работе, и при каждом отказе возвращается на этап выполнения работ, а переходит в процедуру закрытия только после получения подтверждения от пользователя.

Комплект процессной документации

Хороший комплект процессной документации, конечно же, должен включать более расширенное и детальное описание всех процедур и функций, а схема процесса едва уместится на нескольких листах формата А3. Но это будет уже совсем другая история, которая в одной статье не уложится, да и для понимания лучше посетить курсы. Здесь мы говорим про автоматизацию процессов.

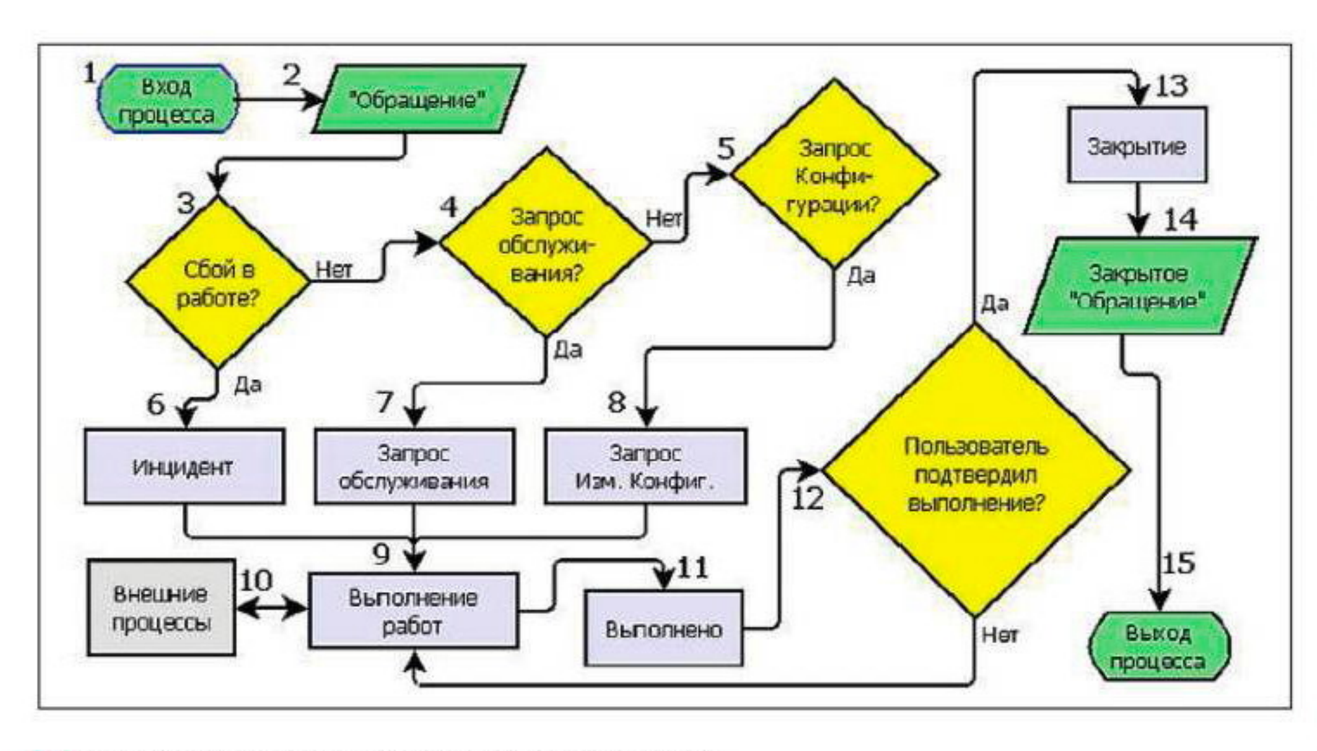
Итак, нарисованных схем и описаний процедур процесса поддержки, как ни странно, будет достаточно, чтобы дотянуться до честного 2-го уровня зрелости ИТ-процессов, в случае проведения аудита по COBIT. Всего по методологии COBIT5 таких уровней 6 (с нулевого по пятый).

ZABBIX

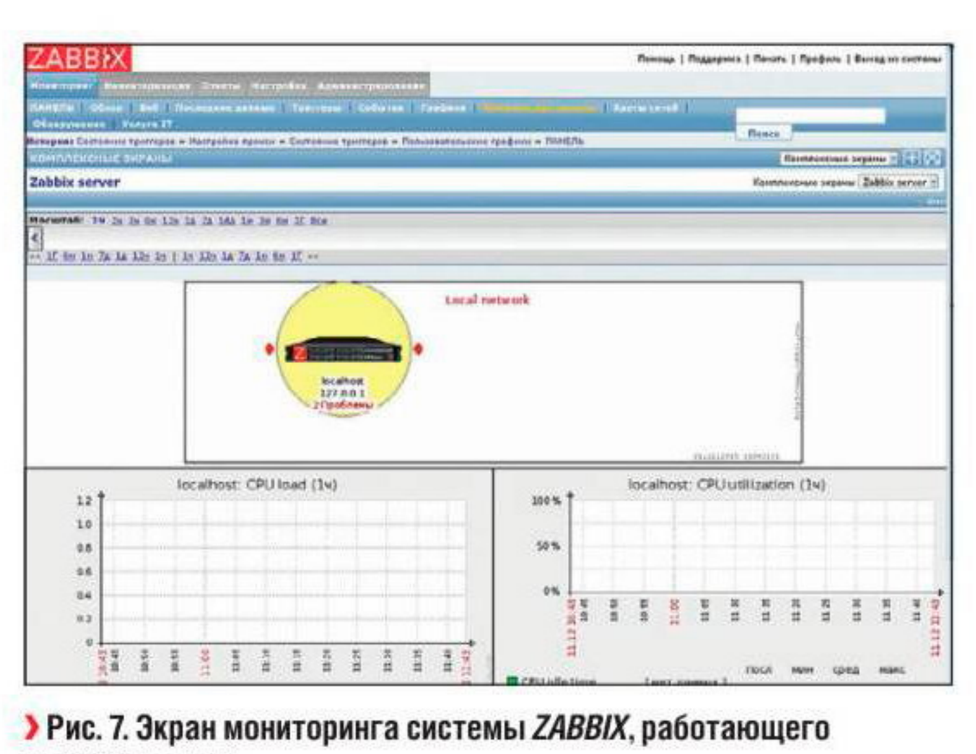
В наш обзор не вошли системы мониторинга — из-за высоких требованиями к быстродействию «железа», в основном по части дисковой подсистемы. Но нам удалось инсталлировать на CubieBoard2, имеющей на борту 2 процессора и 1 ГБ памяти, систему мониторинга *ZABBIX* последней версии (сайт с описанием продукта — http://www.zabbix.com/ru). Сократив число невостребованных «триггеров», используемых для мониторинга, и частоту опросов (до раза в 15 с) на трех опрашиваемых серверах и одном активном сетевом устройстве по протоколу SNMP, мы достигли приемлемой скорости работы. Причем у нас была не очень скоростная SD-карта памяти, с тестовой скоростью записи 15 Мб/сек. С внешним HDD или более быстрой картой памяти вполне возможна эксплуатация микро-ПК такого класса как сервера мониторинга по нескольким параметрам 3–5 серверов и нескольких активных устройств по SNMP и критически важных рабочих станций. Визуально мониторинг выглядит на этом устройстве, как показано на рис. 7.

Заканчивая наш урок, мы хотим пожелать читателям журнала успехов в освоении процессного подхода в управлении ИТ и внедрении систем на базе ITSM. Научиться самостоятельно можно всему, но время жизни не бесконечно. Начните разрабатывать свои системы управления уже сейчас, и пусть они будут строиться на вашей любимой ОС, вашем любимом железе и продуктах, написанных людьми, любящими свое дело и ценящими свое время.

Надеемся, читателям интересна изложенная тематика. Если это так, в новых статьях мы расскажем о своем опыте организации процессов управления релизами, непрерывностью и доступностью, для автоматизации которых, естественно, будет использовано свободное ПО. LXF



› Рис. 6. Схема процесса управления инцидентами.



› Рис. 7. Экран мониторинга системы *ZABBIX*, работающего на CubieBoard2.

Система: Сеть и протокол UDP

В последней части серии статей **доктора Криса Брауна** мы исследуем модель без подключения и сетевое программирование с использованием UDP.

Наш эксперт

Доктор Крис Браун обучает, пишет и консультирует по Linux. Ученая степень по физике элементарных частиц ему в этом совсем не помогает. К сожалению, как вы узнаете позже, это его последняя статья для *Linux Format*.

Скорая помощь

Учтите, что плохо настроенный сервер TFTP способен быть весьма небезопасным. Обеспечивайте его запуск от имени непривилегированного пользователя, например, 'nobody', то есть, по сути, на нем могут размещаться только файлы, доступные для чтения всем. Сервер также можно настроить на доставку файлов только из конкретного каталога (обычно это **/tftpboot**); обязательно воспользуйтесь этой возможностью.

В прошлом месяце мы изучали применение сокетов API для написания ориентированных на подключение клиентов и серверов, использующих протокол TCP. В этом месяце мы рассмотрим модель без подключения, использующую протокол UDP. Что означает «без подключения»? Ну, с точки зрения программиста это означает, что в коде нет шагов, устанавливающих подключение — ни одного вызова методов connect(), listen() или accept(), которые мы видели в **LXF203**. Клиент просто формирует датаграмму, цепляет к ней адрес получателя и отправляет. С точки зрения сервера, сервер на основе UDP иногда можно представить себе как корзину с надписью «Порт 69» или аналогичной. Любой клиент может бросить датаграмму в эту корзину. Сервер на основе UDP может получать сообщения от нескольких клиентов в произвольном порядке по очереди (как показывает моя схема последовательности операций).

Если вы исследуете поток пакетов в сети, также станет ясно, что сервисы на основе UDP в некотором смысле проще сервисов на основе TCP, потому что в них нет обмена пакетами для установки соединения и потом для его разрыва. То есть, например, если программа на компьютере A хочет отправить десяток-другой байтов данных программе на компьютере B, с точки зрения объема данных, передаваемых через сеть, гораздо эффективнее быстро передать датаграмму UDP, а не девять сообщений, требуемых для обмена по TCP (три для установки соединения, одно для отправки данных, одно для подтверждения и четыре — для закрытия соединения).

Конечно, у экономности UDP есть недостаток: нет гарантии доставки. Датаграмма может просто потеряться при передаче, и мы ничего не узнаем. По этой причине UDP иногда полушутя расшифровывают как «Ненадежный протокол датаграмм [Unreliable datagram Protocol]». (На самом деле это «Протокол датаграмм пользователя [User Datagram Protocol]»). Проясним, что здесь означает «ненадежный». Мы не говорим, что UDP ненадежен по своей природе. Мы просто говорим, что в протоколе нет механизма проверки получения датаграммы и ее повторной переотправки, если она не дошла. Заодно проясним, что мы имеем в виду, говоря, что TCP является сервисом с «гарантированной» доставкой. Если кто-то выдернет из сервера сетевой кабель, ваши пакеты не дойдут. И деньги вы обратно не получите. Мы просто говорим, что в TCP есть встроенные механизмы подтверждения приема данных и их повторной передачи при необходимости.

На самом деле, всё еще хуже! UDP не только не гарантирует доставки сообщений, но даже не гарантирует, что их доставят в том порядке, в каком они отправлены. Конечно, если вы отправляете пакеты по локальной сети, шансы на то, что один пакет перепрыгнет другой, невелики, но с переходом к глобальной сети вероятность изменения порядка пакетов возрастает.

Вы можете спросить себя: а стоит ли вообще пользоваться сервисом доставки, который дает такие слабые гарантии? Ну, на самом деле многие приложения не упадут сразу же при потере пакета. Они просто немного ухудшат качество работы. Классический пример — VoIP (Voice Over IP — передача голоса по IP), которое использует протокол RTP (Real-time Transport Protocol — транспортный протокол реального времени), который обычно работает поверх UDP, для передачи мультимедиа. Для таких приложений своевременность доставки важнее надежности. Мы все знаем, что поймем разговоры по телефону со своими детьми, даже если их части утеряются. (Я имею в виду части разговоров, а не части наших детей или их телефонов.)

Текстовые и двоичные протоколы

В контексте этого разговора протокол приложения — это описание того, как клиент и сервер будут взаимодействовать друг с другом: типы сообщений, которые они могут отправлять и получать, и структура этих сообщений. Такие протоколы, как HTTP (Hypertext Transfer Protocol — протокол передачи гипертекста) и SMTP (Simple Mail Transfer Protocol — простой протокол передачи почты) являются текстовыми: сообщения, которыми они обмениваются, это простые строки текста. Конечно, это жестко структурированный текст, но по сути это просто последовательность печатных символов. Если перехватить трафик с помощью анализатора протокола

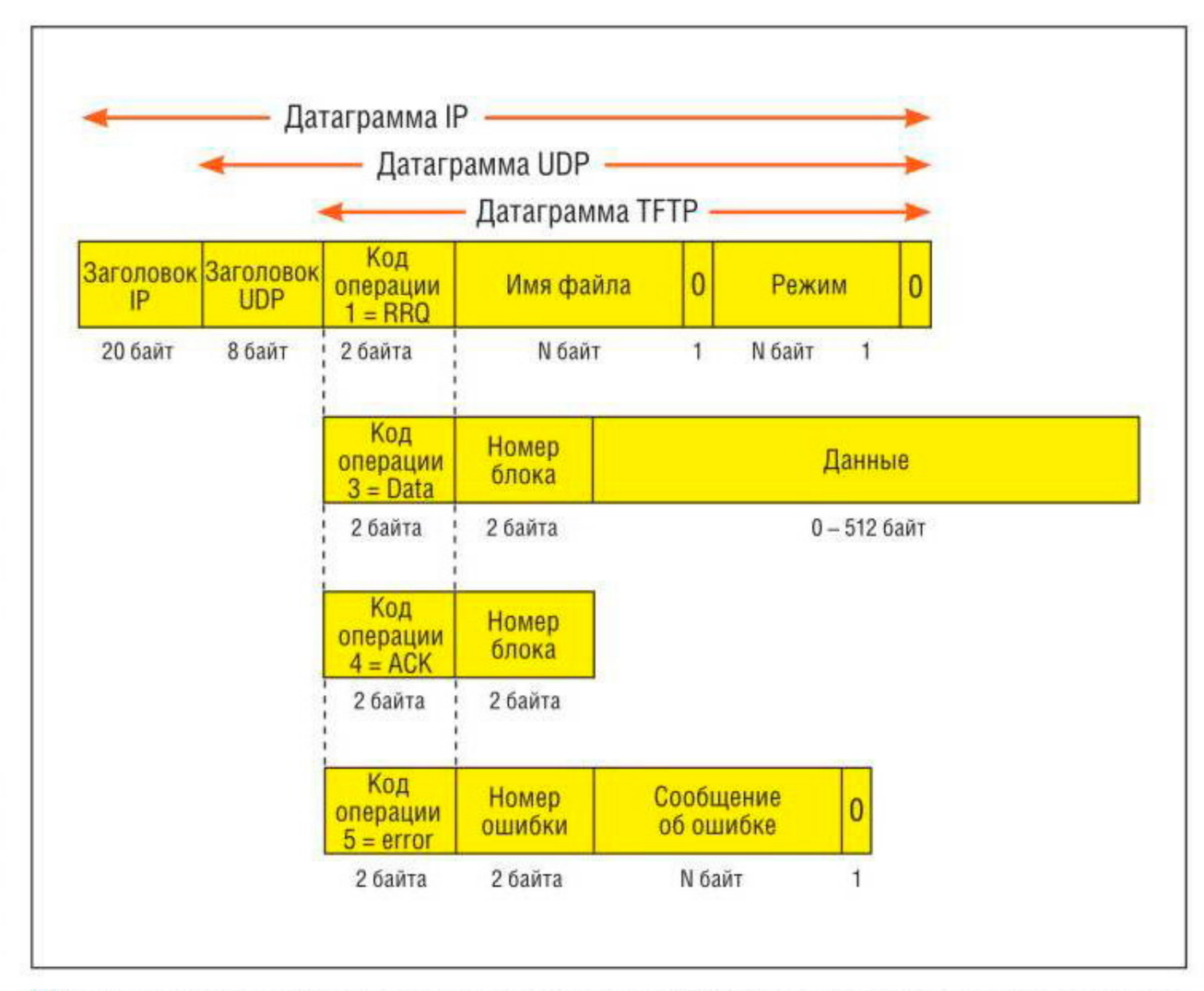
вроде *Wireshark*, и просмотреть его [см. Учебники **LXF191**], вы легко увидите, что происходит. В двоичных протоколах, с другой стороны, данные в сообщениях определяются на гораздо более подробном уровне. Например, протокол сообщения DNS определяет, что биты 16–31 в запросе содержат набор флагов. Внутри этого 16-битного поля шестой бит — это флаг, который означает «авторитетный ответ». Сбор и сопоставление таких сообщений требует другого стиля программирования по сравнению с обработкой строк, необходимой для текстовых протоколов. К счастью, язык C удобен в обработке данных на побитовом уровне. Двоичные протоколы часто сталкиваются с проблемой порядка байтов, о которой я рассказывал месяц назад. Например, если протокол говорит, что байты 4–7 сообщения содержат 32-битное целое, надо указать, в каком порядке они следуют. Идет ли старший байт первым или последним?

Познакомьтесь с Удаленной кошкой

Пора написать немного кода. Программа, о которой я хочу рассказать в этом месяце, называется *rcat*, сокращение от *remote cat* (удаленная *cat*), и с ее помощью можно открыть файл на удаленном компьютере: например, команда `$ rcat pluto poem.txt` откроет файл **poem.txt** с компьютера pluto. Программа является клиентом классического сервиса на основе UDP, TFTP (Trivial File Transfer Protocol — тривиальный протокол передачи файлов), и наряду с демонстрацией создания и использования UDP-сокета пример знакомит нас с концепцией протокола приложения и с тем, как формировать хорошо структурированные сообщения, которые необходимы протоколу.

Типы пакетов, используемые TFTP, показаны на схеме (см. рисунок справа вверху; схема взята из чудесной книги У. Ричарда Стивена [W. Richard Steven] «Иллюстрированное руководство по TCP/IP, том 1», http://amzn.to/1W0aueX). В верхнем ряду схемы мы можем наблюдать идею «инкапсуляции» — сообщение TFTP инкапсулировано в датаграмму UDP, которая, в свою очередь, инкапсулирована в датаграмму IP. На самом деле, заголовки UDP и IP добавляются к датаграмме по мере прохождения пакетом стека протокола на передающем конце и удаляются по мере прохождения пакетом стека протокола на приемном конце. Как разработчики приложений, мы можем игнорировать эти тонкости; наша задача — просто собирать и/или разбирать сообщения TFTP.

Вот как работает протокол. Наш клиент (*rcat*) начинает с отправки сообщения Read Request с кодом операции 1. Запрос включает имя файла переменной длины (завершается символом null) и режим, также завершаемый null. Режим — netascii или octet. Мы используем режим octet, то есть данные будут передаваться в виде 8-битных байтов без какой-либо специальной обработки символов завершения строк. Если все хорошо, сервер ответит, отправив последовательность блоков данных (код операции 3), и наш клиент должен подтвердить каждый блок (код операции 4). Передача заканчивается, когда клиент замечает, что сервер отправил блок данных с данными размером менее 512 байт. Это простая схема подтверждения позволяет избежать любых возможных проблем с доставкой пакетов в неправильном порядке, так как сервер не отправит очередной блок данных, пока клиент не подтвердит получение последнего. Однако это означает, что пропускная способность такой передачи ограничена, поскольку за время полного прохождения блока по сети мы можем отправить только один блок данных; это скорость передачи скорее всего будет ниже, чем та скорость, которую на самом деле может выдержать сеть. При возникновении проблемы (например, если клиент запрашивает несуществующий файл) сервер возвратит сообщение об ошибке (код операции 5) с текстовым описанием.



› На схеме показаны форматы пакетов протокола TFTP, использованные в нашем примере, но за полным описанием нужно обратиться к RFC1350 [*Ред.*: — О, спасибо!].

За более подробным описанием призываем обратиться к RFC1350. Это хороший вводный пример в достаточно формализованный мир спецификаций протоколов.

А подать сюда код!

Давайте взглянем на код *rcat*. Для простоты мы разбили его на две части. Вот первая: »

Настройка сервера tftp

Желая протестировать программу *rcat*, установите где-нибудь сервер *tftp*. В Ubuntu 14.04 это можно сделать командой

```
$ sudo apt-get install xinetd tftpd
```

Теперь создадим файл **/etc/xinetd.d/tftp** со следующим содержимым:

```
service tftp
{
protocol = udp
port = 69
socket_type = dgram
wait = yes
user = nobody
server = /usr/sbin/in.tftpd
server_args = /tftpboot
disable = no
}
```

Создайте каталог, который будет обслуживать сервер, и установите разрешения для него:

```
$ sudo mkdir /tftpboot
$ sudo chmod 777 /tftpboot
$ sudo chown nobody /tftpboot
```

Перезапустите демон *xinetd* командой

```
$ sudo /etc/init.d/xinetd restart
```

и, наконец, проверьте, что *xinetd* слушает порт 69, скомандовав

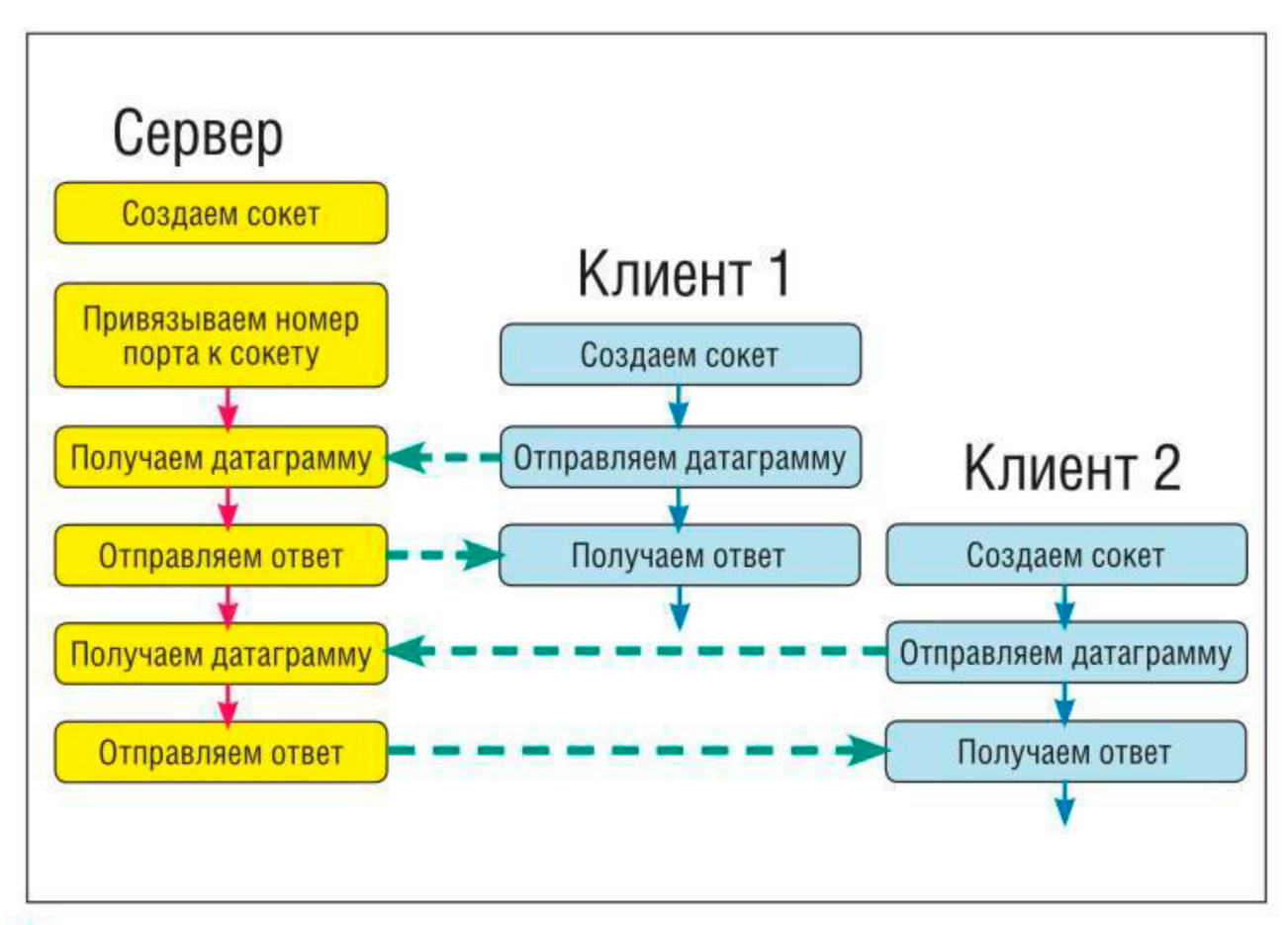
```
$ sudo lsof -i | grep tftp
```

Вы должны увидеть нечто вроде

```
xinetd 836 root 6u IPv4 9645 0t0
UDP *:tftp
```

» **Не хотите пропустить номер?** Подпишитесь на www.linuxformat.ru/subscribe/!

```
1 /* Usage: rcat hostname filename */
2
3 #include <netdb.h>
4 #include <stdio.h>
5 #include <string.h>
6 #include <stdlib.h>
7
8 #define TFTP_PORT 69 /* известный номер порта tftp */
9 #define BSIZE 600 /* Объем буфера данных */
10 #define MODE "octet"
11
12 #define OP_RRQ 1 /* Коды операций TFTP  */
13 #define OP_ACK 4
14 #define OP_ERROR 5
15
16 int main(int argc, char *argv[])
17 {
18   int sock; /* Описание сокета */
19   struct sockaddr_in server; /* Адрес сервера */
20   struct hostent *host; /* Информация о хосте сервера */
21   char buffer[BSIZE], *p;
22   int count, server_len;
23
24   if (argc != 3) {
25     fprintf(stderr, "использование: rcat hostname filename\n");
26     exit(1);
27   }
28
29   /* Создаем сокет датаграммы*/
30   sock = socket(AF_INET, SOCK_DGRAM, 0);
31
32   /* Получаем адрес сервера */
33   if ((host = gethostbyname(argv[1])) == NULL) {
34     fprintf(stderr, "хост не найден: %s\n", argv[1]);
35     exit(2);
36   }
37   server.sin_family = AF_INET;
38   memcpy(&server.sin_addr.s_addr, host->h_addr, host->h_length);
39   server.sin_port = htons(TFTP_PORT);
```



› Как показано на рисунке, со временем сервер без подключения может обнаружить себя принимающим датаграммы от нескольких клиентов на один сокет.

Часть кода может быть вам знакома по примеру TCP прошлого месяца — в частности, структура sockaddr_in, которая содержит адрес конечной точки (IP-адрес и номер порта) сокета. Строки 24–26 проверяют, что пользователь указал достаточно параметров, и если нет, журят его. В строке 30 мы создаем наш клиентский сокет; обратите внимание, что его тип на сей раз — SOCK_DGRAM, что по сути означает использование UDP. В строке 33 мы пытаемся разрешить имя хоста, которое указал пользователь, в IP-адрес (возвращаемый в структуре hostent); если этого сделать не удалось, мы сообщаем об ошибке и выходим из программы. Строки 37–39 связаны с заполнением структуры sockaddr_in.

Этот код по сути тот же, что был приведен в прошлом месяце; различие в том, что на сей раз мы не будем выполнять вызов connect() к серверу (так как подключение отсутствует); вместо этого мы воспользуемся им, чтобы направить датаграммы UDP в правильное место.

Пока все хорошо, хотя почти ничего еще не произошло. Ни один байт не прошел по сети, и сервер пребывает в блаженном неведении о нашем существовании.

И остальное...

Вот остальная часть кода. Здесь код становится интереснее, поскольку мы начинаем обмениваться сообщениями с сервером:

```
41   /* Формируем пакет tftp Read Request. Это сложно, поскольку
42    у полей переменная длина и нельзя использовать структуру.
43    По завершении указатель p будет после конца полного пакета
44   */
45   *(short *)buffer = htons(OP_RRQ); /* Код операции - op-code */
46   p = buffer + 2;
47   strcpy(p, argv[2]); /* Имя файла */
48   p += strlen(argv[2]) + 1; /* Храним nul */
49   strcpy(p, MODE); /* Режим Mode */
50   p += strlen(MODE) + 1;
51
52   /* Посылаем Read Request на сервер tftp */
53   count = sendto(sock, buffer, p-buffer, 0,
54     (struct sockaddr *)&server, sizeof server);
55
56   /* В цикле собираются пакеты данных с сервера, пока
57     не прибудет короткий пакет. Это означает конец файла.
58   */
59   do {
60     server_len = sizeof server;
61     count = recvfrom(sock, buffer, BSIZE, 0,
62          (struct sockaddr *)&server, &server_len);
63       if (ntohs(*(short *)buffer) == OP_ERROR) {
64          fprintf(stderr, "rcat: %s\n", buffer+4);
65       }
66     else {
67       printf("Пришло %d байт с s\\сервера\n", count -4);
68       write(1, buffer+4, count-4);
69       /* Шлем пакет ACK. Подтверждаемое число блоков уже
70        в буфере, остается изменить код операции (opcode).
71        Учтите, что ACK посылается на номер порта,
72          с которого сервер отправил данные, а НЕ на порт
73          69
74       */
75       *(short *)buffer = htons(OP_ACK);
76       sendto(sock, buffer, 4, 0, (struct sockaddr *)&server,
77          sizeof server);
78     }
79   } while (count == 516);
```

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

```
80
81    return 0;
82 }
```

Начнем с сообщения запроса на чтение. Как можно предположить из комментария в коде, оно немного запутанное, так как поля “filename” и “mode” этого сообщения имеют переменную длину, и мы не можем просто так объявить структуру C, которая отражала бы схему сообщения. Вместо этого в строках 45–50 мы собираем сообщение по кусочкам, проталкивая указатель (p) по нашему буферу (buffer). Важно, что когда мы выходим из этого кода, p указывает на предпоследний байт сообщения, а buffer по-прежнему указывает на начало, поэтому по правилам арифметических действий с указателями в C, выражение (p-buffer) позволит определить длину сообщения. Здесь явно предполагается, что символ равен одному байту; это предположение в C распространено настолько широко, что, подозреваю, мои компьютер, смартфон, телевизор и автомобиль сразу же перестали бы работать, будь это не так. Также обратите внимание на вызов макроса htons() (host to network short — хост в сетевое короткое) в строке 45, который делает так, чтобы код операции имел сетевой порядок байтов — т.е. от старшего к младшему.

И настает момент наивысшего восторга: в строке 53 мы отправляем сообщение «запроса на чтение» серверу. Системный вызов sendto() похож на классический вызов write(), но имеет дополнительные параметры, которые указывают адрес конечной точки, которой мы хотим отправить сообщение. Помните, что это модель без подключения — при отправке каждой датаграммы нужно указывать, куда мы хотим ее отправить. Цикл, начинающийся в строке 59, получает содержимое файла обратно с сервера, по 512 байт. Вызов recvfrom() получает сообщение из сокета UDP; на подробный разбор вызова у меня не хватит места, и я верю, что вы заглянете на man-страницу recvfrom(), если будет интересно.

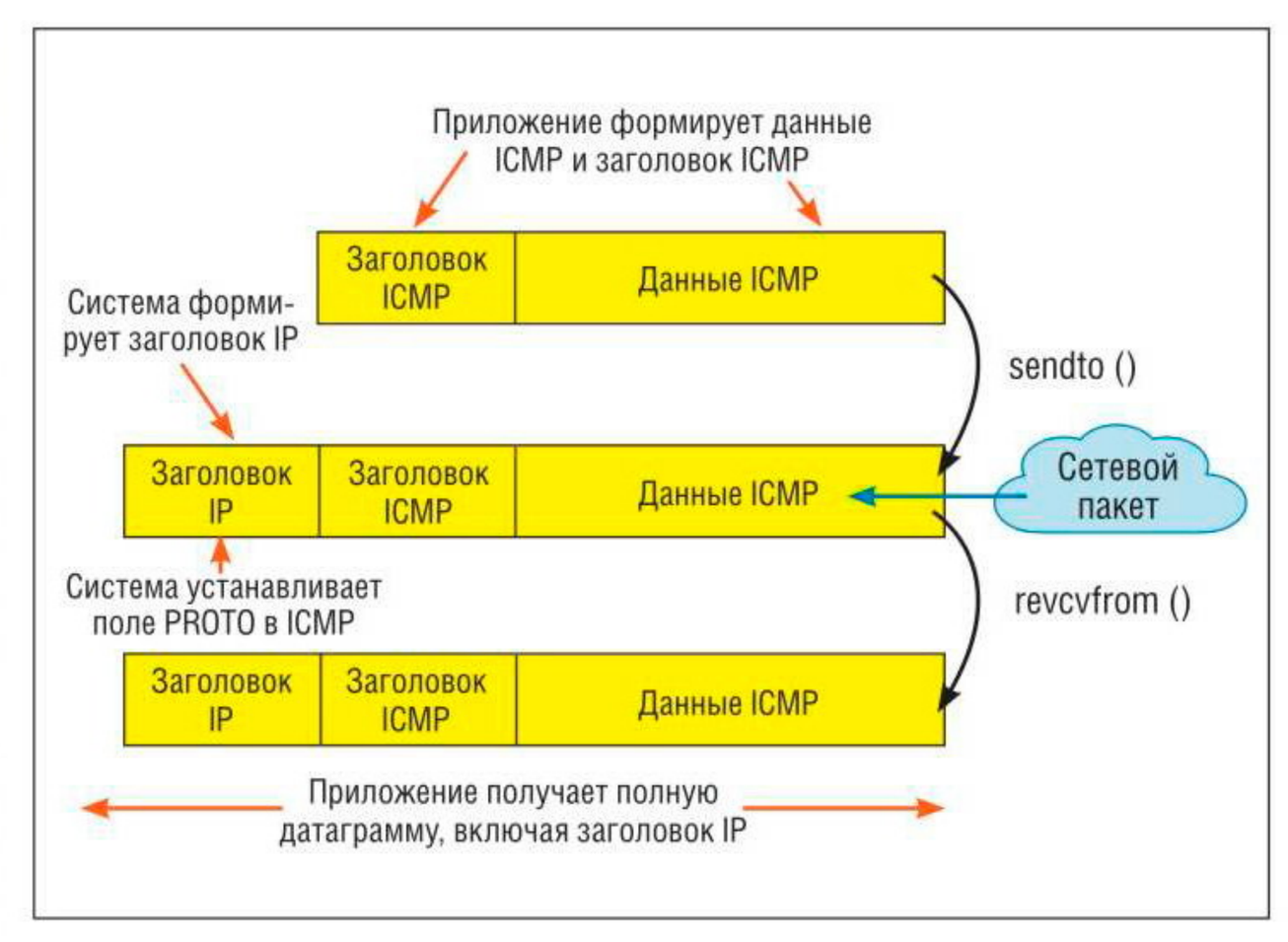
Строка 63 — это «момент мармита [британский пищевой спред, который одни считают вкусным, а другие мерзким; его рекламный лозунг — «любить или ненавидеть», — *прим. пер.*]». Здесь мы извлекаем код операции из первых двух байтов ответа, и вам или понравится сжатость этого кода, или вы уроните руки и подумаете, что COBOL, пожалуй, был не так уж плох. Так или иначе, дело тут в том, что если сервер возразит на наш запрос, в строке 64 мы сообщим об ошибке, которую он возвратил.

Если все хорошо, то в строке 68 мы записываем порцию данных буфера в файловый дескриптор 1 (стандартный вывод). Затем (в строках 75–77) нам надо уважить протокол TFTP и подтвердить пакет. Единственное, что может быть неясно из кода — если вы не читали комментарии — то, что адрес конечной точки, в который мы доставляем подтверждение (переменная server в строке 76) — это тот адрес, который был заполнен вызовом recvfrom() в строках 61 и 62.

Честно говоря, протокол TFTP в нашем коде реализован не лучшим образом. В частности, RFC1350 утверждает, что на обоих концах должен быть реализован таймаут и повторная передача пакетов, если они не были подтверждены; а мы этого не сделали. Но это уже начало, и в сети без потери пакетов код будет работать отлично.

В защиту UDP

Возможно, наше обсуждение TCP и UDP за последние два урока оставило у вас впечатление, что UDP — своего рода «бедный родственник» TCP. Чтобы подправить равновесие, я бы хотел упомянуть две возможности UDP, которых TCP не хватает. Во-первых, UDP поддерживает идею границ сообщений: например, если компьютер A отправляет три датаграммы UDP компьютеру B, одну в 200 байт, одну в 300 байт и одну в 100 байт, именно это B и получит: три отдельных датаграммы. Для соединения TCP, если A отправляет B 200, 300 и 100 байт, то все, что узнает B — что получено 600 байт, которые он должен прочесть. Если трафик построен как три отдельных сообщения, то протокол приложения должен предоставить способ определения окончания каждого сообщения.



› Использование сырого сокета для отправки и приема сообщений ICMP.

В UDP есть еще одна магическая штука, на которую не способен TCP, и это широковещание. Оно полагается на широковещание более низкоуровнего протокола IP. Идея состоит в том, что если часть «хост» IP-адреса состоит только из единиц, то датаграмма будет получена всеми хостами этой сети. На практике роутеры обычно настроены не пропускать широковещательный трафик в другие сети (этот трафик называется «направленным широковещанием»), но широковещание в пределах одной сети обычно возможно и открывает ряд интересных возможностей.

Прощайте, и спасибо за рыбку

Тут наша серия о системном программировании завершается, и это моя последняя статья для *Linux Format*. Я подвел некоторые итоги. За эти несколько лет я написал 43 руководства и 87 статей «Рецептов», и это — почти полмиллиона слов (не обязательно разных — некоторые из них могли повторяться, как вы понимаете). Спасибо тем из вас, кто написал слова похвалы; всегда приятно знать, что твои труды не отправляются в **/dev/null**. Мне особенно понравилось письмо от переводчика журнала на русский. Спасибо всем моим редакторам за эти годы — Нику Вейчу, Полу Хадсону, Грэму Моррисону и Нейлу Мору. И спасибо Future Publishing за то, что подбрасывали рыбку. LXF

Сырые сокеты

В дополнение к созданию сокетов TCP и UDP также можно создавать так называемые «сырые» сокеты. К таким сокетам система не добавляет заголовок TCP или UDP, при этом заголовок IP добавляется. Для создания такого сокета программа должна быть запущена с правами root. Зачем вам может потребоваться сырой сокет? Ну, классический пример — это программа *ping*, которая отправляет и принимает ICMP-сообщения. В этом случае сокет создается примерно так: s = socket(AF_INET, SOCK_RAW, IPPROTO_ICMP), и ваш код должен создать не только сообщение ICMP, но и заголовок ICMP. При отправке пакета с помощью вызова sendto() система добавляет для вас заголовок IP. А вот при получении сырого сокета с помощью вызова recvfrom() вы получаете полную датаграмму, включая IP-заголовок.

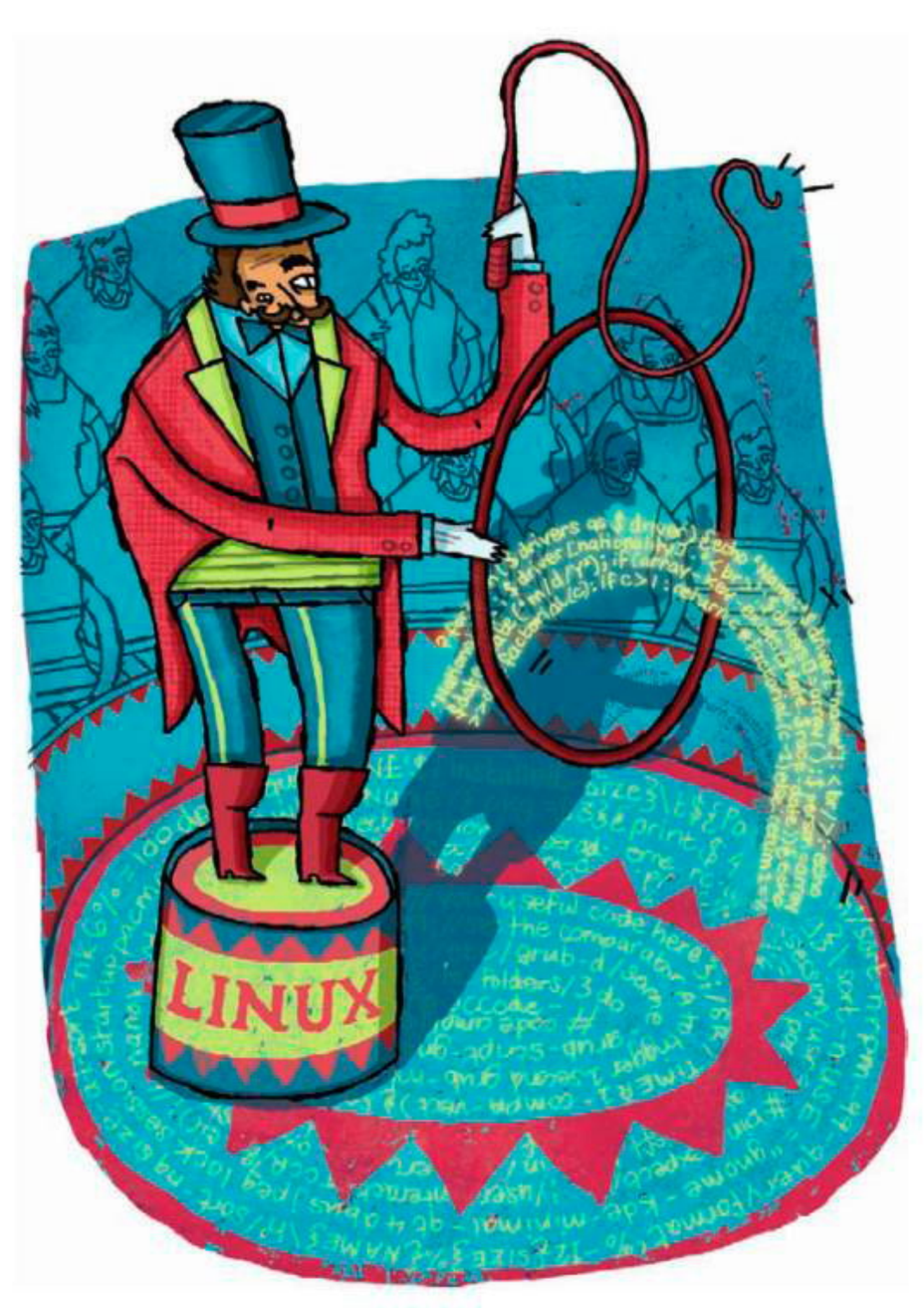
Еще один пример использования сырых сокетов — программа перехвата пакетов, такая как *Wireshark* (хотя, подозреваю, она должна обращаться к сетевой карте на еще более низком уровне, и даже отображает заголовок пакета Ethernet.)

Python: GTK-GUI с Glade

Ваши программы на Python выиграют от стильных интерфейсов пользователя на *GTK* — спасибо *Glade*. Примкнувший **Джонни Бидвелл** вас научит.

Наш эксперт

Джонни Бидвелл устал от серых домов и тротуаров; он мечтает о хижине на озерном острове Иннисфри, чтобы жить в одиночестве на полянке, где жужжат пчелы.



Мы в *Linux Format* перепробовали все виды программирования на начальном уровне: возились с *Minecraft*, древними настольными играми на Python, играми в слова на PHP [*Ред.*: — То есть наши авторы целыми днями в игры играют?!]. Но все это или было чисто текстовыми заморочками в консоли, или зависело от некоторых заранее созданных графических интерфейсов; разве не очевидно, что вы должны иметь шансы сделать собственный? Большинство графических приложений (по крайней мере, тех, что запускаются в собственном окне), с которыми вы столкнетесь в Linux, делятся на два лагеря: *Qt* (произносится «кьюти») и *GTK* (произносится... [*Ред.*: — Ох, да ладно!]. Это два разных инструментария, на которых строятся приложения KDE и Gnome соответственно. Почтенную, но полную ошибок версию 2 *GTK* вытеснила версия 3, тогда как большинство приложений KDE (и солидная часть приложений *Qt* из прочих областей экосистемы) до сих пор не портированы в *Qt 5*, самую свежую версию данного инструментария. По этой причине мы пошли путем *GTK 3*, и инструмент, который мы употребим для создания нашего прекрасного интерфейса, называется *Glade*. Вполне возможно создавать чудесные интерфейсы *Qt* с помощью *Qt Creator* и вдыхать в них жизнь кодом Python, но это не то, чем мы будем заниматься сегодня. *Glade* никак не связан с Python, он может работать с любым количеством языков программирования, но в данном руководстве мы обойдемся Python.

На нашем уроке мы создадим (взаправду) простое приложение календаря. Оно позволит вам выбирать дату и вводить всяческие подробности своего насыщенного графика. Это подробности будут запоминаться в файле, что обеспечит их сохранность. Календарь будет выделять дни, когда что-то будет происходить, и, помимо следования стандартным протоколам закрытия окон, это все, чем он будет заниматься. Простенько.

Старый инструментарий *GTK 2* использовал для общения с Python модуль PyGTK, однако *GTK 3* использует более новые и куда более подходящие для наших целей библиотеки *GObject Introspection*. *Linux Format* продолжает поддерживать переход на Python 3, а если вы отказываетесь уходить от предыдущей версии, то код потребует некоторых изменений. Вам также понадобится *Glade*; мы использовали версию 3.18, но не употребляем никаких передовых функций, поэтому берите то, что есть в репозитории вашего дистрибутива. Мы предполагаем, что все наши календарные файлы проживают в директории под названием **~/gtk3cal**, хотя в коде это не «зашито». Вы найдете все файлы на **LXFDVD**, так что можете скопировать их в эту папку, если вам неохота набирать добрую сотню строк кода. Запуск программы из ее папки на диске корректно не сработает, поскольку там она только для чтения. Если у вас есть все инструменты, можете сразу запустить проект, скомандовав **python calendar.py** внутри вышеупомянутой папки. Если у вас выскочит ошибка, возможно, вы пропустили какие-то пакеты, так что читайте дальше.

Пакеты для Glade

Для систем на базе Debian можно найти все требуемые пакеты разом с помощью

```
$ sudo apt-get install glade python3 python3-gi
```

Пользователи Arch Linux должны учесть, что связи *GObject* именуются *python3-gobject*. *Glade* поместит иконку в ваше меню Application, возможно, под Programming или Development, но если вы не можете найти иконку, запустите его из терминала. Приступая сразу к делу, следуйте шестишаговому руководству (см. стр. 87), чтобы создать наш маленький, но совершенный по форме UI. *Glade* сохраняет свои файлы в виде XML, так что вы можете повозиться с ними в текстовом редакторе, если вам этого хочется. В ином случае, следующий фрагмент Python позволит вам сделать демо вашего творения:

```
#! /usr/bin/env/python3
from gi.repository import Gtk
builder = Gtk.Builder()
builder.add_from_file('calendar.glade')
window = builder.get_object("window1")
window.show()
```

Ваш друг FleXML

Мы решили использовать XML для хранения наших данных. Он хорошо поддерживается в Python через модуль ElementTree и означает, что мы можем хранить и работать с деревом XML в памяти, что способствует обычной загрузке и сохранению. Если программа заботится о своей человеко-читаемости, то календарь, содержащий одно назначение, будет выглядеть так:

```
<calendar>
  <appt>
    <date>
      <year>2015</year>
      <month>0</month>
      <day>1</day>
    </date>
    <detail>Happy new year</detail>
  </appt>
</calendar>
```

Как видите, виджет календаря имеет индекс 0 для месяца и больше ничего. Следует признать, что это отчасти странно, но проблему легко обойти. К сожалению, читаемости эта программа не придает особого значения, поэтому все хранится в одной строке.

Дерево объектов XML хранится в переменной self.root. Функция loadFile() пытается загрузить его из файла, но если он не находится, то вместо него создается пустое дерево. Затем мы можем обратиться к saveAppt(), чтобы посмотреть, как создавать требуемую структуру:

```
def saveAppt(self, *args):
  ...
  newAppt = ET.SubElement(self.root,'appt')
  newDate = ET.SubElement(newAppt, 'date')
  Year = ET.SubElement(newDate, 'year')
  Month = ET.SubElement(newDate, 'month')
  Day = ET.SubElement(newDate, 'day')
  Year.text = str(date[0])
  Month.text = str(date[1])
  Day.text = str(date[2])
  newDetail = ET.SubElement(newAppt,'detail')
  newDetail.text = detail
```

```
window.connect("delete-event", Gtk.main_quit)
if __name__ == "__main__":
  Gtk.main()
```

Вы можете увидеть пару предупреждений насчет того, что опции *GTK 3* устарели; предупреждения можно спокойно игнорировать, *Glade* иногда помещает лишние и старые директивы в свои файлы. В приведенном выше коде мы импортировали требуемый модуль, загрузили наш интерфейс через объект builder, отобразили основное окно и предоставили часть кода, чтобы приложение закрывалось должным образом (иначе Python повиснет). Последние две строки шаблона могут показаться вам знакомыми; они просто дают приложению команду запуститься, когда оно вызывается из командной строки. Сохраните его как **~/gtk3cal/gladetest.py** и затем запустите его с помощью python3 gladetest.py. Собственно говоря, прямо сейчас оно ничего не делает, но вам простительно решить, что оно что-то делало. По крайней мере, вы можете пробежаться по календарю и увидеть, что Вальпургиева ночь в следующем году приходится на субботу.

Селфи, селфи

Графические приложения привержены объектно-ориентированному подходу. Данные должны перемещаться между различными компонентами, и намного проще, если все они будут в одном объекте, что определено в нашем calendarApp. Если вы знакомы только с процедурным программированием, вам бросится в глаза неумеренное употребление ключевого слова self. Переменные в области именования self принадлежат к классу, и они доступны из любой функции, наследующей этот класс, т.е. любой, которая содержит ключевое слово self как первый аргумент в своем определении. При определении этого объекта вызывается специальный метод __init__. Соответствующие шаги инициализации из демокода здесь пропущены, и вот фрагмент кода из того, что осталось:

```
class calendarApp:
def __init__(self):
...
handlers = {"onDeleteWindow":Gtk.main_quit,
"onButtonClick":self.saveAppt,
"onDaySelect":self.checkAppt,
"onPageChange":self.markDays}
self.builder.connect_signals(handlers)
self.buffer = self.textview.get_buffer()
self.calendar = self.builder.get_object("calendar1")
self.textview = self.builder.get_object("textview1")
self.window = self.builder.get_object("window1")
```

Здесь мы применили другой способ соединить сигналы с функциями-обработчиками: определение их в словаре и соединение всех сразу. Затем нам понадобится сделать несколько переменных для различных виджетов, которые мы ищем по ярлыкам, присвоенным им в *Glade*. Виджет textview может делать что угодно — он сложнее обычного Text Entry. Как таковому, ему не хватает простого доступа к своему контенту: доступ осуществляется через буферный объект. Остаток кода __init__ здесь не показан — он вызывает некоторые функции, которые мы еще не написали, и содержит следующий удобный прием для обеспечения запуска приложения с выделением сегодняшней даты:

```
today = datetime.datetime.now().date()
self.calendar.select_day(today.day)
self.calendar.select_month(today.month - 1,today.year)
```

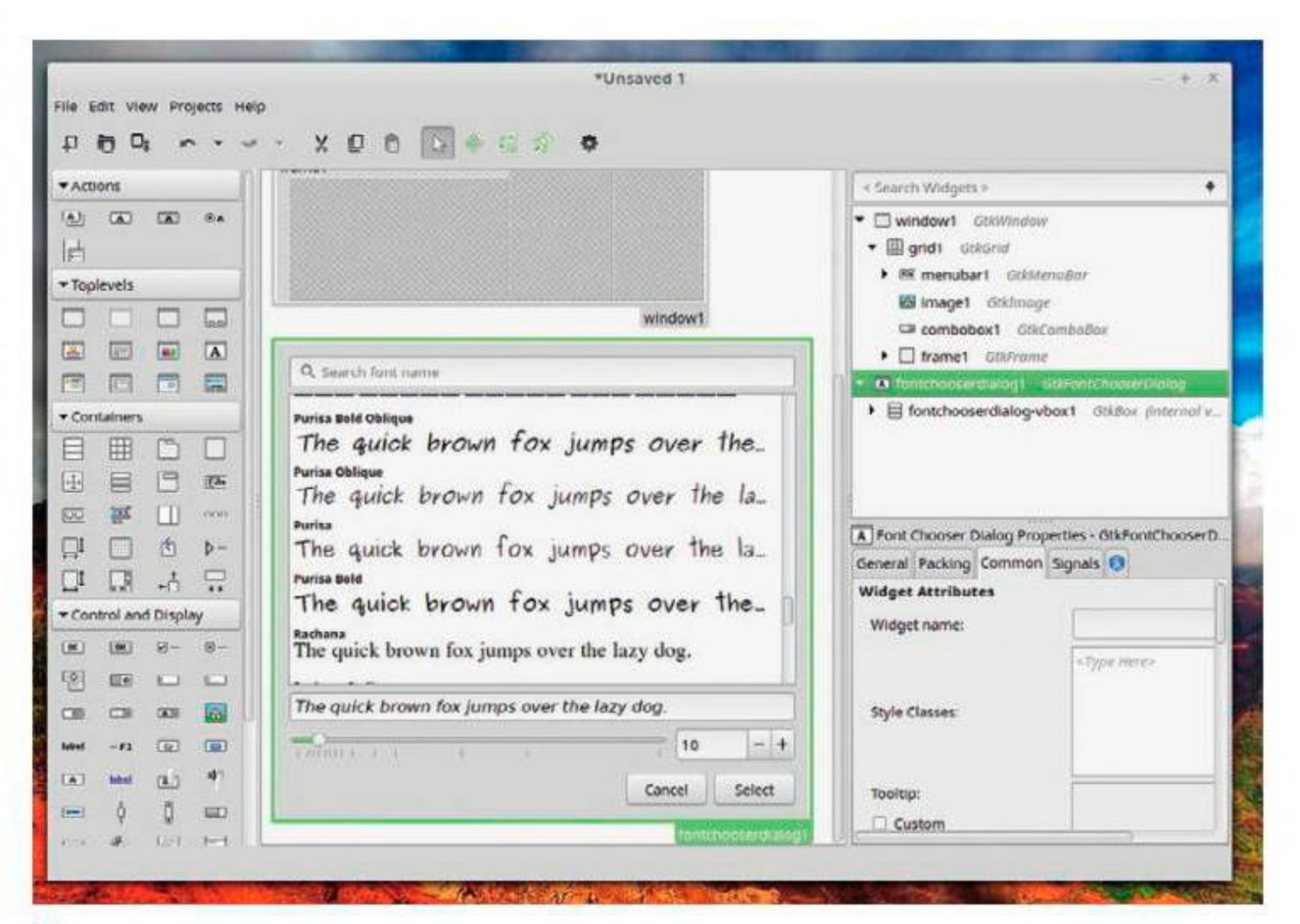
Наш класс определяет пять других функций помимо __init__, и вот обзор того, что они делают:

» **loadFile()** Считывает базу данных в ElementTree, если она существует, и запускает пустую, если нет.

» **markDays()** Отслеживает наличие встреч в этом месяце/году и отмечает их.

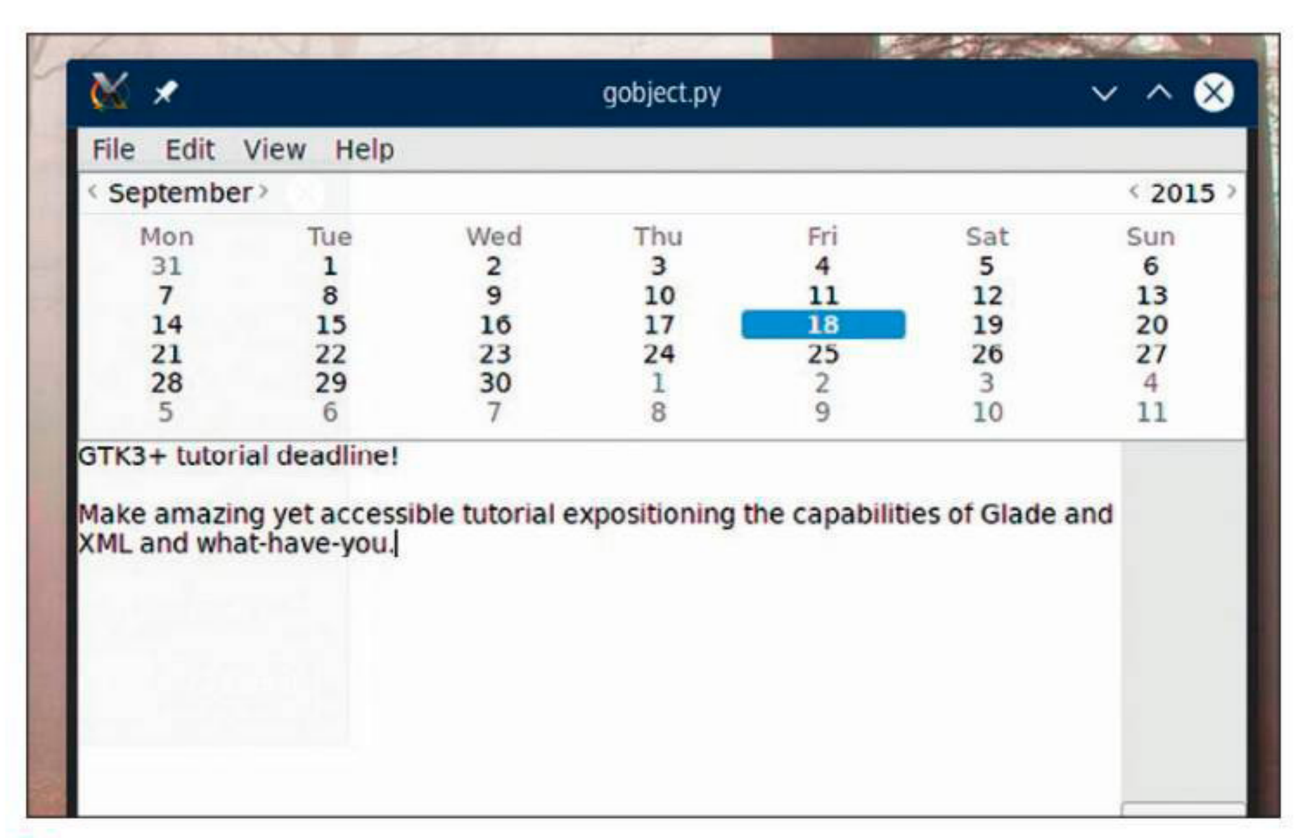
Скорая помощь

Если вы всерьез намерены создавать человеко-читаемые файлы XML, лучшим решением будет использовать модуль LXML.Можно уговорить ElementTree на красивую печать, но для этого придется попотеть. Загляните в пост нашего Норберта [Norbert] — http://bit.ly/XMLtrees.



› Палитра *GTK 3+* весьма разнообразна: вам предоставляются диалоговые окна шрифтов, а также календарей, но вы не обязаны их использовать.

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



› Что, этот учебник нужен уже сегодня?! Вот бы нашему автору хорошее приложение календаря, чтобы отслеживать подобные вещи.

» dateObj2Tuple() Извлекает даты из хранилища XML и конвертирует их в кортеж целых чисел.

» checkAppt() Проверяет наличие встреч в выбранном календарном промежутке и очищает текстовое окно при их отсутствии.

» saveAppt() Сохраняет текущую встречу, переписывает существующие встречи на выбранный день.

Мы воспользуемся отличной процедурой исправления ошибок Python, чтобы избежать ошибки, возникающей, если файл **calendar.xml** не найден в текущей директории:

```
def loadFile(self):
  try:
    self.tree = ET.parse(self.datafile)
    self.root = self.tree.getroot()
  except:
    self.root = ET.Element('calendar')
    self.tree = ET.ElementTree(self.root)
```

Корневой элемент ElementTree в нашем случае относится к самому ближнему внешнему тэгу <calendar> (см. врезку «Ваш друг FleXML», стр. 85). Мы проводим навигацию по всему относящемуся к этому дереву, и Python даже предоставляет метод iter() для поиска по подтэгам. Удаление тэгов (и любых данных внутри них) осуществляется с помощью метода remove().

Даты, возвращенные календарем, организованы в том же виде, что и кортеж формы (год, месяц, день), но XML целиком основан на строковых переменных. Таким образом, у нас появляется уродливая функция для конвертирования последней в первую, под названием dateObj2Tuple(). Мы не подгоняем календарные месяцы, начинающиеся с нуля, поскольку нам бы понадобилось заново подгонять их позднее, и мы уже решили, что человеческие существа XML видеть не будут. Календарь любезно предлагает собственный метод выделения дат, чем мы законно пользуемся в нашей функции markDays():

```
def markDays(self, *args):
  self.calendar.clear_marks()
  calDate = self.calendar.get_date()
  for j in self.root.iter('date'):
    date = self.dateObj2Tuple(j)
    if calDate[0] == date[0] and calDate[1] == date[1]:
      self.calendar.mark_day(date[2])
```

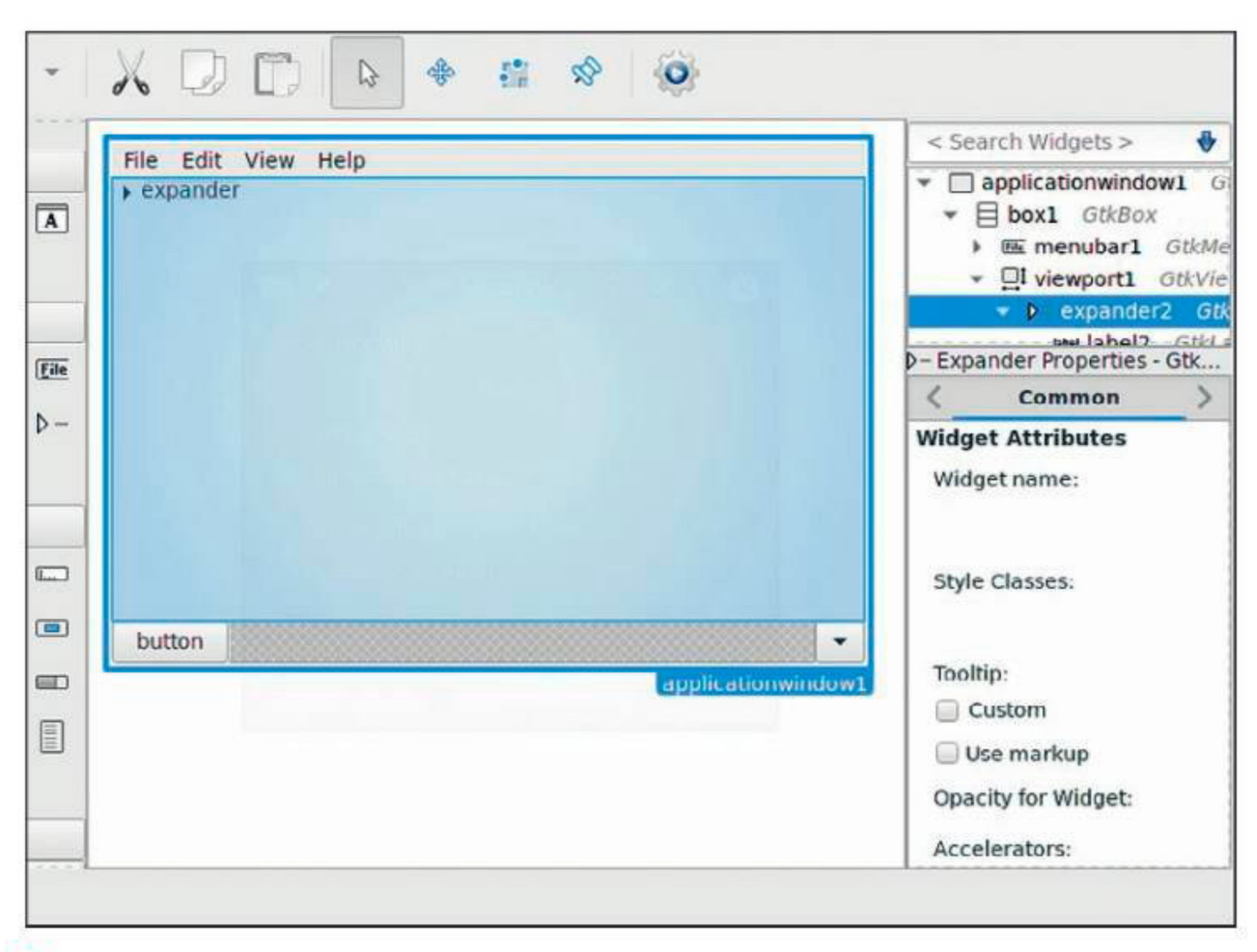
Метод календаря mark_day() отмечает даты в месяцах или годах, так что предпоследняя строка проверяет, что встреча состоится именно в этом месяце и году. Чтобы такое обеспечить, при каждой смене месяца и года вызывается функция markDays (обратите внимание на вызов clear_marks() — он заботится об удалении старых отметок). Итерируя все даты в нашем цикле for, мы можем легко совместить наши встречи с текущим месяцем. Используй мы для сохранения своих данных обычный файл CSV, вместе с удалением пунктов это было бы намного запутаннее. Вдобавок тогда нам пришлось бы принять на себя обязанность избегать нештатных символов в тексте записи.

Функции markDays и checkAppt() (следующая, которую мы рассмотрим) обе содержит в своих определениях странное ключевое слово *args. На самом деле, это всего лишь заполнитель для неожиданных/нежелательных дополнительных аргументов. Когда функция вызывается сигналом страницы, ей часто передаются лишние данные: например, то, какая кнопка мыши была нажата. Поскольку вызов функции с неверным количеством аргументов вызывает ошибку в Python, мы используем *args для того, чтобы по-тихому выявить и проигнорировать ошибки.

```
def checkAppt(self, *args):
  date = self.calendar.get_date()
  for j in self.root.iter('appt'):
    dateTup = self.dateObj2Tuple(j[0])
    if dateTup == date:
      self.buffer.set_text(j[1].text)
      break
    else:
      self.buffer.set_text('')
```

Здесь мы извлекаем любые встречи на текущий день, прогоняя все их через цикл. Мы можем получить доступ к подтэгам с помощью нотации списка, поэтому j[0] относится к части даты встречи, а j[1] — к ее подробностям. Мы используем нашу вспомогательную функцию dateObj2Tuple(), чтобы эта функция оставалась аккуратной.

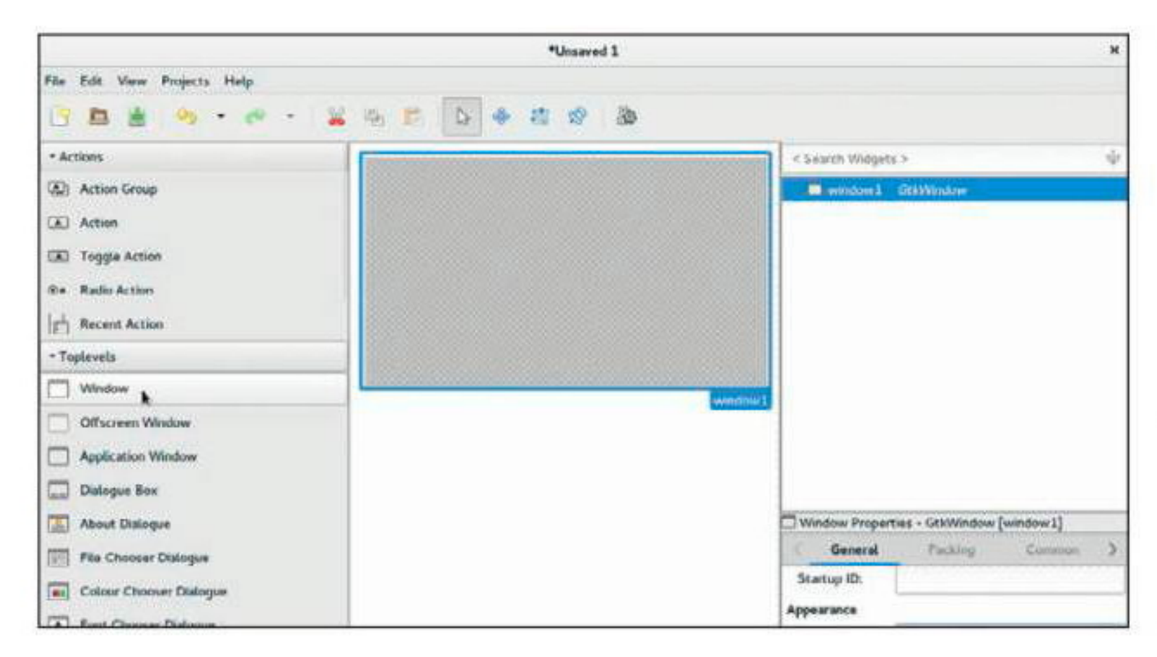
На чем и завершается наш очередной замечательный урок; мы рассмотрели не все строки кода, но, надеюсь, достаточно, чтобы вы уловили суть. Проекты такого типа склонны к расширяемости — можно добавлять ко встрече время, менять формат XML, чтобы он импортировался в другие форматы календарей, добавлять нудные уведомления в системный лоток при приближении встречи... список можно продолжать бесконечно.



› У вас нет власти над расположением элементов в *Glade*; его структура текуча и хорошо ведет себя в рабочих окружениях, настроенных не вами.

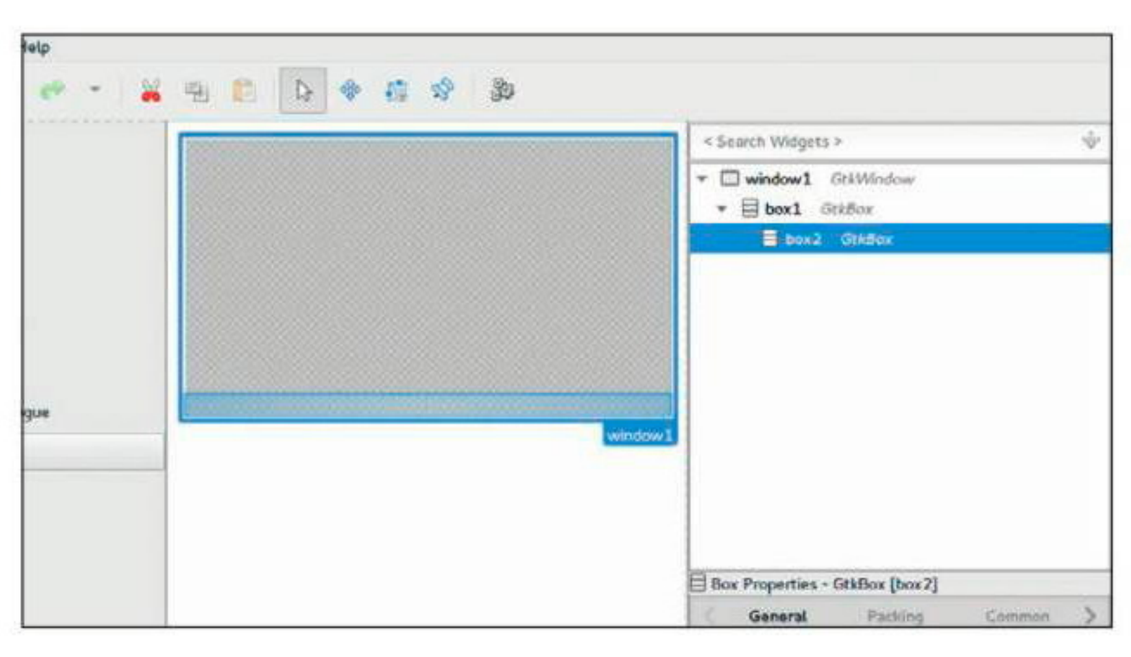
» **Пропустили номер?** Узнайте на с. 108, как получить его прямо сейчас.

Создаем интерфейс пользователя



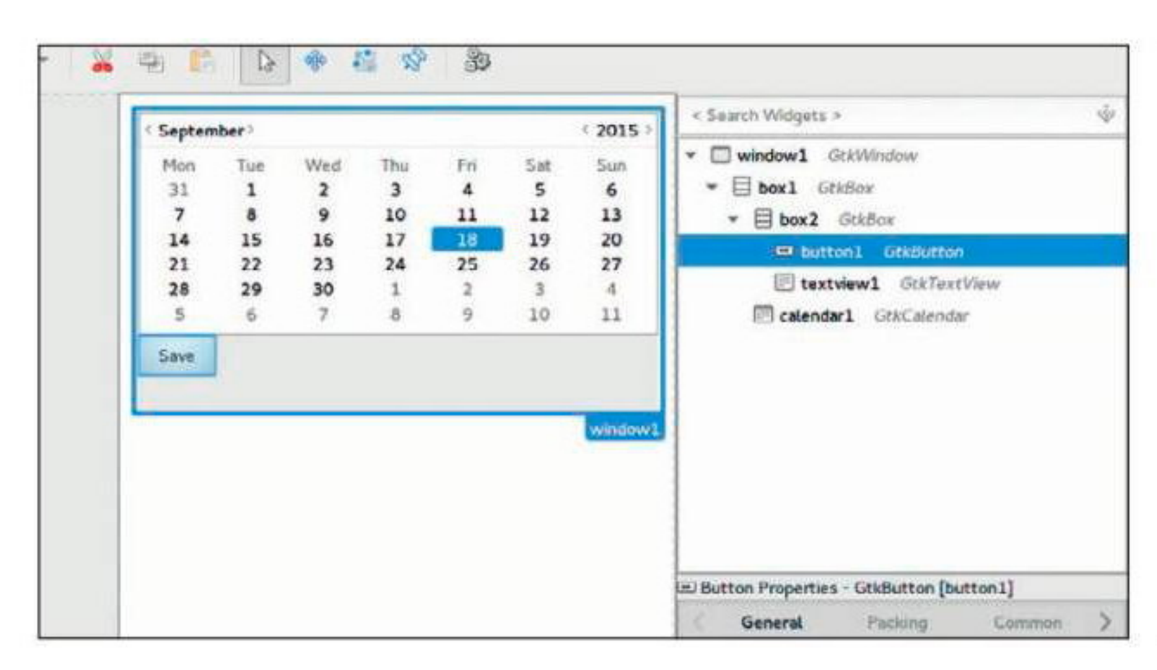
1 Окно в мир

Интерфейс *Glade* может немного напомнить вам Visual Basic, но не давайте этому сходству сбить вас с толку. Выберите инструмент Window из раздела Toplevels на левой панели. Щелкните в любом месте фона, чтобы разместить ваше окно. Прокрутите раздел Appearance на панели внизу справа и укажите подходящее имя для вашего окна — например, **LXF** Calendar.



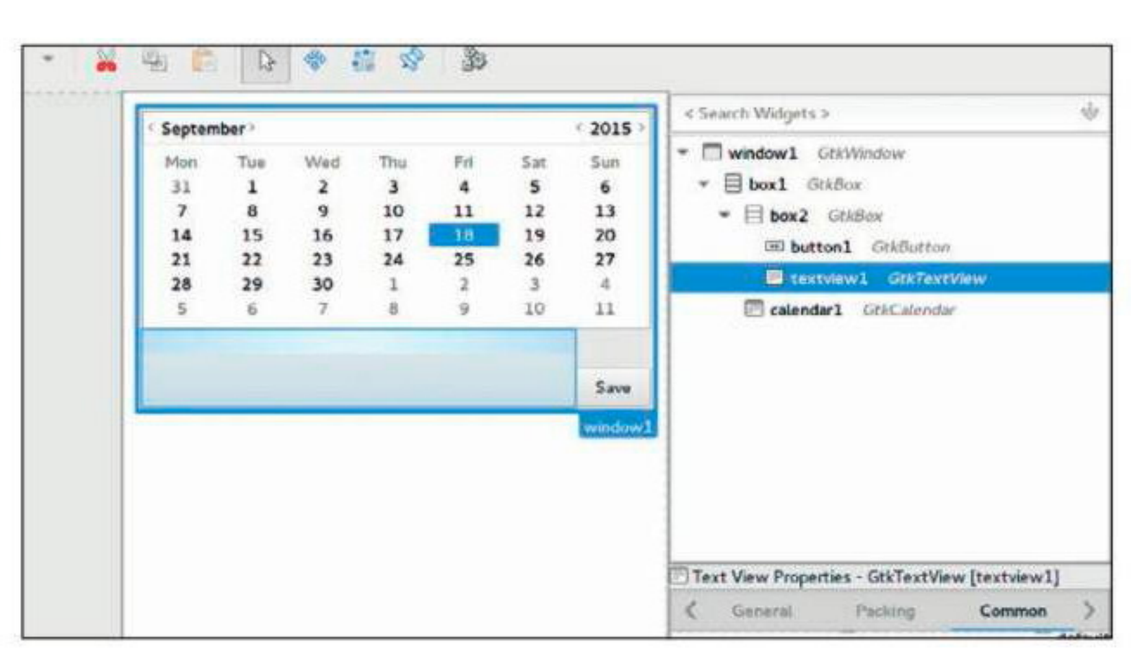
2 Окна внутри окон

Все наши виджеты прямо в объект Window вставить нельзя, их надо поместить в контейнер. Мы воспользуемся виджетом Box из раздела Containers. Поместите окно внутрь вашего Window и задайте ему два пункта, когда вас об этом попросят. Поместите еще одно окно с двумя пунктами внутри низшего уровня первого окна и смените его ориентацию на горизонтальную. Настройте для него опцию Expand.



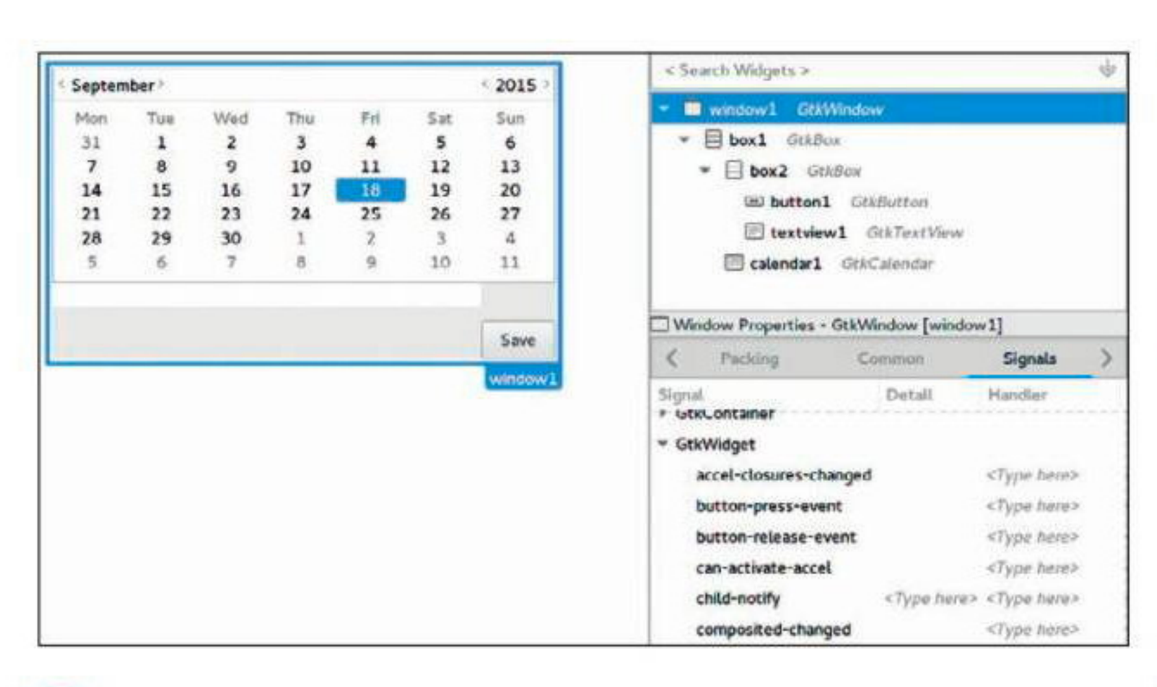
3 Так смеется календарь

Найдите виджет Calendar в разделе Control and Display и поместите его на верхнюю половину вашего окна. Найдите виджет Text View и поместите его внизу слева вашего окна. Разместите Button внизу справа и из вкладки General панели Properties в нижней правой части *Glade* задайте ему имя 'Save'.



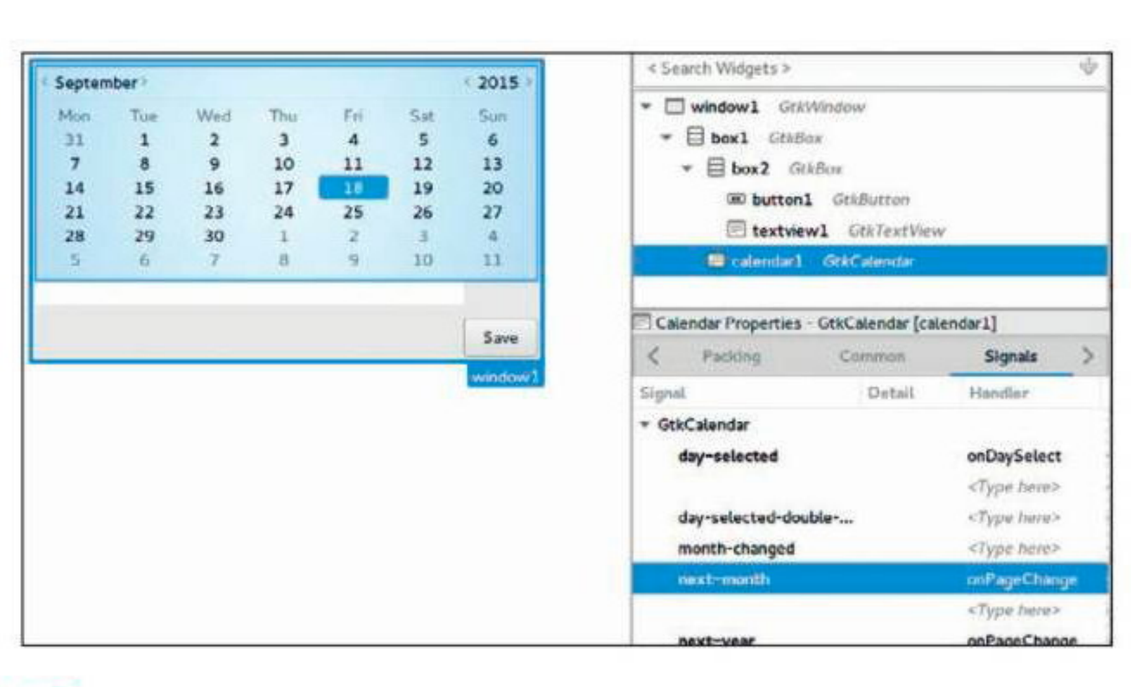
4 Размещение элементов

Во вкладке Packing в свойствах Text View (внизу справа) активируйте переключатель Expand. Тогда блок увеличится, если изменится размер окна. Во вкладке Common установите Vertical Alignment на Fill, чтобы использовать всю высоту блока. Задайте Vertical Alignment для нашей кнопки как End, иначе она будет нелепо огромной.



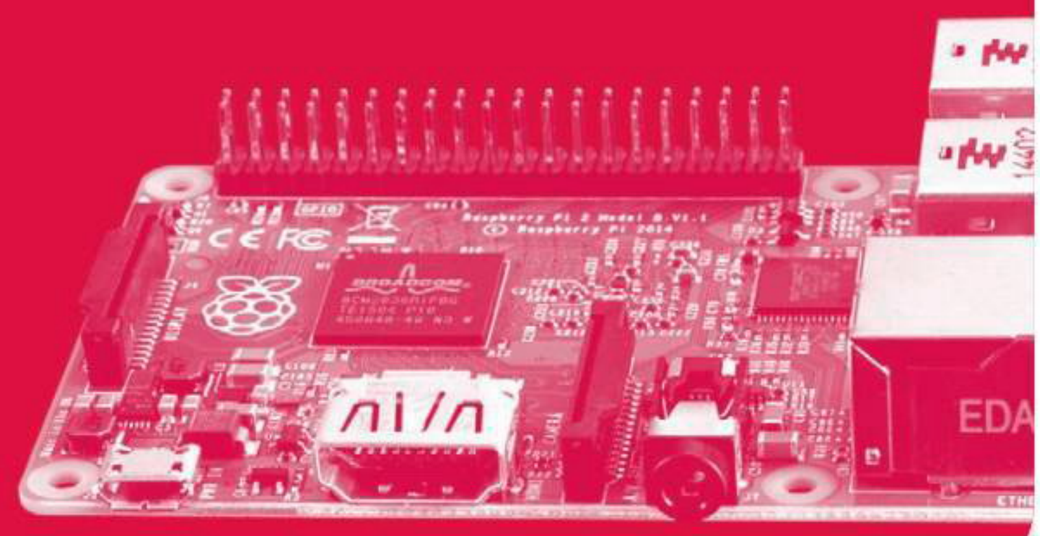
5 Соединение сигналов и обработчиков

Теперь нам нужно соединить сигналы нашего приложения с функциями Python. Это достигается определением обработчика для каждого события, которое нам важно. Нам интересно, когда пользователь решает закрыть окно, так что выберите window1 из верхней правой панели. Во вкладке Signals панели Properties найдите 'delete-event' и установите его обработчик на 'onDeleteWindow'.



6 Невозможно остановить сигналы

Нас интересует сигнал «щелчка» по кнопке; установите его обработчик на 'onButtonClick'. Для календаря понадобится дать сигналу 'day-selected' обработчик 'onDaySelect'. Наконец, также понадобится связать ряд сигналов, таких как 'month-changed', 'prev-year' и 'next-year' с обработчиком 'onPageChange'. Сохраните ваш свеженький интерфейс как **~/gtk3cal/calendar.glade**. LXF



Ваша порция смачных новостей, обзоров и учебников от Raspberry Pi

АЛЕКС ЭЙМС работает на RasPi. ТВ и среди многих других вещей спроектировал RasPiO.

Привет...

Проведя 3 успешных проекта на Kickstarter (каждый с более чем 1000 инвесторов), достигших финансовой цели задолго до срока, поделюсь с вами советами.

Чтобы преуспеть на Kickstarter, надо представить, что еще не сделано и вызывает потребность, и известить об этом потенциальных инвесторов. Крайне важно создать видео с четким разъяснением идеи.

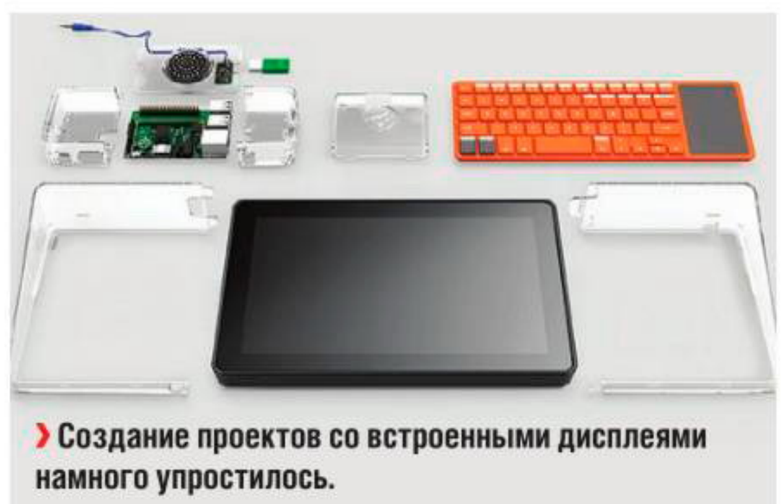
Успех краудфандинга — в хороших связях. Сумев выразить энтузиазм насчет своей идеи, вы заразите им людей. Ваша страница на Kickstarter должна быть хорошо написана и нашпигована фото — какие классные вещи делает ваш прототип; но надо также указать привлекательные пути увеличить его ценность. Странице надо вызывать интерес и апеллировать к влиятельным членам сообщества, способным продвинуть проект. Установите реалистичную финансовую цель и предложите хорошо продуманные награды участникам. HDMIPi, например, стал блокбастером, учтя все вышеперечисленное. Если лично у вас мало последователей, плените воображение тех, у кого их много. Задача Kickstarter — создать шум, но надо быть весьма специфичным, чтобы понравиться блоггерам. До запуска точно оцените свой продукт, включая стоимость доставки. И обдумайте, что будет, если вы добьетесь мега-успеха и превысите порог НДС. Запустив кампанию, работайте над ней. Довольно легко собрать £1000 (скажем, от 100 человек по £10); свыше нескольких тысяч добыть куда сложнее. Постоянно сообщайте и хорошие, и плохие новости. Людям нравится быть в курсе.

Мы в восторге от экранов Pi

На Pi обрушился ливень крошечных экранов.

Кому нужен 30-дюймовый дисплей-гигант, если у вас крошечный ПК? Ведь вышло множество великолепных новых дисплеев для Raspberry Pi. Первый из них — официальный 7-дюймовый Touchscreen Display для Raspberry Pi (http://bit.ly/LXF204piscreen) — это замечательный тачскрин с разрешением 800×480, питаемый от Pi. При цене £48 он позволит Pi оседлать многие отличные проекты.

В Kano также выпустили свой собственный набор экранов (www.kano.me/screen), и этот самосборный 10,1-дюймовый тачскрин предлагает разрешение 1280×720, но стоит куда дороже — £110. Наконец, приняли участие в этом и Pimoroni, выставив экран, включенный в набор Picade, в качестве самостоятельной единицы. Это классный 8-дюймовый TFT-экран 4:3 и 800×600 без тачскрина, причем доступный всего за £45; подробнее о нем см. на http://bit.ly/LXF204pimoroni.



› Создание проектов со встроенными дисплеями намного упростилось.

Astro Pi обрел форму

Наконец, Pi готов к полету.

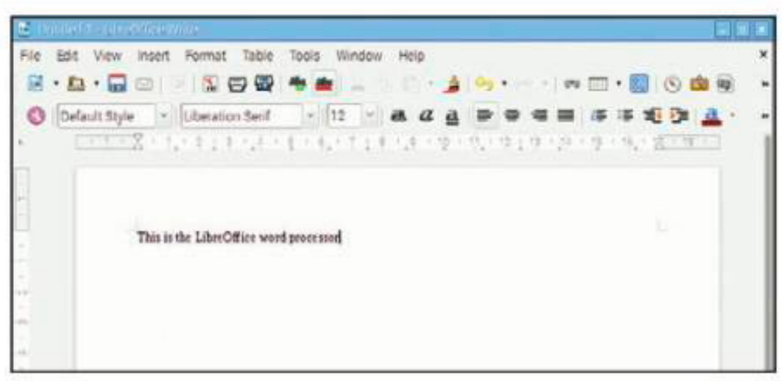
Спустя год проект по отправлению в космос варианта Pi под названием Astro Pi готов к запуску. Последней стадией были лазерное травление алюминиевого корпуса и маркировка каждого порта стандартными именами. Всего сделано 4 наземных единицы и 2 для отправки в космос. Последние две переправляются в Казахстан, чтобы лететь вместе с Тимом Пиком [Tim Peake] на «Союзе-ТМА19М» [миссия ISS-45S]. Подробности см. на http://astro-pi.org.

› Окончательная Astro Pi выглядит прочнее гвоздей.

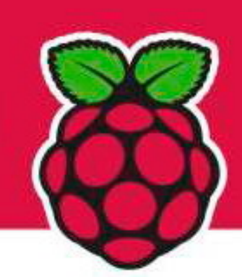
Jessie на Pi

Теперь обновлен до Debian 7.

Официальный дистрибутив Raspberry Pi, Raspbian на базе Debian, обновлен до последнего релиза Jessie, он же Debian 7. Он идет вместе со множеством исправлений и измененными «внутренностями». Pi Foundation также воспользовалась возможностью обновить интерфейс рабочего стола до *GTK 3* и портировать известные приложения, такие как популярный офисный пакет *LibreOffice* и *Claw Mail*. www.raspberrypi.org/downloads.



› Debian 7, он же Jessie, никогда еще не смотрелся так хорошо!



Дисплей Raspberry Pi

Лес Паундер наложил лапы на долгожданный официальный сенсорный дисплей Raspberry Pi — надеемся, успев облизать пальцы от малинового варенья.

Вкратце

» **Сенсорный экран разработан Raspberry Pi Foundation и обеспечивает 7-дюймовый сенсорный интерфейс для проектов. Задуман как портативный и простой в использовании для встраивания в проекты или работы в классе.**

Хотя уже продано шесть миллионов штук Raspberry Pi, у него есть один порт, никогда ранее не применявшийся: это дисплей. Он расположен над слотом микро-SD и разработан специально для недавно выпущенного официального дисплея Raspberry Pi.

Официальный дисплей Raspberry Pi оснащен 7-дюймовым экраном с разрешением 800×480 пикселей и имеет емкостной сенсорный интерфейс. Дисплей окружен черной рамкой, придающей ему вид планшета, и обращаться с ним надо осторожно: он довольно тонок и легко гнется. У дисплея нет ни чехла, ни подставки, поскольку Raspberry Pi Foundation предлагает сообществу изготовить их самостоятельно, а компаниям вроде Pimoroni — удовлетворить потребность.

На тыльной части экрана находятся два тонких разъема для видео и для сенсорного интерфейса, подключаемых к плате пульта. На этой плате — три соединительных шлейфа: два с дисплея, а третий подключается к порту дисплея Raspberry Pi. Pi можно установить поверх платы пульта, прикрепив четырьмя винтами. Питание дисплея обеспечивается через порт микро-USB, лучше от блока питания 5 В, 2 А. Pi можно запитать двумя способами: через USB — подключением кабеля микро-USB или соединив перемычками соответствующие контакты 5V и Ground [Земля] на платах пульта и Raspberry Pi; последний вариант позволит обойтись без добавочных плат.

Если используется последний официальный образ Raspbian, Pi автоматически определит правильное разрешение для дисплея.

› **Официальный 7-дюймовый дисплей Raspberry Pi — портативный и аккуратный сенсорный экран для применения в процессе обучения и технических кружках.**

Установка программного обеспечения исключительно проста и не требует большего, чем sudo apt-get update и sudo apt-get upgrade для установки драйверов сенсорного экрана. Это свежее решение в сравнении с другими дисплеями, такими как 5-дюймовый сенсорный экран Adafruit, использующий сенсорный контроллер AR1100 и требующий калибровки с помощью ПК с Windows.

Портативный сенсорный

Сенсорный интерфейс дисплея Raspberry Pi реагирует очень быстро. Мы протестировали его с Pi 2 и не ощутили никаких зависаний или подтормаживаний. Интерфейс может обнаружить десять точек касания одновременно, что позволит применить в ваших проектах возможности мультитач. Для обычного настольного применения интерфейс работает хорошо, но пока нет функций правой кнопки мыши — это ограничение X, не предназначенного для сенсорного экрана.

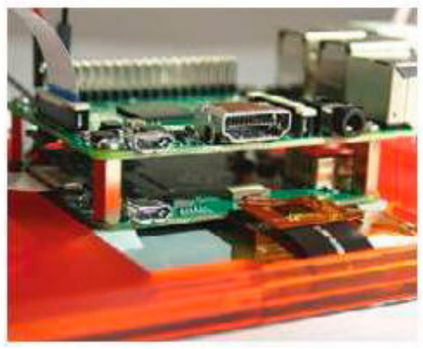
Дисплей яркий и имеет комфортный угол обзора; и хотя разрешение меньше, чем мы успели привыкнуть, для творчества в любимом приложении места всё же немало. Дисплей может использоваться с HDMI-экраном, открывая перспективу системы с двумя дисплеями, но это требует некоторых изменений настроек приложений, чтобы те выбирали правильный экран.

Пока дисплей продается в виде комплекта частей и требует небольшой сборки, хотя намечено, что будущие версии будут поставляться готовыми. Он не заменяет основной монитор, но удовлетворяет две потребности. Во-первых, необходимость портативного экрана для использования в технических кружках и классах. Размер дисплея и его цена для этого хорошо подходят. Во-вторых, дисплей легко встроить в проект, позволяющий на базе Pi создать ряд сенсорно управляемых решений.

Официальный 7-дюймовый дисплей Raspberry Pi — отличная часть комплекта. Поскольку он работает как сенсорный экран, в ближайшие месяцы мы увидим его во множестве отличных проектов. LXF

Свойства навскидку

Плата пульта
Управляет дисплеем и сенсорным интерфейсом и обеспечивает питание и интерфейс I2C для вашего Raspberry Pi.



Простая сборка
Дисплеи поставляются в разобранном виде как комплект частей, но их сборка не потребует от вас специальных знаний.

LINUX FORMAT Вердикт

Дисплей Raspberry Pi

Разработчик: Raspberry Pi Foundation
Сайт: www.raspberrypi.org
Цена: £48

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

» С невысоким разрешением, без подставки или чехла, но предлагает экономичное встроенное сенсорное управление для проектов.

Рейтинг 8/10

Astro Pi: Основы

Лес Паундер осваивает плату Sense HAT, которая отправится на Международную космическую станцию с космонавтом Тимом Пиком.

Наш эксперт

Лес Паундер — странствующий маэстро Pi. Он кочует по Великобритании, проповедуя Путь Pi от имени щедрого фонда Raspberry Pi Foundation. В свободное время увлекается хакерством и пишет об этом на http://bigl.es.

Скорая помощь

Мечтаете изменять цвет текста, но не умеете смешивать значения цвета RGB? Эти значения можно найти во многих графических редакторах, например, в *GIMP*. Поэкспериментируйте с различными цветами, запишите их значения и попробуйте их на Sense HAT.

Sense HAT — замечательная плата с датчиками температуры, влажности и давления. Она также имеет акселерометр, гироскоп и компас, пригодные для определения местоположения. В довершение всего на плате имеется светодиодная матрица 8×8 и джойстик. В данном проекте мы используем это чудесное устройство для измерения температуры, влажности и давления и создадим анимации на матрице.

Кроме Sense HAT, вам потребуется Raspberry Pi 2, A+ или B+, дистрибутив Raspbian и подключение к Интернету. Весь код для этого проекта можно найти на http://bit.ly/LXF204AstroPi. Установить Sense HAT поверх выключенного Pi очень легко; затем подключите все кабели к Pi и загрузитесь в рабочий стол Raspbian. Для работы с Sense HAT надо установить его программное обеспечение. Для этого откройте *LXTerminal* (его значок находится в верхней левой части экрана, он напоминает монитор) и введите в нем следующие команды:

```
$ sudo apt-get update
$ sudo apt-get install sense-hat
$ sudo pip-3.2 install pillow
```

Это мы обновляем список доступного ПО до актуального состояния, а затем устанавливаем программы Sense HAT. Мы используем менеджер пакетов *pip* для установки *pillow*, ветки *Python Imaging Library* (*PIL*). После установки потребуется перезагрузка и возврат на рабочий стол Raspbian. Вновь открыв *LXTerminal*, введите следующее, чтобы запустить *IDLE* для Python 3:

```
$ sudo idle3 &
```

В *IDLE3* нажимаем File > New, чтобы создать новый пустой документ. Рекомендуется сразу сохранить свою работу, поэтому щелкните по File > Save и сохраните вашу работу как **sensors.py**. Начнем создание проекта с импорта библиотек:

```
import pygame
from pygame.locals import *
from sense_hat import SenseHat
import time
```

Сначала импортируем библиотеку *pygame* с сопутствующей библиотекой *pygame.locals*. Используем библиотеку *pygame.locals* при обнаружении действия джойстика. Импортируем библиотеку *SenseHAT*, чтобы иметь возможность использовать плату HAT, и, наконец, импортируем функцию time, чтобы управлять темпом нашего проекта. Создадим переменную с именем sense — sense = SenseHat() и припасем функцию SenseHat, чтобы поменьше набирать, после чего инициализируем библиотеку *pygame* для использования с pygame.init().

В следующей части кода мы создаем ряд функций, которые будут обрабатывать действия в проекте. Начнем с отображения картинки на экране:

```
def image():
  pygame.display.set_caption("Linux Format представляет...")
  picture = pygame.image.load("image.png")
  screen = pygame.display.set_mode((688,361))
  screen.blit(picture,(0,0))
  pygame.display.flip()
```

Вызовем функцию image — она создаст окно с заголовком. Затем создадим переменную с именем picture и используем ее, чтобы сохранить готовое изображение. Далее создадим еще одну переменную, с именем screen, и установим размер экрана по размеру изображения, которое мы будем загружать. Потом сольем (blit) это изображение в память, чтобы быстро обновить содержимое памяти с данными изображения. Последним мы обновляем экран с помощью функции pygame.display.flip().

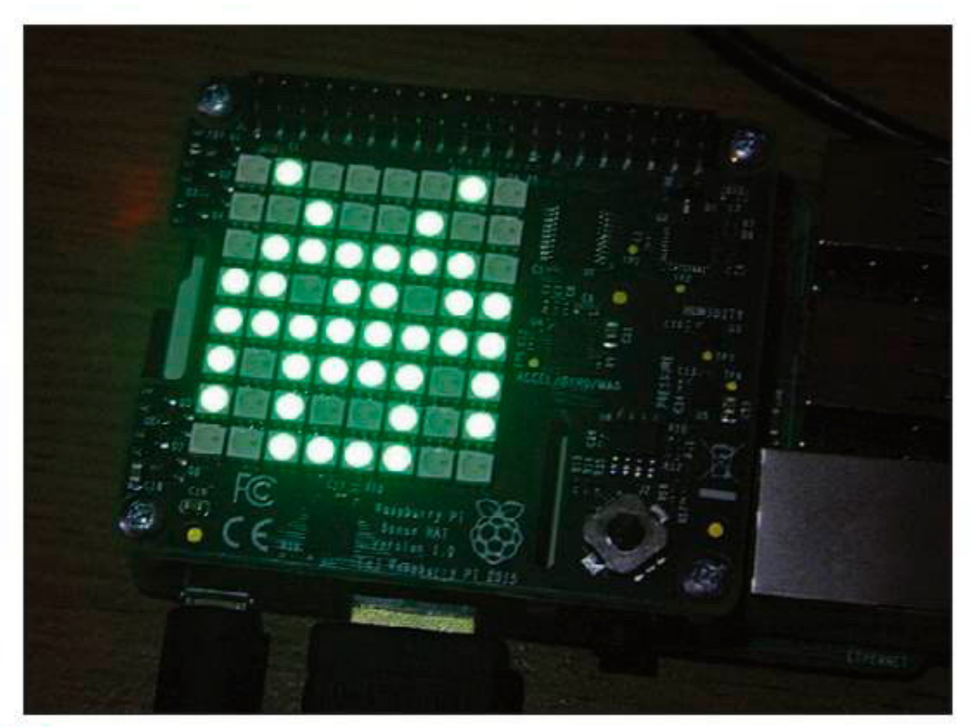
Функции температуры и джойстика

Наша следующая функция обрабатывает сигналы джойстика. Используем условный оператор if...elseif для обработки действий четырех событий — нажатие кнопок вверх, вниз, влево и вправо:

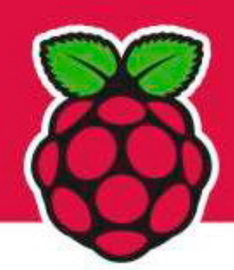
```
def joystick(event):
  if event.key == pygame.K_DOWN:
    invader()
  elif event.key == pygame.K_UP:
    pressure()
  elif event.key == pygame.K_LEFT:
    temperature()
  elif event.key == pygame.K_RIGHT:
    humidity()
```

Когда условие станет равным true, вызовется соответствующая функция. Далее мы создаем функцию temperature — эта функция считывает текущую температуру и сохраняет ее как переменную с именем temp:

```
def temperature():
  temp = round(sense.get_temperature(),1)
  if temp < 20:
    sense.show_message(" Температура %s C" % temp,
text_colour=[0,0,255])
  elif temp > 20 and temp < 30:
    sense.show_message(" Температура %s C" % temp,
text_colour=[0,255,0])
```



› Sense HAT вписывается в Pi и оставляет много места для подключения новых дисплея и камеры Pi.

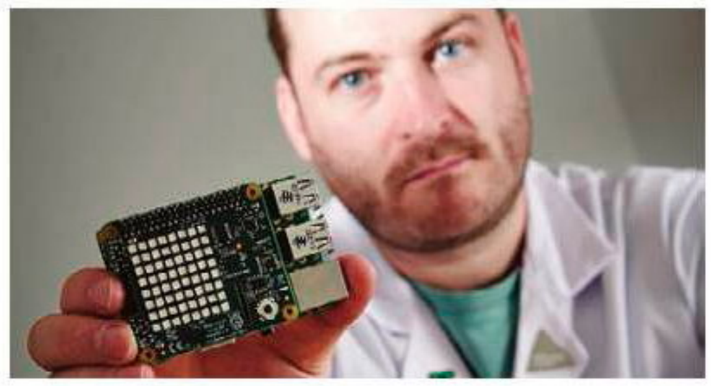


Проект Astro Pi

Astro Pi — совместный проект Raspberry Pi Foundation и Европейского космического агентства (EKA). Проект создал платформу аппаратных датчиков Sense HAT, которая будет отправлена на Международную космическую станцию (МКС), где космонавт EKA Тим Пик проведет ряд экспериментов с программами на Python, написанными школьниками. Проекты были созданы в рамках конкурса для начальных и средних школ по всей Великобритании.

Код будет тщательно проверен на ошибки совместно с абсолютно исчерпывающим тестом аппаратных средств, чтобы гарантировать, что и HAT, и Pi готовы к полету на МКС.

Код, написанный победителями, Тим Пик запустит во время полета по орбите на борту МКС, а результаты будут доступны всем. Этот проект позволит детям повторить эксперименты на Земле и сравнить их с результатами на МКС. Проект вызвал огромный интерес в космическом сообществе, особенно в НАСА, где пристально следят за результатами проекта с целью повторить его в будущем конкурсе для детей всего мира. Потенциально Astro Pi стимулирует изучение наук, технологий, инженерии и математики, и Pi может вдохновить новое поколение программистов, инженеров, математиков и ученых.



› Дэйв Хонесс [Dave Honess] работал над проектом Astro Pi более года и сотрудничал с EKA по сертификации проекта на пригодность к полету.

```
else:
  sense.show_message("Температура %s C" % temp,
text_colour=[255,0,0])
```

Можно видеть, что мы используем round() для округления значения температуры до одной десятой, давая нам точное значение. Далее мы сравниваем значение temp, используя условный оператор if...elseif так, что при температуре ниже 20°C текст выделяется синим цветом. Если температура больше 20°C и меньше 30°C, текст зеленый, а на все остальное мы получаем красный текст. Для каждого состояния мы пишем текст на светодиодной матрице, используя функцию show_message. Мы включаем значение temp в текст с помощью %s, что велит Python'у заменить значение этой переменной на переменную temp с помощью % temp. Также можно изменить цвет текста, используя цветовую схему RGB (Red, Green, Blue — красный, зеленый, синий).

Функции влажности и давления

Далее создадим функцию для измерения влажности. При этом используется структура для хранения и отображения информации, аналогичная той, что мы употребили для функции температуры:

```
def humidity():
  humid = round(sense.get_humidity(),1)
  sense.show_message("Влажность: %s %%" % humid)
```

Повторно используем структуру функции влажности, чтобы создать еще одну функцию для получения и отображения локального давления воздуха:

```
def pressure():
  pressure = round(sense.get_pressure(),1)
  sense.show_message("Давление %s миллибар" % pressure)
```

Наша последняя функция — для забавы: она рисует на светодиодной матрице анимированного космического пришельца.

```
def invader():
  for i in range(8):
    x = [0,255,0]
    o = [0,0,0]
    invader = [
      o,x,o,o,o,o,x,o,
      o,o,x,o,o,x,o,o,
      o,x,x,x,x,x,x,o,
      x,x,o,x,x,o,x,x,
      x,x,x,x,x,x,x,x,
      x,o,x,x,x,x,o,x,
      x,o,x,o,o,x,o,x,
      o,o,x,x,x,x,o,o,
    ]
    sense.set_pixels(invader)
    time.sleep(0.5)
```

Функция invader использует цикл for, повторяющийся восемь раз. Далее создаем два списка с именами x и o, где x присваивается значение 0,255,0, т. е. зеленый в RGB. Для o используем значение 0,0,0 или «нет цвета», по сути выключая светодиод. Создаем список invader, где храним восемь строк и восемь столбцов x и o. Где x, там на матрице отображается зеленый цвет. Python задает пиксели, используя список invader в качестве шаблона. После задержки в 0,5 секунды мы изменяем конфигурацию списка invader, затем перезагружаем пиксели, чтобы показать изменения. После последней итерации цикла экран очищается. Здесь приведен не весь код, но это есть в заключительном примере кода.

Следующий блок кода отвечает за перевод светодиодной матрицы в режим слабого излучения, слегка приглушая яркие светодиоды, чтобы не слепили глаза. Затем вызываем функцию image() для отображения экрана с инструкциями и очищаем светодиодную матрицу перед выводом сообщения «ГОТОВО» зеленого цвета.

```
sense.low_light = True
image()
sense.clear()
sense.show_message("ГОТОВО",text_colour=[0,255,0])
```

Наконец, цикл будет постоянно искать ввод от джойстика, и при его появлении происходит событие, вызывающее одну из соответствующих функций, созданных нами ранее. Мы также создаем метод выхода, который можно вызвать, закрыв окно с инструкциями или нажав Ctrl+c.

Проект закончен. Сохраните вашу работу и нажмите на Run > Run Module для запуска кода. Поздравляем! Вы сделали первые шаги с Sense HAT и создали инструмент для измерения нескольких видов данных. LXF

Выполнение кода

В *IDLE* выполнить код легко, но наступит момент, когда вы захотите сделать свой код исполняемым. Простейшим способом будет добавить #!/usr/bin/env python3 первой строкой в проект **sensors.py**. Затем откройте *LXTerminal* и перейдите в каталог, где находится **sensors.py**.

Чтобы сделать файл исполняемым, необходимо изменить его права доступа, и мы сделаем это командой

```
$ chmod +x sensors.py
```

Теперь у нас есть файл, готовый к исполнению, и мы запустим его в том же терминале, введя $ sudo ./sensors.py.

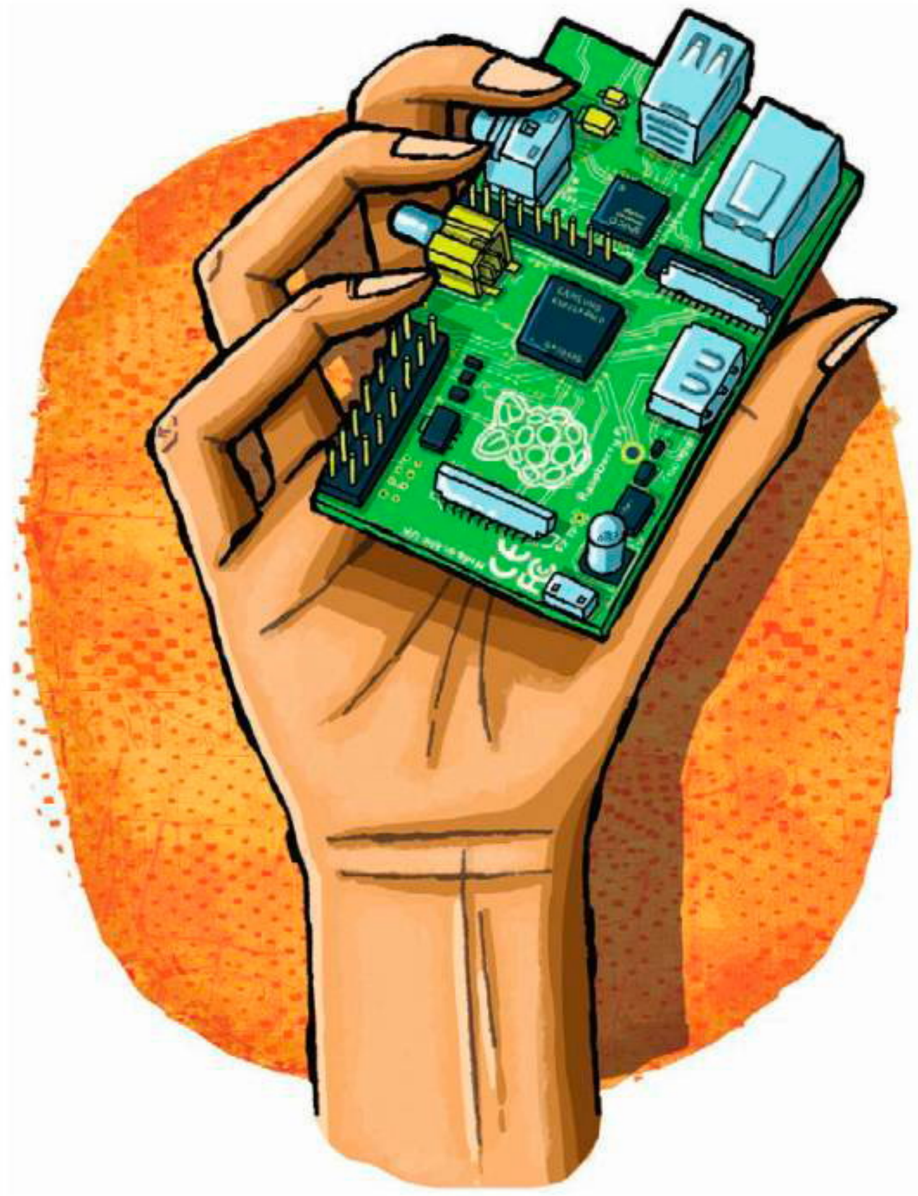
» **Подпишитесь на печатную или электронную версии** на www.linuxformat.ru/subscribe!

CUPS: Печать

Сделать Raspberry Pi беспроводным сервером печати очень просто, имея Raspberry Pi и *CUPS*. **Маянк Шарма** вам всё расскажет.

Наш эксперт

Маянк Шарма был пишущим редактором на www.linux.com и сотрудничал с Linux Today, Digg и PC Plus.



Принтеры — не самые удобные из периферийных устройств. Они выглядят чужеродно на большинстве письменных столов и создают изрядный шум, выплевывая страницы. Лучше отвалить пару-другую сотен фунтов на шикарный новенький сетевой принтер, который будет тихо сидеть где-нибудь в уголке и получать заказы на печать с любого компьютера в локальной сети; или просто подцепить свой обычный USB-принтер к Raspberry Pi и наслаждаться тем же удобством, которое предлагают лучшие сетевые принтеры.

Если вы еще не пользовались своим принтером в Linux, то прежде чем начинать этот проект, стоит заглянуть на www.openprinting.org/printers и проверить, совместим ли ваш принтер с ПО сервера печати *CUPS*. Если ваш принтер в списке совместимых, подключите его к Raspberry Pi с помощью одного из USB-портов. Для нашего проекта мы использовали дистрибутив Raspbian на Pi, соединенном с локальной сетью через совместимый беспроводной адаптер. Однако вы также можете присоединить свой Pi через проводной порт Ethernet.

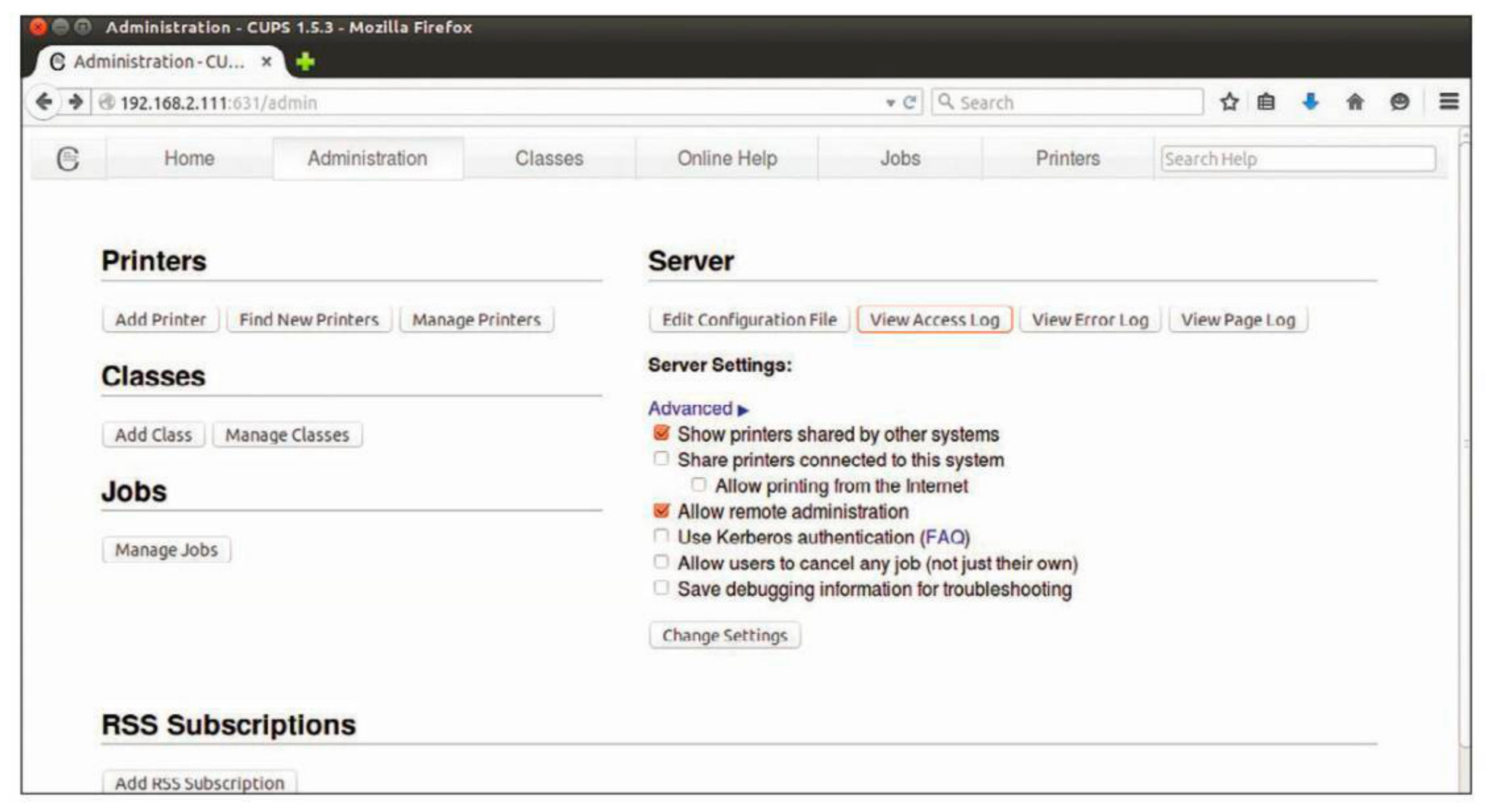
Вы можете следовать инструкциям данного урока, получив доступ к Raspberry Pi удаленно с любого другого компьютера в сети. Просто активируйте SSH-сервер внутри Raspbian инструментом *raspi-config*. Также хорошей идеей будет присвоить вашему Pi фиксированный IP-адрес, что вы с легкостью сможете сделать со страницы администрирования вашего роутера. На этом уроке мы будем предполагать, что IP-адрес вашего Pi — **192.168.3.111**. Теперь вы можете получить доступ к Pi из Windows с помощью клиента *PuTTY* или из любого дистрибутива Linux с помощью команды SSH CLI:

```
$ sudo ssh pi@192.168.3.111
```

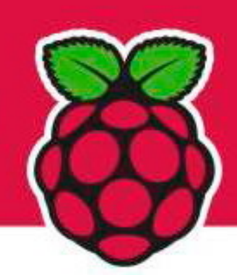
Установка CUPS

Итак, зайдя в Raspbian, обновите репозитории с помощью `$ sudo apt-get update` и затем установите все обновления командой `$ sudo apt-get upgrade`. Затем установите сервер печати *CUPS* — `$ sudo apt-get install cups`.

После его установки добавьте нового пользователя в группу, созданную *CUPS*, под названием lpadmin, имеющую доступ к очереди на печать. Если вы не создали собственного пользователя, имя пользователя по умолчанию в Raspbian — pi. Используйте



› Вы также можете просмотреть его обширную документацию из работающей через браузер панели управления *CUPS*.



Утилиты командной строки CUPS

Система печати *CUPS* поставляется с рядом грамотных утилиток командной строки. В принципе, из CLI можно настроить все аспекты вашего принтера. Давайте пробежимся по самым полезным командам, которые помогут вам лучше управлять принтером.

Мы уже видели команду lp, которая ставит файл в очередь на печать на принтере по умолчанию. Принтер по умолчанию указан в переменной PRINTER. Вы можете указать его с помощью команды export PRINTER=printer-name, где printer-name — имя принтера, который вы указываете на шаге 2 инструкции (см. стр. 94). Если у вас несколько принтеров, используйте опцию -d, чтобы указать принтер, на котором хотите печатать. Например, lp -d HP-printer file.txt печатает файл на принтере HP, который не указан как принтер по умолчанию.

Чтобы повлиять на характеристики вывода принтера, укажите разные параметры через опцию -o. Например, lp -o landscape -o fit-topage -o media A4 file.jpg размещает файл на A4 и печатает его в режиме альбома. Обратитесь к документации *CUPS* (www.cups.org/documentation.php/options.html#OPTIONS) за списком всех поддерживаемых опций печати.

Если вы по ошибке отправили на печать большой файл и хотите остановить печать, пока не ухлопали слишком много бумаги, выведите список задач печати в очереди командой lpq.

Команда также выводит файл, печатаемый каждой задачей, и его размер, и вы легко определите ID задачи, присвоенной каждому файлу. Запомните его, потому что он понадобится вам для отмены задачи печати. Например, cancel 2 отменяет задачу с ID 2.

следующую команду, чтобы позволить ему взаимодействовать с принтером: $ sudo usermod -a -G lpadmin pi. Здесь мы используем инструмент *usermod*, чтобы добавить (-a) пользователя pi в группу lpadmin (-G). По умолчанию *CUPS* может быть настроен только с локального компьютера, на который он установлен. Поскольку в нашем случае это не работает, надо отредактировать его файл настроек, чтобы позволить себе вносить изменения на сервер с удаленного компьютера. В первую очередь надо создать резервную копию файла изначальной настройки с помощью

```
$ sudo cp /etc/cups/cupsd.conf /etc/cups/cupsd.conf.orig
```

Затем откройте файл с помощью текстового редактора *nano*: $ sudo nano /etc/cups/cupsd.conf. Внутри файла прокрутите вниз до следующего раздела:

```
# Слушать подключения только с локальной машины
Listen localhost:631
```

Раскомментируйте эту строку и добавьте другую, чтобы попросить *CUPS* принимать соединения с любого компьютера в сети. Убедитесь, что раздел выглядит как-то так:

```
# Слушать подключения только с локальной машины
# Listen localhost:631
Port 631
```

Прокрутите дальше вниз в файле настройки, пока не увидите раздел Location, и добавьте новую строку, которая читается как Allow @local, перед самым закрытием раздела. Раздел с прикрепленной строкой должен выглядеть примерно так:

```
< Location / >
# Ограничить доступ к серверу
Order allow,deny
Allow @local
< /Location >
```

Теперь добавьте строку Allow @local к двум другим разделам Location — Location /admin и Location /admin/conf. Сохраните файл и перезагрузите сервер *CUPS* с помощью $ sudo /etc/init.d/cups restart.

Отныне вы должны быть в состоянии получить доступ к панели администрирования *CUPS* через любой компьютер в вашей локальной сети, указав браузеру на свой Pi. Затем следуйте инструкциям на странице, чтобы добавить ваш принтер в *CUPS*.

Некоторые дистрибутивы Linux поставляются с политикой ограничивающего брандмауэра *iptables*, которая не позволяет соединяться через порты *CUPS*. Raspbian такого не делает, но все же убедитесь, что он не выкидывает неожиданные ошибки, пробивая дыры в брандмауэре, с помощью:

```
$ sudo iptables -A INPUT -i wlan0 -p tcp -m tcp --dport 631 -j ACCEPT
$ sudo iptables -A INPUT -i wlan0 -p udp -m udp --dport 631 -j ACCEPT
```

Если вы подключены к Raspberry Pi через Ethernet, а не беспроводной адаптер, измените команды, заменив wlan0 на eth0. После настройки принтера через панель администрирования *CUPS* пора будет сделать его доступным для других компьютеров в вашей сети. Дистрибутивы Linux без проблем будут определять ваш новый сетевой принтер, но чтобы сделать его видимым для устройств Windows и Apple, придется предпринять еще пару шагов.

Доступ по всей сети

Для Windows установите сервер *Samba* на Pi с помощью $ sudo apt-get install samba. Затем откройте его файл настроек (**/etc/samba/smb.conf**) в текстовом редакторе *nano*, найдите раздел, помеченный [printers], и проверьте в нем наличие строки

```
guest ok = yes
```

Далее прокрутите вниз до раздела [print$] и измените его путь на следующий:

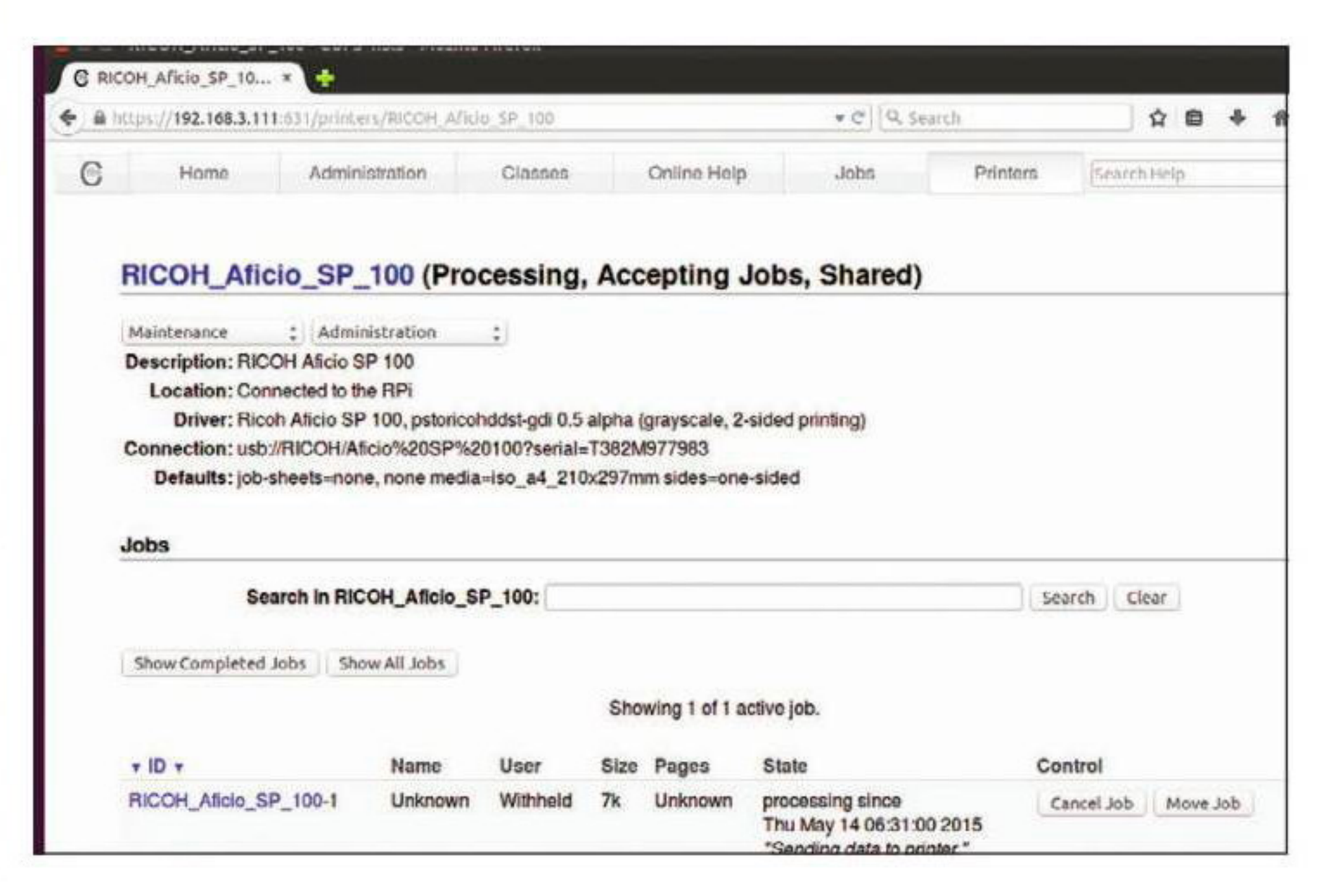
```
path = /usr/share/cups/drivers
```

Затем прокрутите вверх до раздела Global Settings в верху файла настроек. Измените параметр рабочей группы внутри него, чтобы указать на имя вашей рабочей группы, которая по умолчанию называется WORKGROUP. Включите также поддержку wins, добавив строку wins support = yes.

Сохраните файл и перезагрузите *Samba*, с помощью

```
$ sudo /etc/init.d/samba restart.
```

Затем перейдите на компьютер с Windows, запустите мастера *Add New Printer* и щелкните по опции установки сетевого принтера. Благодаря измененным настройкам *Samba* мастер обнаружит и перечислит все принтеры, присоединенные к Pi. Если у вас есть устройства Apple, можете включить поддержку для системы Apple AirPrint, разрешающую печать с iPad и iPhone. Для этого просто установите демон *Avahi* с помощью sudo apt-get install avahi-daemon на Pi, после чего присоединенный принтер станет видимым для совместимых с AirPrint устройств. »



› Из вкладки Printers вы можете отслеживать статус каждой задачи каждого принтера.

Вдобавок к возможности использовать наш сетевой принтер из графических приложений на всех платформах, мы также можем использовать его для печати из интерфейса командной строки. Более того, можно еще и взаимодействовать с принтером с помощью языка программирования Python.

Печать из Python

Сервер печати *CUPS* устанавливает набор инструментов командной строки (см. *Администрирование CUPS* на стр. 95) для взаимодействия с сервером и всеми подключенными принтерами. Вы можете отправлять файлы принтеру с помощью команды lp, такой как

```
$ lp ~/docs/a_text_file.txt
```

Если у вас несколько принтеров, можете печатать на указанном принтере с указанным именем, например:

```
$ lp ~/docs/another-text.txt -d EPSON_LX-300
```

Когда вы используете команду с PDF-файлом или файлом изображения, *CUPS* конвертирует файл с помощью драйверов принтера. Можно также использовать Python для создания дружественного принтеру контента. Лучше всего это делать с помощью библиотеки *PyCups*, обеспечивающей привязки Python для сервера *CUPS*. Установите библиотеку командой

```
$ sudo apt-get install python-cups
```

Затем создайте скрипт Python **example.py**, вот такой:

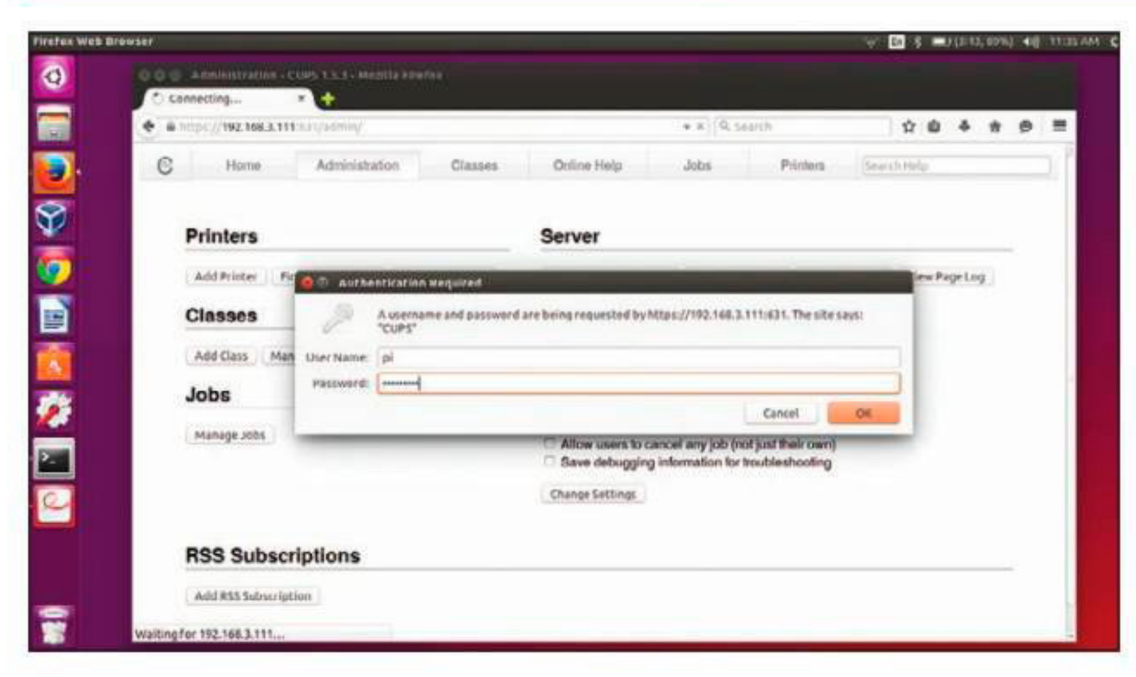
```
import cups
conn = cups.Connection()
printers = conn.getPrinters ()
for printer in printers:
  print printer, printers[printer]["device-uri"]
```

Скрипт извлекает подробности обо всех принтерах, управляемых *CUPS*, и выводит их имя и адрес устройства на экран. При выполнении скрипт создает вывод, похожий на следующий:

```
EPSON_LX-300 usb://EPSON/LX-300+?serial=L010209081
RICOH_Aficio_SP_100 usb://RICOH/Aficio?serial=T382M977983
```

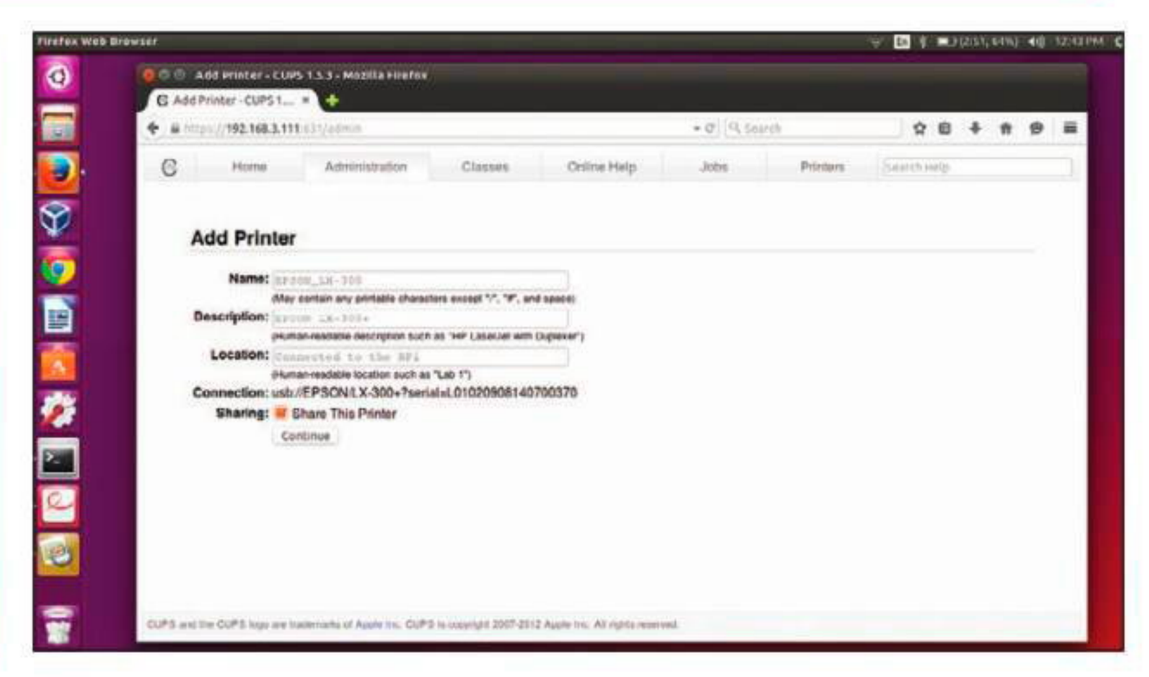
Вы также можете печатать файлы из скрипта Python с помощью функции printFile, указывая ее в формате

Добавляем принтер



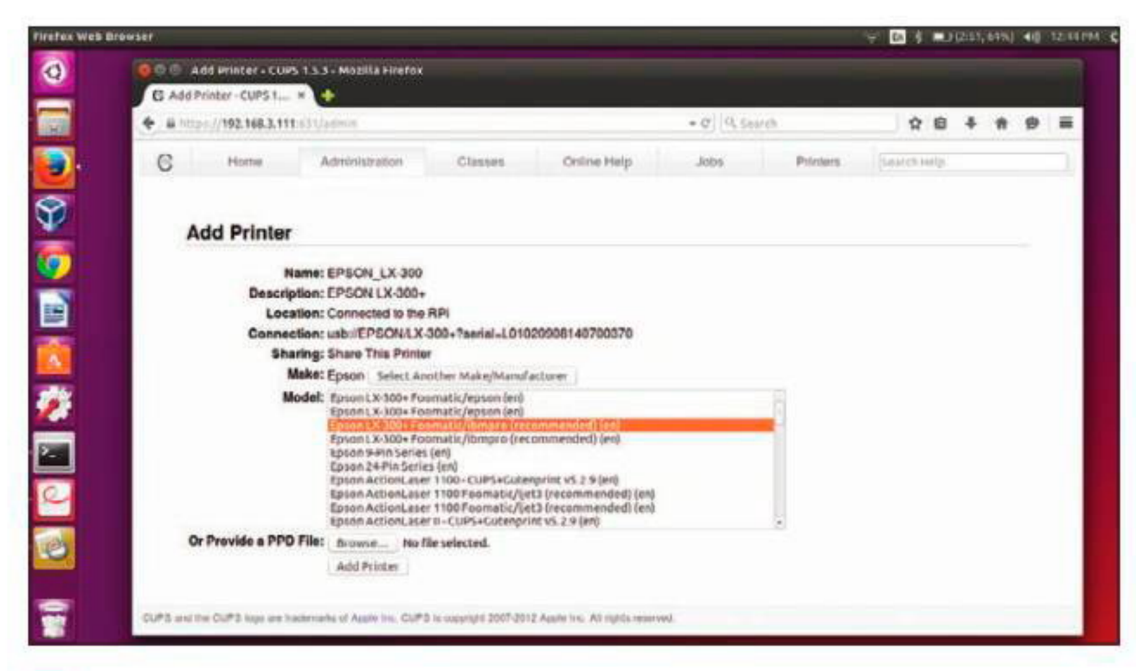
1 Приборная доска CUPS

В сервер печати *CUPS* встроен web-сервер, который управляет его панелью настройки. Он работает на порте 631 в Raspberry Pi, который в нашем случае — 192.168.3.111:631. Перейдите по этому адресу из любого браузера в сети. Вам придется принять его защитный сертификат и затем войти в интерфейс с помощью идентификационных данных пользователя, добавленного вами в группу lpadmin, который, в нашем случае, pi.



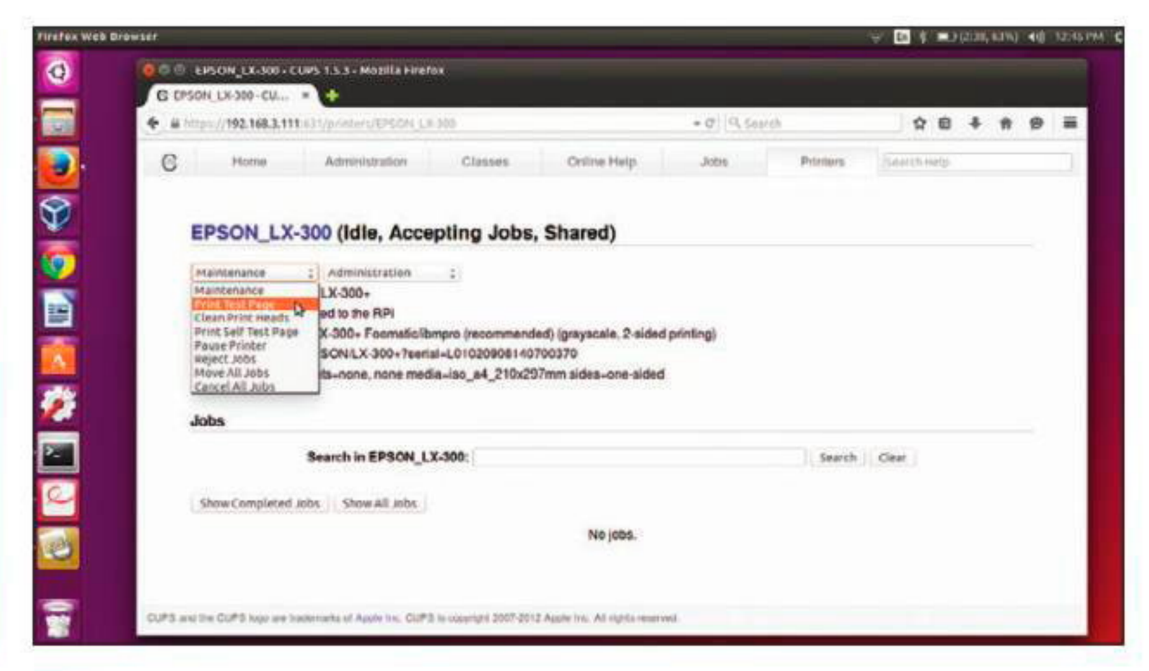
2 Добавьте принтер

После того, как вы войдете, перейдите на вкладку Administration и нажмите на кнопку Add Printer, которая выведет список принтеров. Поставьте переключатель в положение вашего принтера и перейдите к следующему шагу. Здесь вас попросят добавить или изменить имя, описания или расположение принтера. Позаботьтесь включить опцию Share This Printer, чтобы сделать принтер доступным для всей своей сети.



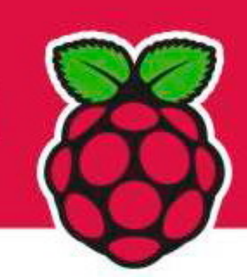
3 Выберите драйвер

На следующем шаге вас попросят указать драйверы для выбранного принтера. *CUPS* покажет вам список драйверов, основанных на марке принтера. Скорее всего, некоторые из драйверов будут помечены Recommended. Тем не менее, прокручивайте список до тех пор, пока не найдете драйвер именно для вашей модели. Как вариант, если у вас есть PPD-файл для драйвера принтера, нажмите на кнопку Browse и перейдите к нему.



4 Установите настройки по умолчанию

На финальном шаге *CUPS* позволяет установить некоторые общие настройки печати, такие как размер страницы и источник. Количество и тип опций варьируются от принтера к принтеру и могут различаться по разным разделам. Закончив устанавливать свои предпочтения, нажмите Set Default Options. Вы перенесетесь на главную страницу администрирования для этого принтера. Воспользуйтесь выпадающим меню Maintenance, чтобы напечатать тестовую страницу.



```
$ printFile (имя принтера, имя файла для печати, название работы, опции)
```

Откройте предыдущий скрипт **example.py** и добавьте к нему следующие строки:

```
file = "/home/pi/testfile.txt"
printer_name=printers.keys()[0]
conn.printFile (printer_name, file, "Project Report", {})
```

Первая сохраняет имя файла, который вы хотите напечатать, внутри переменной с именем file. Вторая извлекает список принтеров и сохраняет первое имя — это принтер по умолчанию, записанный в переменной printer_name. Третья затем использует первые две переменные и печатает файл в указанном формате.

Конвертация из HTML в PDF

Более интересный способ конвертировать HTML-страницы в PDF-файлы — использовать инструментарий *wkHTMLtoPDF*, который передает PDF-файл на принтер из скрипта Python.

Прежде чем устанавливать инструментарий, установите сперва требуемые компоненты и набор шрифтов для обработки web-страниц:

```
$ sudo apt-get install xvfb xfonts-100dpi xfonts-75dpi xfonts-scalable xfonts-cyrillic
```

Затем установите инструмент, командой

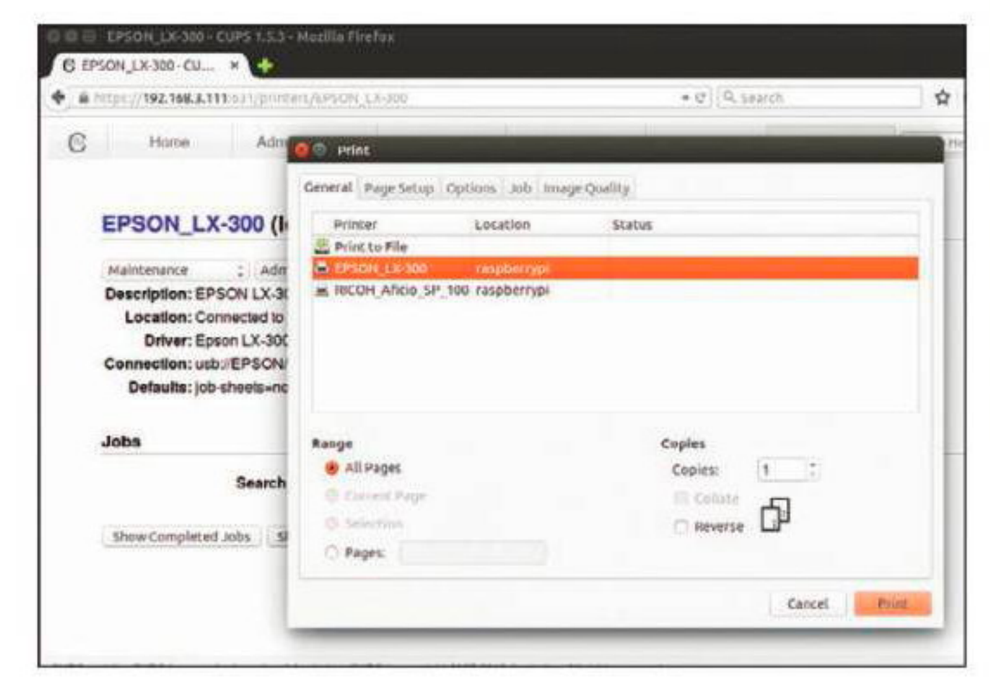
```
sudo apt-get install wkhtmltopdf
```

и надстройку Python, с помощью

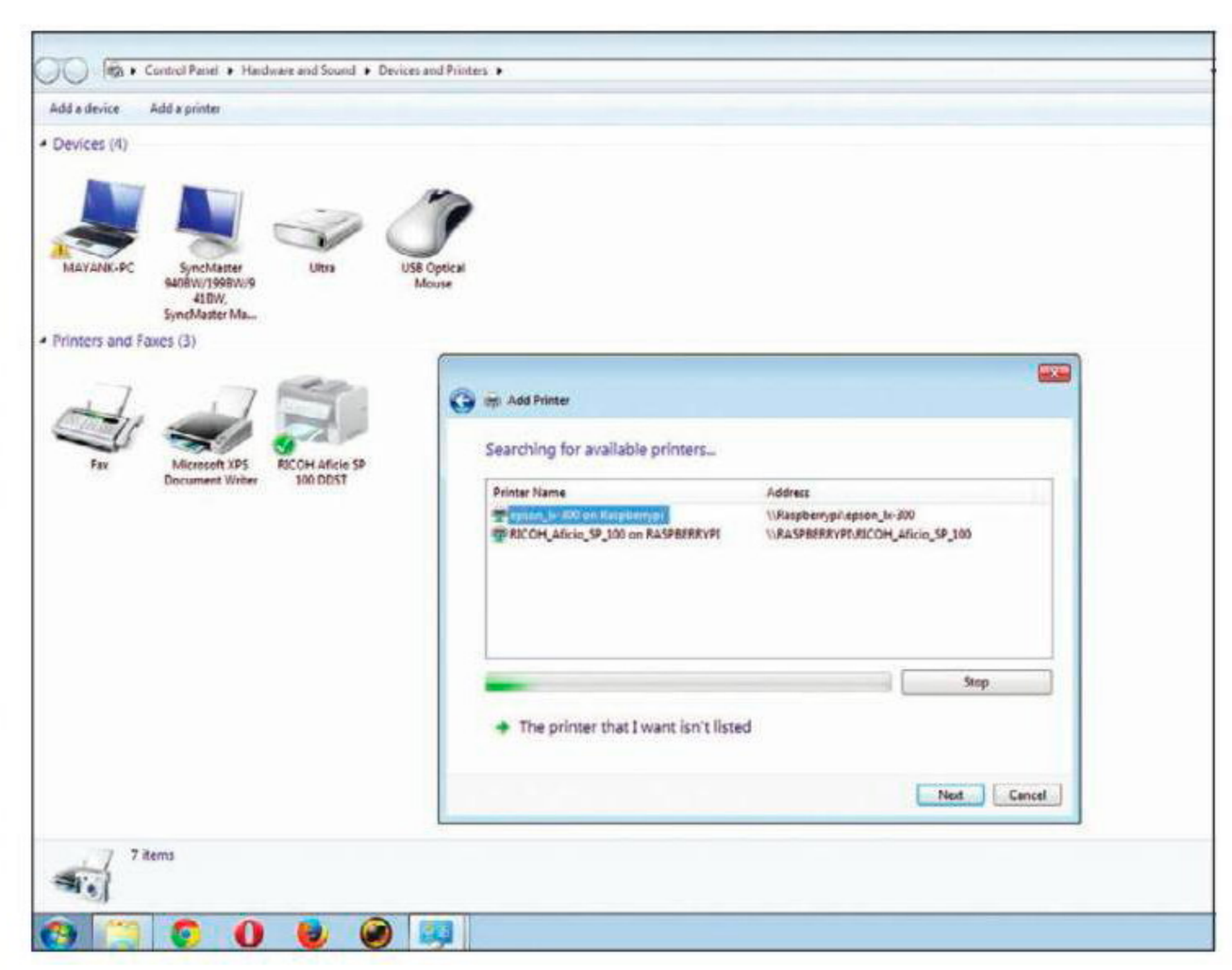
```
$ sudo pip install git+https://github.com/qoda/pythonwkhtmltopdf.git
```

Следующий код конвертирует web-страницу в PDF-файл:

```
from wkhtmltopdf import WKHtmlToPdf
wkhtmltopdf = WKHtmlToPdf (
  url='http://www.linuxformat.com',
  output_file='/home/pi/docs/lxf.pdf',
)
```



› Все дистрибутивы Linux получают доступ к USB-принтерам, подключенным к Raspberry Pi, без перенастройки дистрибутива.



› Для доступа к сетевым принтерам в Windows придется установить и настроить *Samba*.

```
wkhtmltopdf.render()
```

При выполнении этот код сохраняет главную страницу сайта *Linux Format* как PDF-файл в директории **/home/pi/docs**.

Листинг внизу показывает, как все кусочки складываются вместе — *wkHTMLtoPDF* конвертирует страницу в PDF и печатает ее.

```
#!/usr/bin/env python
import cups
from wkhtmltopdf import WKHtmlToPdf
wkhtmltopdf = WKHtmlToPdf(
  url='http://www.techradar.com',
  output_file='/home/pi/techradar.pdf',
)
wkhtmltopdf.render()
conn = cups.Connection()
printers = conn.getPrinters()
for printer in printers:
  print printer, printers[printer]["device-uri"]
file="/home/pi/techradar.pdf"
printer_name=printers.keys()[0]
conn.printFile (printer_name, file, "PDF Print", {})
```

Сперва скрипт конвертирует домашнюю страницу www.techradar.com в PDF, затем соединяется с *CUPS*, выводит список прикрепленных и настроенных принтеров на экран и использует принтер по умолчанию для печати PDF. Библиотека *PyCups* (https://pythonhosted.org/pycups) полна методов, которые вы можете использовать, чтобы управлять всеми аспектами сервера печати *CUPS*. Удачного кодинга! LXF

Администрирование CUPS

Кроме добавления принтеров, web-интерфейс *CUPS* обеспечивает доступ к другим полезным настройкам. Большинством печатных задач можно управлять из вкладки Administration, содержащей параметры настройки под различными категориями. В разделе Server, например, имется опция для отлаживания настроек сервера, а также приведены различные типы протоколов доступа и ошибок.

Используя кнопку Manage Printers в разделе Printer, можно управлять настройками индивидуальных принтеров. Каждая страница принтера содержит опции, находящиеся в двух выпадающих меню под названиями Maintenance и Administration. Из меню Maintenance можно напечатать тестовую страницу, страницу автотестирования, очистить головки принтера и управлять заданиями на печать.

Настроить поведение принтера — отладить его настройки по умолчанию, установить его в качестве принтера по умолчанию, ограничить доступ пользователя, изменить его настройки или целиком удалить его с сервера *CUPS* — поможет меню Administration. Помимо вкладки Administration, есть пара других важных вкладок, которые нам также стоит упомянуть.

Если вы новичок, переключитесь на вкладку Classes для управления классом принтера. Класс — это набор из нескольких принтеров. При отправке задания на печать классу *CUPS* автоматически отдает задание следующему свободному принтеру, а не ждет, пока освободится указанный.

Затем идет вкладка Jobs, позволяющая наблюдать и управлять всеми заданиями на печать, которые сейчас находятся в очереди.

Ответы

Есть вопрос по открытому ПО? Пишите нам по адресу answers@linuxformat.ru, и мы найдем ответ.

В этом месяце мы ответим на вопросы про...

1. Дистрибутив грузится только в командную строку
2. Понижение версии Wine
3. Дурное поведение Clonezilla
4. Ошибки при загрузке DVD
5. Добавление обоев
6. Переразбивка на разделы

1 GUI тю-тю

В Каждый месяц я беру запасной компьютер, устанавливаю на него все версии Linux с диска LXF и ставлю им оценки по 100-бальной шкале. Да, это бессмысленно, но позволяет избежать неприятностей. Теперь о проблеме. OpenMediaVault с DVD LXF202 установился прекрасно, но загружает только командную строку, и я не могу войти в графический интерфейс. Так как я веду записи всех тестов (страшно сказать, за несколько лет и несколько «запасных» компьютеров), я обратил внимание, что то же самое происходило с версией 1.09 той же ОС в прошлом году. Из своих записей я узнал, что это происходило и раньше и с другими версиями Linux; на самом деле, целых три раза за последние два года в выборке из 46 установок. Иногда можно запустить GUI командой startx, но это сбивает меня с толку. Как перейти в графический интерфейс из командной строки, если вы в ней застряли? Есть ли какой-то общий способ, или нужно знать какие-то особенные команды для каждой версии Linux?

Дэйв Фалксмен [Dave Fulksman]

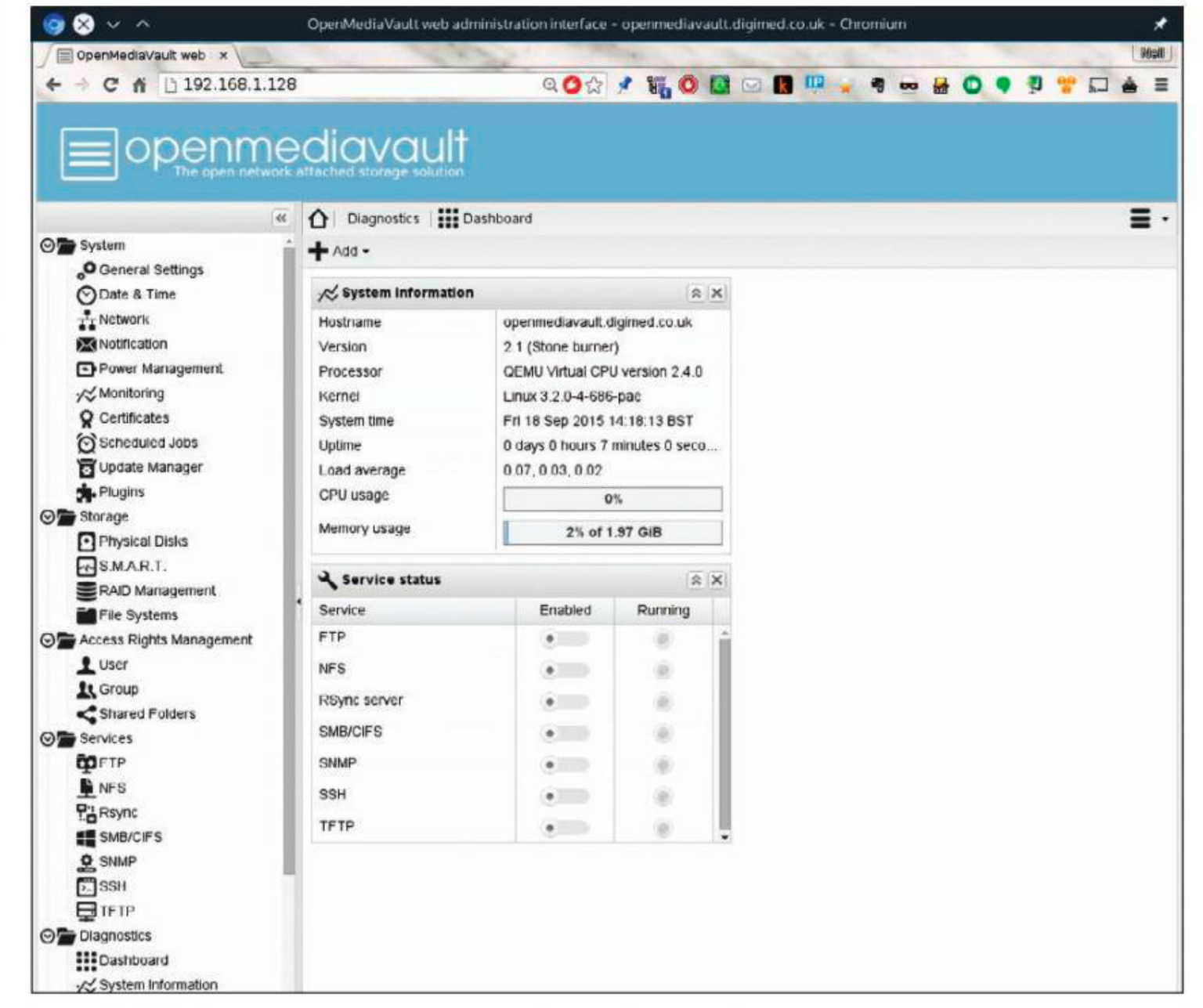
О Существует две основные причины, по которым система может загружаться в командную строку вместо графического рабочего стола. Наиболее распространенная — проблема с драйверами видеокарты в дистрибутиве; это обычно происходит с картами Nvidia, когда новые драйвера не работают со старыми картами. Иногда, как Вы обнаружили, графический интерфейс запускается командой startx, хотя, если проблема связана с проприетарными драйверами, запуск команды часто приводит к переходу на стандартный драйвер VESA. Это означает, что рабочий стол может работать медленно, но достаточно хорошо для установки корректных драйверов. У каждого дистрибутива есть собственная команда для загрузки графического стола, но в конце концов все они в той или иной форме вызывают startx. С переходом всё большего количества дистрибутивов на *Systemd* команда для переключения в графический рабочий стол становится следующей и одинаковой для всех:

```
$ systemctl isolate graphical.target
```

Другая причина, по которой дистрибутив может не загружаться в графический интерфейс — его просто об этом не просят. Что и справедливо для OpenMediaVault, представляющего собой в чистом виде сервер. Всё взаимодействие с ним выполняется через web-интерфейс, а логин и пароль для входа появляются во время загрузки с напоминанием изменить пароль администратора. Для подключения к серверу Вам потребуется его IP-адрес, поэтому сначала войдите в систему в командной строке и выполните одну из следующих команд:

```
$ hostname -I
$ ifconfig eth0
$ ip addr show eth0
```

Любая из этих команд выведет IP-адрес сервера. Наберите его в адресной строке браузера или на любом устройстве, подключенном к сети, войдите как пользователь admin с паролем openmediavault, и Вы увидите графический интерфейс OpenMediaVault.



› OpenMediaVault не загружает локальный графический интерфейс — для его просмотра пользуйтесь браузером.

2 Винтажный Wine

В На моем главном игровом агрегате установлена Xubuntu 14.04 LTS, и я совершил ошибку, позволив системе обновить мою версию *Wine* до последней (1.7.50). Несколько игр и приложений, которыми я пользуюсь, стали нерабочими. Теперь я хочу переустановиться на более раннюю версию. Когда я удаляю версию 1.7.50 и пробую установить версию 1.7 командой sudo apt-get install wine1.7, устанавливается только самая последняя версия. На www.winehq.com доступны только 1.7 и более ранняя 1.6. В идеальном случае я хотел бы переустановить версию 1.7.29.

reklan, с форумов

О Ни одного из пакетов *Wine 1.7* в репозиториях Ubuntu нет, поэтому Вы, похоже, устанавливались из PPA *Wine*. Просмотреть содержимое PPA можно, перейдя по ссылке launchpad.net/~ubuntu-wine/ +archive/ubuntu/ppa. Для версии 1.7.29 пакета сейчас нет — причина не указана, но могла возникнуть проблема, которая привела к его удалению, поэтому, возможно, будет разумнее на нем не настаивать; однако стоит

Терминалы и суперпользователи

Мы часто предлагаем в качестве решения проблемы ввести те или иные команды в терминале. Хотя обычно то же самое можно сделать с помощью графических утилит дистрибутива, такие решения будут слишком конкретными (будут зависеть от дистрибутива). Команды в терминале более гибкие и — самое главное — ими можно пользоваться во всех дистрибутивах. Команды настройки системы часто нужно выполнять от имени суперпользователя, называемого также root. Существует два основных способа это делать, в зависимости от используемого дистрибутива. Во многих дистрибутивах, особенно в Ubuntu и его производных, перед командой можно написать sudo — при этом будет запрошен пароль пользователя, и ему будут предоставлены привилегии root только на время выполнения команды. В других дистрибутивах применяется команда su, для использования которой требуется ввести пароль root и которая предоставляет полный доступ root до того момента, пока вы не наберете logout. Если в вашем дистрибутиве используется su, запустите ее один раз и выполняйте любые заданные команды без предшествующей sudo.

попробовать как более раннюю, так и более позднюю версию: 1.7.18 или 1.7.44. Удалите текущую версию (это не повлияет на настройки или на программы, уже установленные в *Wine*), затем загрузите и установите соответствующий файл **.deb** из Launchpad. Загрузив файл **.deb**, дважды щелкните по нему, и должна открыться *Gdebi* для его установки.

Установив требуемую версию, можно велеть *APT* не обновлять этот пакет, командой

```
$ sudo apt-mark hold wine
```

Процесс, возможно, придется повторить для связанных пакетов, таких как библиотеки. Но надо учитывать, что этот подход отчасти рискованный. Пакеты для дистрибутивов тестируются на совместимость с конкретным дистрибутивом, и обновление дистрибутива с некоторыми пакетами, остающимися в более старых версиях, может вызвать проблемы в будущем (хотя, если быть оптимистом, может и не вызвать).

Альтернатива — установить *Wine* из исходников. На домашней странице WineHQ перечислены только текущие стабильные и бета-версии, но tar-архивы всех версий доступны на странице sourceforge.net/projects/wine/files/Source. Загрузите и распакуйте необходимую версию и следуйте инструкциям из файла **readme**. Для сборки из исходников Вам надо установить пакет *build-essentials*. Потребуется чуть больше работы, но зато Вы получите полный контроль над установленной версией, и менеджер пакетов уже не будет пытаться обновить пакет. Если пакет перестанет работать из-за изменений в ОС (а эта проблема возможна, если пакет не обновляется), Вы сможете вернуть пакет в рабочее состояние, повторив процедуру установки.

3 Clonezilla шалит

В Я успешно пользовался *Clonezilla* многие годы в качестве системы резервного копирования для Linux, Windows и FreeDOS на компьютере с одним диском в 500 ГБ. Недавно на моем разделе Windows стало заканчиваться место, поэтому я добавил диск в 2 ТБ (sdb) к существующему 500 ГБ (sda), переформатировал и переразбил оба диска и переустановил всё. Теперь у меня есть FreeDOS (sda1), Windows 7 (sda2), Kubuntu 12.04 (sdb1), Kubuntu 14.04 (sdb2), /home (sdb5) и *Grub*, установленный на sda.

Удовлетворившись тем, что получилось, я создал резервную копию этих разделов на внешний USB-диск с помощью *Clonezilla live 2.4*, и программа даже проверила, что образы можно восстановить. Недавно мне понадобилось восстановить разделы Windows 7 и Kubuntu 14.04, но в обоих случаях *Clonezilla* ответила сообщением:

```
The image was saved from more than 1 disk. To
restore the image to different disk names is not
supported for more than 1 disk! Only the same disk
names will work.
[Образ сохранен более чем на одном
диске. Восстановление образа на диски
с другими именами для нескольких дисков
не поддерживается! Работают только одинаковые
имена дисков.]
```

***Clonezilla* предложила мне попробовать *cnvt-ocs-dev*, но я не могу найти никакой информации об этом. Я просто пытаюсь восстановить каждый образ в исходное место, из которого он создавался, а не на другой диск или раздел, как можно было бы предположить.**

Колин Морли [Colin Morley]

О Образы *Clonezilla* должны восстанавливаться на раздел с тем же именем, с которого они создавались. Считайте это мерой предосторожности, которая помешает Вам случайно уничтожить не ту файловую систему из-за мелкой опечатки. Странно, что *Clonezilla* пытается выдать такое сообщение при восстановлении в исходное местоположение; но имена дисков точно одинаковы? Например, если Вы загрузили *Clonezilla* с CD для создания копии и скопировали **/dev/sda**, но при восстановлении загрузились с флешки, то флешка будет **/dev/sda**, а диск — **/dev/sdb**. USB-диски особенно часто вызывают эту проблему, так как они могут менять порядок обнаружения дисков.

Программа, которую Вы упомянули, *cnvt-ocs-dev*, предназначена для изменения целевого устройства для образа. Выберите Shell [Оболочка] в меню *Clonezilla* и запустите ее следующим образом:

```
$ sudo cnvt-ocs-dev -d /home/partimag
IMAGENAME sda sdb
```

Здесь -d указывает каталог, содержащий Ваши образы, следующий аргумент — имя образа, и последние два — исходное и целевое устройства соответственно. Sudo необходима, потому что »

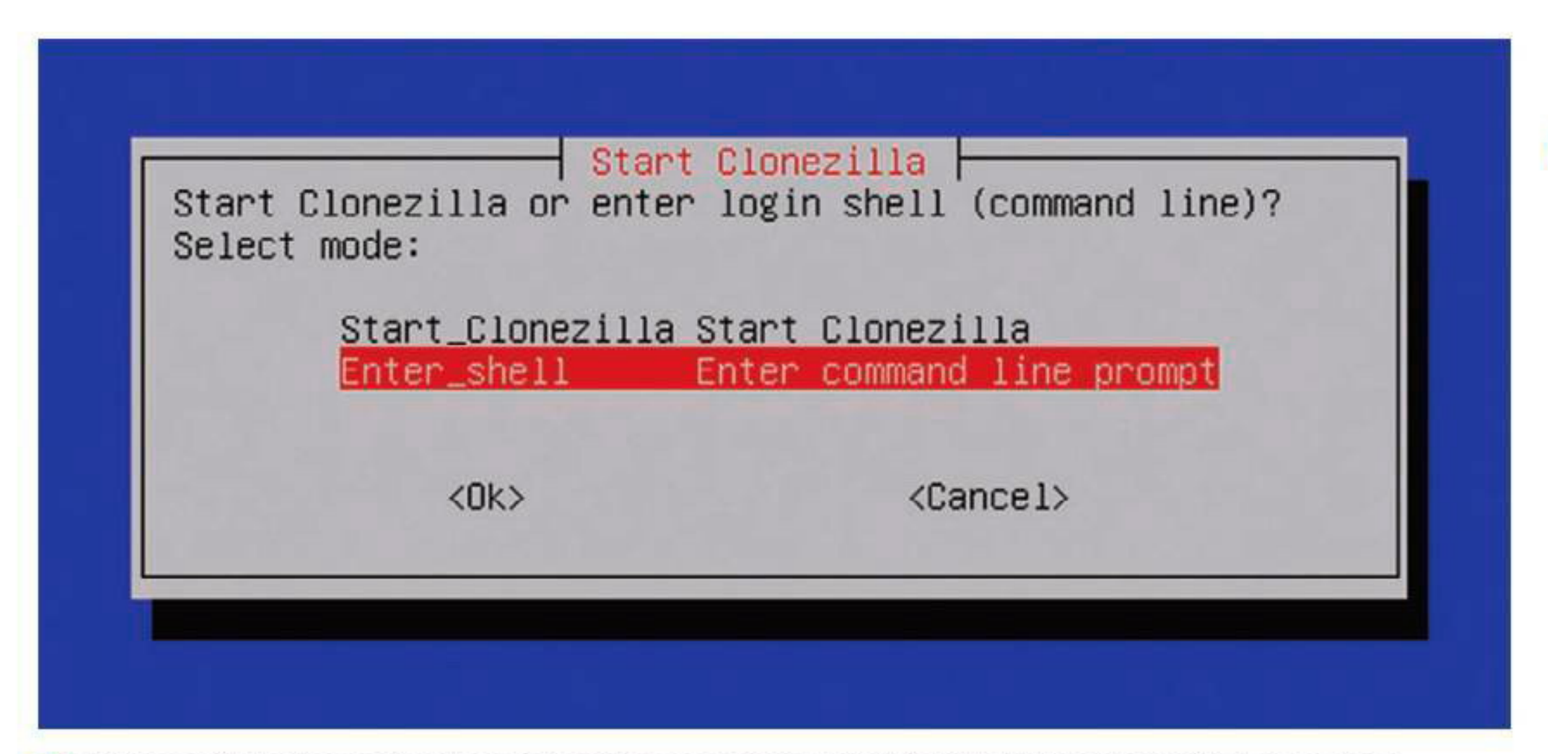
Коротко про...

Обновления ядра

Иногда менеджеру пакетов вашего дистрибутива надо установить более новую версию ядра Linux, сердца операционной системы. Пакет ядра не похож на остальные пакеты в том, что одновременно может быть установлено несколько версий. Каждый пакет помещает собственное ядро и файл initramfs в каталог **/boot**, а соответствующий набор модулей — в **/lib/modules**. Он также добавляет отдельную запись в меню загрузки, так что при установке нового ядра старое остается доступным. Это важно, если вы пользуетесь сторонними драйверами — теми, что не включены в ядро; часто это проприетарные драйвера видеокарты и беспроводного сетевого адаптера. Если вы обнаружили, что после обновления ядра беспроводная сеть не работает или вы загрузились в командную строку вместо своего любовно украшенного рабочего стола, вы сможете выбрать предыдущую версию из меню загрузки и вернуться к старому ядру, где можно исправить ошибку. Это особенно удобно, если затронут сетевой адаптер, так как для установки драйверов сетевой карты обычно необходимо сетевое подключение (что и логично с технической, если не с практической точки зрения).

Некоторые дистрибутивы скрывают меню загрузки; в этом случае необходимо несколько секунд удерживать специальную клавишу — обычно это клавиша Shift, но иногда бывает и Esc — чтобы открыть меню для продолжения загрузки.

Одно из последствий такого метода установки пакетов ядра в том, что со временем накапливается все больше и больше пакетов ядра, и в каталоге **boot** появляется множество файлов **vmlinux-***, благодаря чему вы можете высвободить немного места на диске, удалив пакеты для старых ядер, которыми не пользуетесь.



› Если Clonezilla ведет себя не так, как ожидалось, меню Enter Shell [Войти в оболочку] даст доступ кдополнительным инструментам.

программа должна запускаться с правами root; но пароль вводить не надо. После изменения местоположения образа можно вернуться в меню *Clonezilla* — командой `$ sudo clonezilla -d0 #` (это ноль, а не буква «O»). Чтобы избежать этой проблемы в будущем, можно воспользоваться ссылками в **/dev/disk/by-uuid** для использования дисков вместо файлов устройств, поскольку эти ссылки всегда будут указывать на должное устройство, даже если его имя изменилось.

4 Незагружаемые DVD

В У меня есть несколько старых компьютеров, на которых я тестирую разные дистрибутивы Linux с помощью внешнего DVD-привода Lacie. Время от времени я покупаю *Linux Format* или какой-нибудь другой журнал о Linux с DVD. Недавно я столкнулся с проблемой на более старых компьютерах. При запуске диска я получаю сообщение:

```
Welcome to GRUB
[Добро пожаловать в Grub]
error:unknown filesystem.
[ошибка: неизвестная файловая система]
Entering rescue mode ...
[Переход в спасательный режим]
grub rescue>
```

Интересно, что, по мнению 'grub rescue', я должен сделать?

fossiili

О Это странно. Файловая система — стандартная CD/DVD ISO9600, ничего экзотического, и с ее загрузкой не должно возникнуть никаких проблем на любом компьютере, собранном в этом веке. Там, где выводится сообщение 'Welcome to Grub', привод просто считывает меню *Grub* с DVD. Процесс загрузки на наших DVD не меняется с номера 178, около двух лет (до этого мы использовали *isolinux* вместо *Grub*), так что проблема вряд ли связана с ним, особенно если учесть, что проблема повторяется и с другими DVD с *isolinux*.

Мы тестируем DVD на реальном и виртуальном «железе» и не сталкивались с этой проблемой. Предполагаю, что проблема может быть связана с неисправностью Вашего DVD-привода, особенно потому, что ошибка появляется в разных системах, и привод — это то общее, что у них есть. Также возможна проблема с BIOS, но только если Вы недавно меняли настройки BIOS. Оптические приводы стареют, энергия лазеров может выдыхаться, линзы — загрязняться, особенно если привод используется в пыльной или прокуренной среде; всё это приводит к потере мощности. Если у Вас есть другой внешний привод или Вы можете его одолжить, попробуйте исключить привод как причину проблемы.

Загружаете ли Вы компьютер «холодным»? Когда я пробую сделать это с внешним диском на своем ноутбуке, он не загружает DVD, потому что привод не успевает раскрутиться и не дает распознать BIOS, и BIOS ищет другой источник загрузки. Перезагрузка с помощью Ctrl+Alt+Delete, при которой привод не останавливается, позволяет успешно загрузить DVD.

Аварийная оболочка *Grub* — это окружение командной строки, в котором можно запускать команды *Grub*, но она по-настоящему полезна только если Вы достаточно хорошо знакомы с *Grub*. В этой среде нельзя запускать стандартные команды оболочки.

5 Невидимые обои

В Я хочу положить свои обои в каталог /usr/share/backgrounds и сделать их доступными в обоях по умолчанию. К сожалению, несмотря на все мои попытки, включая изменения прав доступа и владельца с `sudo` или `su`, эти обои все равно не доступны в каталоге. На самом деле, один раз я даже не смог получить доступа к обоям по умолчанию — система радостно сообщила мне, что я (даже как root) не являюсь их владельцем. Я сравнительно новый пользователь и подозреваю, что все значительно проще, чем кажется с виду.

Есть ли какие-либо другие изменения, которые нужно сделать, чтобы мои обои стали доступны из каталога с обоями? На форумах мне попадались сообщения об этой же проблеме, но никто, судя по всему, не нашел решения. Это не должно быть проблемой, так как отнюдь не выглядит сложной задачей для Linux. Сейчас я пользуюсь Ubuntu 14.04 LTS. Я только что решил полностью отказаться от Windows, вот и помогите мне убедиться в правильности своего решения.

Cozzie

О Каталоги в **/usr** находятся под управлением менеджера пакетов, который изначально установил туда обои. Добавлять файлы в **/usr** случайным образом — не слишком хорошая идея. Не то что Вам нельзя этого делать, но вторгаться в области, контролируемые менеджером пакетов, не рекомендуется. Если Вы хотите добавить файлы, которые должны быть доступны системе, и не затронуть сферу интересов менеджера пакетов, рекомендуется поместить их в **/usr/local/share**. Базовое правило состоит в том, что дистрибутив устанавливает файлы в **/usr**, внешние файлы устанавливаются в **/usr/local**, а файлы отдельных пользователей — в **~**.

Так как это открытое ПО, Вы можете делать почти всё, что хотите, но стандарты и соглашения обычно существуют не зря, и если Вы их нарушаете, остается собирать осколки.

Однако это не причина, по которой добавленные Вами файлы не видны. Добавить файлы в список по умолчанию Unity или Gnome — нетривиальная задача. Чтобы они появились в списке обоев, их нужно также добавить в XML-файл в **/usr/share/gnome-background-properties**, создав отдельную запись для каждого файла.

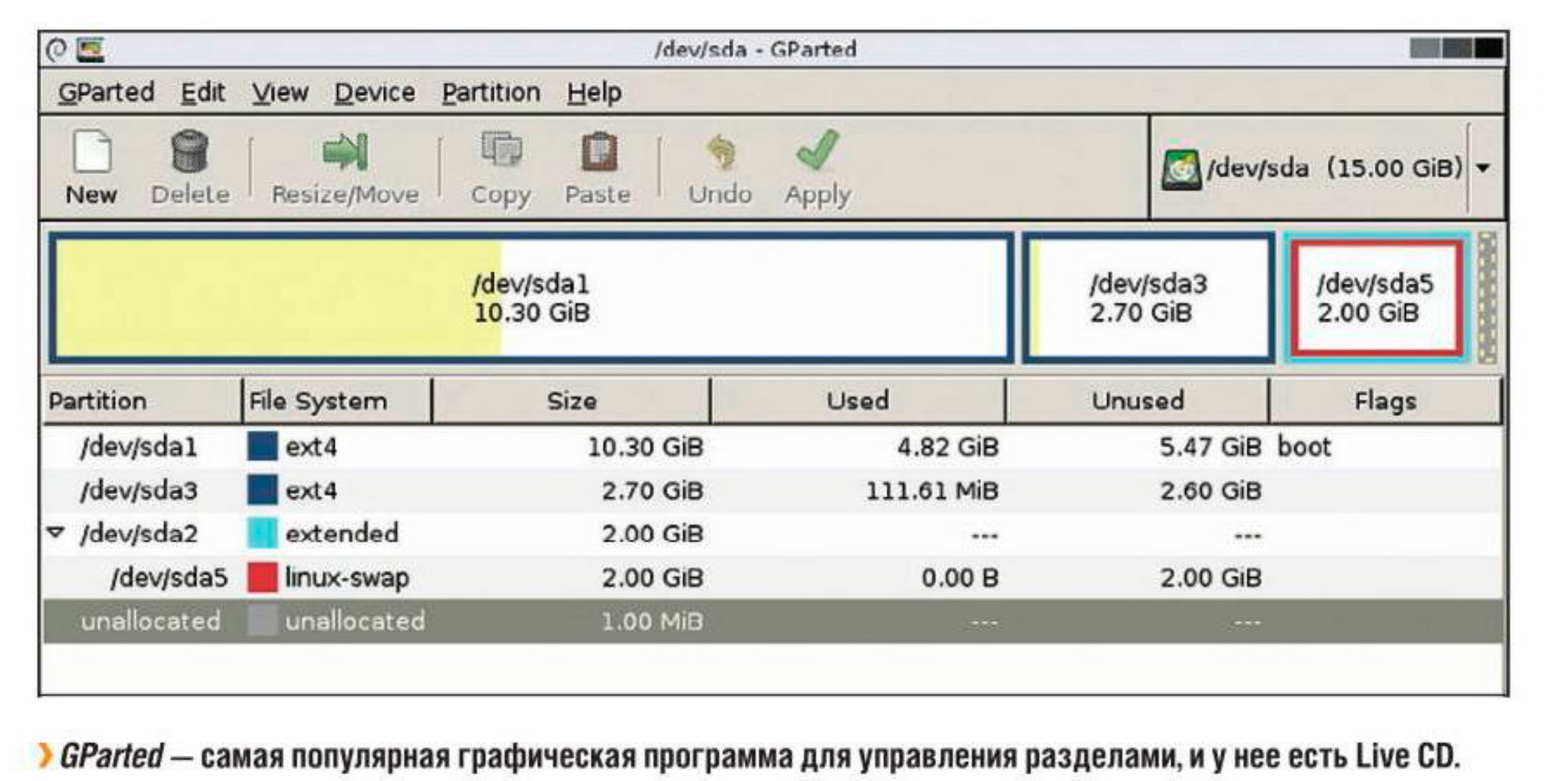
Но если поместить все файлы в отдельный каталог, скажем, **/usr/local/share/backgrounds** для глобального использования или **~/.local/share/backgrounds** для использования только Вами, их можно будет выбрать в окне Appearance [Внешний вид], и они появятся в окне Pictures [Изображения]. При этом они не обязаны находиться в каталоге **Pictures** — их можно положить куда угодно. Учтите, что указанный XML-файл специфичен для Gnome. В большинстве других рабочих столов можно поместить свои файлы в стандартный каталог (или создать символические ссылки на них) в домашнем каталоге или в каталог **/usr/local**.

6 Увеличить или сдвинуть?

В У меня установлены Linux Mint 16 и Windows XP, но я хочу увеличить размер раздела Linux. Я уменьшил размер раздела Windows, и теперь у меня есть незанятое место. Как увеличить размер раздела Linux, не потеряв данные?

trevorw

О У Вас теперь между концом раздела Windows и началом корневого раздела Linux есть пустое пространство. Это оставляет Вам два варианта действий. Первый — переместить начало раздела Linux; при этом увеличится его размер. Это нетривиальная задача, но с ней справится *GParted*. Вам надо загрузиться с Live CD с *GParted* — его можно скачать с http://gparted.org/livecd.php. Вы просто перетаскиваете начало раздела Linux влево и нажимаете Apply [Применить]. Но для *GParted* это сложная задача, и она займет некоторое время. Если в процессе установки система «падала» или отключалось электричество,



› *GParted* — **самая популярная графическая программа для управления разделами, и у нее есть Live CD.**

Ваша корневая файловая система может быть повреждена, поэтому для начала сделайте полную резервную копию.

Другой вариант — создать на пустом пространстве отдельный раздел и использовать его для **/home**. Это позволяет избежать необходимости изменять размер и дает Вам преимущество в виде отдельного раздела **/home**, который удобен при смене или переустановке дистрибутива. Опять же, с помощью Live CD и *GParted* создайте новый раздел в незанятом пространстве и установите тип файловой системы ext4. Сделав это, смонтируйте старый и новый разделы, что можно сделать из терминала. Предположим, исходный раздел Linux называется **/dev/sda2**, а новый — **/dev/sda4**. Откройте терминал и выполните такие команды:

```
$ sudo mkdir -p /mnt/{old,new}
$ sudo mount /dev/sda2 /mnt/old
$ sudo mount /dev/sda4 /mnt/new
```

Теперь в указанных каталогах появилось содержимое этих двух разделов, и можно переместить файлы в новый каталог *home*, выполнив команду

```
$ sudo mv /mnt/old/home/* /mnt/new/
```

Обратите внимание, что Вы перемещаете файлы из каталога **home** на старом разделе напрямую в новый, так как на следующем шаге он будет смонтирован в раздел **/home**. По окончании перемещения откройте в текстовом редакторе файл **/etc/fstab**, в котором описываются разделы для монтирования:

```
$ sudo leafpad /mnt/old/etc/fstab
```

Под строкой, которая монтирует раздел /, добавьте следующую строку:

```
/dev/sda4 /home ext4 defaults,noatime 0 0
```

не забыв заменить /dev/sda4 именем нового раздела. Затем перезагрузитесь, и в каталоге **home** и корневой файловой системе должно появиться много свободного места. LXF

Помогите нам помочь вам

Ежемесячно мы получаем несколько писем, на которые не в состоянии ответить, поскольку проблема описана в них недостаточно полно. Чтобы дать вам наилучший ответ, нам необходимо знать как можно больше.

Если у вас появляется сообщение об ошибке, приведите его точный текст и опишите конкретные условия, когда оно появляется. При возникновении проблемы с устройствами перечислите нам все устройства, которые у вас установлены.

Если Linux уже запущен, можете применить для этого отличную программу *Hardinfo* (http://sourceforge.net/hardinfo.berlios) — она сохранит подробную информацию об устройствах и о состоянии системы в HTML-файле, который вы сможете приложить к своему письму, отправляемому нам.

Не уступающий в удобстве альтернативный вариант — *lshw* (http://ezix.org/project/wiki/HardwareLiSter). Одна из указанных программ непременно должна быть включена в ваш дистрибутив (а иногда и обе).

Если вы не хотите или не можете их установить, выполните приведенные ниже команды в терминале от имени root и приложите файл **system.txt** к письму. Это здорово поможет диагностике.

```
uname -a >system.txt
lspci >>system.txt
lspci -vv >>system.txt
```

Коротко про...

strace

› Что такое *strace*?

Утилита, которая записывает все системные вызовы программы, запущенной с помощью этой утилиты.

› А в чем тут смысл?

Пока программа работает правильно, особо ни в чем. Когда программа начинает вести себя некорректно, *strace* покажет вам, что она пытается сделать, какие файлы пытается открыть и какие библиотечные вызовы пытается выполнить.

› Поможет ли это мне?

Способно помочь: шансы увидеть, что программа пытается сделать, часто помогают выявить причину проблемы.

› Как ее запустить?

```
$ strace имя_программы
```

› Она выводит какую-то абракадабру. Что с этим делать?

Вы можете посмотреть, какие системные вызовы завершаются неудачно, и это должно помочь понять, что делать дальше. Вывод программы также можно вставить в отчет об ошибке, просьбу о помощи или web-запрос, чтобы другие пользователи помогли вам разобраться в нем.

› Вывода гадски много. Можно ли сохранить его?

Чтобы сохранить вывод в файл, воспользуйтесь параметром -o имя_файла. Если вы используете перенаправление оболочки, она также сможет перехватить вывод программы. Для перехвата того и другого отдельно можно использовать -o и перенаправление.

› Он все равно слишком длинный. Можно ли как-то ограничить объем отчета?

Параметр -e позволяет указать системные вызовы, которые надо отслеживать. Если программа жалуется, что не может найти конфигурационный файл, или параметры, помещенные в такой файл, игнорируются, можно посмотреть вызовы open, чтобы увидеть, какие файлы она пытается открыть.

```
$ strace -e open -o report.txt
имя_программы
```

› А если я хочу отслеживать несколько типов вызовов?

Это можно сделать с помощью несколько иного синтаксиса для -e, используя список вызовов, разделенных запятыми:

```
$ strace -e trace=open,read -o report.
txt имя_программы
```

› При запуске рабочего стола программа уже запущена. Можно ли ее отследить?

Да, можно, и именно здесь *strace* оказывается наиболее полезна. Сначала найдите идентификатор процесса с помощью ps или pgrep, затем передайте его *strace* с параметром -p:

```
$ ps ax | grep kwin 12345
$ strace -p 12345 -o kwin.txt
```

LXFHotPicks

Midori » Trinity » Add-on converter » ZSH » Terminator » sView » Briss » Mega Mario » Oilwar » Drawpile » Solaar

Александр Толстой
из почтения к «Охотникам за привидениями» нацепил свой ускоритель частиц и бдительно пасет самых резвых призраков СПО в мюонной ловушке.

Web-браузер

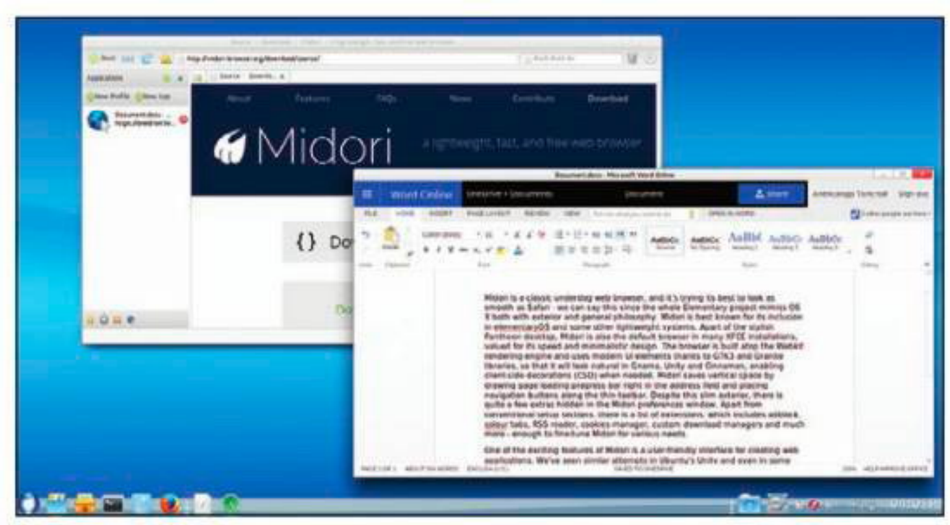
Midori

Версия: 0.5.11 **Сайт:** http://midori-browser.org

Midori — классический неудачник среди web-браузеров, который изо всех сил старается выглядеть таким же стильным и лощеным, как *Safari*. Смеем так утверждать, поскольку весь проект Elementary, который занимается разработкой elementary OS, имитирует внешний вид и общую философию OS X.

Midori больше всего известен благодаря включению в производную от Ubuntu elementary OS и несколько других легковесных дистрибутивов. *Midori* также является браузером по умолчанию во многих установках с *Xfce*, поскольку там ценят его скорость и минимализм. Браузер создан на движке рендеринга Webkit и использует современные элементы интерфейса пользователя из библиотек *GTK 3* и *Granite*, а значит, естественным образом впишется в Gnome, Unity и Cinnamon, и при необходимости включит оформление клиентской стороны [CSD, client-side decorations].

Midori сэкономил место по вертикали, сдвинув строку состояния загрузки страницы прямо в адресную строку и разместив кнопки навигации на узкой панели инструментов. При всей внешней экономности, здесь немало дополнительных функций, скрытых в окне Midori Preferences. Помимо обычных разделов настройки, имеется список расширений, включающий AdBlock, цветные вкладки, читалку RSS, менеджер куки, индивидуальные менеджеры скачивания и т.д., чего вполне хватит для очень тонкой настройки *Midori*.

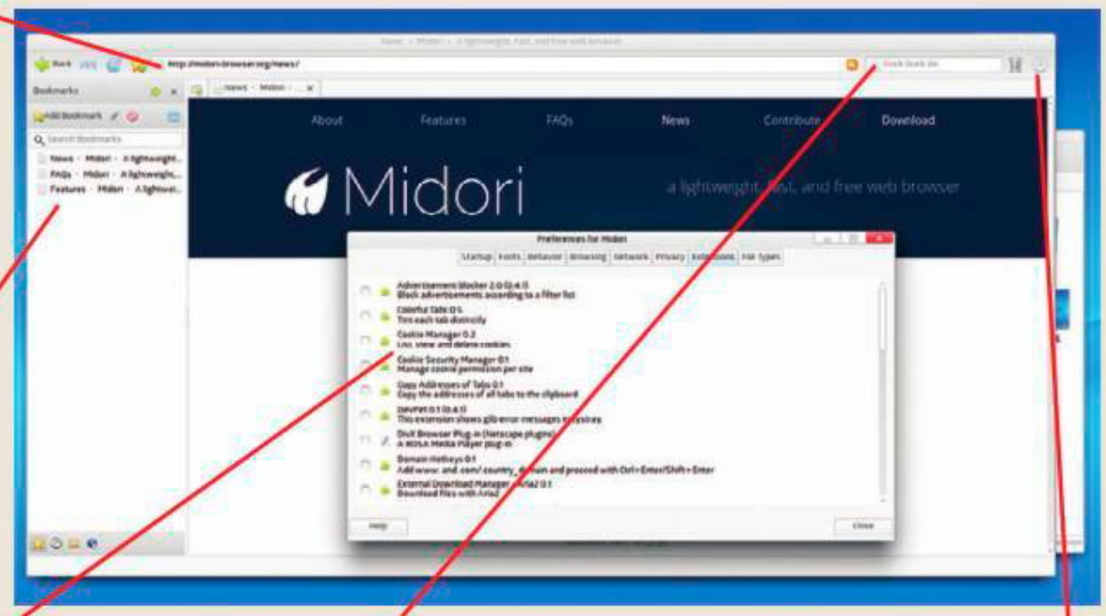


› ***Midori*** **умеет инкапсулировать сайты в приложения рабочего стола.**

Создание приложений с сайтов может стать сущим удовольствием.

Одна из интересных функций *Midori* — дружелюбный к пользователю интерфейс для создания web-приложений. Мы видели подобные попытки в Ubuntu Unity и даже в некоторых сторонних плазмоидах для KDE, но реализация *Midori* элегантнее. Например, при выборе Applications в основном меню (помеченном кнопкой с шестеренкой) с левой стороны окна *Midori* появится боковая панель, где можно создавать приложения одним щелчком мыши. Всего-то и надо открыть нужный сайт, а затем нажать кнопку New App. *Midori* создаст XDG-совместимую запись, то есть при любом вашем рабочем столе он автоматически обнаружит эту запись в меню приложений или программе запуска. Такие приложения запускают *Midori* в специальном режиме без видимых кнопок управления. Создание своих приложений с сайтов может стать сущим удовольствием, как, например, имитация несуществующей (пока) Linux-версии *Microsoft Office* (на рис. вверху). Многие дистрибутивы предлагают две версии *Midori*: 0.4 с классическим интерфейсом *GTK 2* и самую последнюю серию 0.5 с библиотекой *Granite* (это расширение *GTK 3* с индивидуальными виджетами и управлением). Выглядят эти вариации *Midori* неодинаково, но web-страницы обрабатывают без особых различий. Пользователи Ubuntu могут добавить PPA *Midori* (**ppa:midori/ppa**) для получения самого последнего релиза.

Исследуем интерфейс Midori



Адресная строка
Здесь отображается также и состояние загрузки страницы в браузере, как в *Safari* от Apple.

Боковая панель
Эту панель можно использовать для отображения закладок, журнала, передачи файлов или web-приложений.

Расширения
Вы сможете добавить несколько личных штрихов, познакомившись со встроенными расширениями *Midori*.

Утиный поиск
Midori использует поисковый движок Duck Duck Go [duck — утка, — *прим. пер.*], который уважает вашу конфиденциальность и избегает персонализации.

Секретная кнопка
Кнопка с шестеренкой в верхнем правом углу скрывает массу добавочных функций и расширений.

Среда рабочего стола

Trinity

Версия: R14.0.1 **Сайт:** www.trinitydesktop.org

В Интернете полно архаичных (или, если вам так больше нравится, ностальгических) сайтов, которые демонстрируют древнее и популярное утверждение, что «новее — не всегда лучше». Это особенно правдиво применительно к *Trinity*, ответвлению рабочего стола KDE 3, разработка которого официально прекратилась в 2008 г. Команда *Trinity* полагает, что он сохраняет функции и форму традиционного рабочего стола в отличие от современного контент-ориентированного подхода (и прочих HID).

Trinity тщательно поддерживает и улучшает ответвление KDE 3.5.x, внося в него современные усовершенствования, включая *PulseAudio* и интеграцию с Bluedevil, и многое другое, например, XDG-совместимые меню и улучшенные уведомления D-BUS. Год назад разработчики проекта наконец-то отказались от нумерации версии 3.5.xx в пользу серии R14.

Внешний вид *Trinity* уводит нас во времена стесанных кнопок в стиле Keramik, медиа-плейера *Kaffeine* и исключительно богатой функциями панели Kicker. Более того, *Trinity* возвращает к жизни классический музыкальный плейер *Amarok 1.x*, web-браузер *Konqueror* на Flash и дружелюбный к *Network Manager* системный лоток. В именах всех компонентов строку KDE заменили на TDE — это выглядит куда лучшим подходом, чем предпринятый в Mate, где знакомым приложениям дали совершенно новые названия. В результате, *Trinity* вполне комфортно усядется рядом с рабочим столом KDE 4 или Plasma 5.

Кроме того, компоненты TDE упакованы с использованием префикса /opt — так, на всякий случай — для многих дистрибутивов Linux. *Trinity* доступен для Ubuntu и его производных на https://quickbuild.pearsoncomputing.net, а также для Debian, Fedora и RHEL/CentOS.

› **Единственное визуальное различие между *Trinity* и исходным рабочим столом KDE 3 — новый логотип и новая подпись меню.**

Комфортно усядется рядом с рабочим столом KDE 4 или Plasma 5.

Прагматичный довод для установки *Trinity*, помимо очевидной ностальгии, заключается в его впечатляющей отзывчивости, что очень важно для старых компьютеров, вместе с отлично поддающейся индивидуальной настройке структуре, и он используется многими из тех, кто в былые дни работал с KDE.

Конвертор расширений

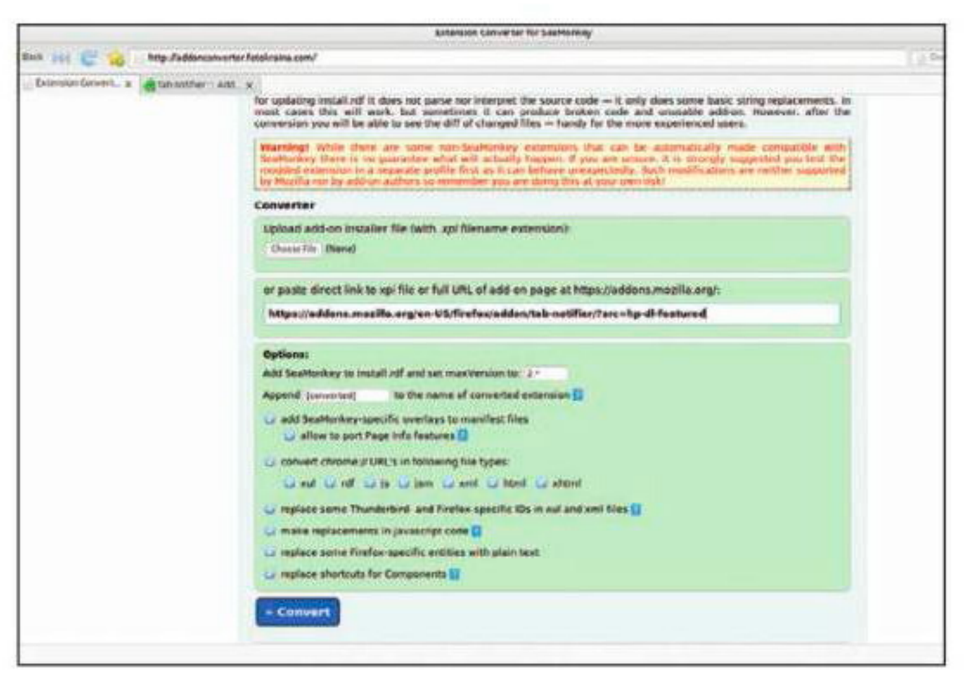
Add-on converter

Версия: Git **Сайт:** http://bit.ly/AddonConverter

Firefox много лет является популярнейшим в мире web-браузером, и один из побочных эффектов этого мирового господства — потрясающее количество сторонних дополнений, расширяющих его функции. В терминологии Mozilla дополнение [add-on] — это общее название для расширений, тем и подборок тех и других. Официальный Mozilla Add-On каталог для *Firefox* содержит более 14 000 дополнений, самые популярные из которых служат для выполнения главных запросов среднестатистического пользователя, например, скачивания видео YouTube, блокировки рекламы, добавления визуальных эффектов, и т. д.

Однако *Firefox* — не единственный браузер, официально разрабатываемый и поддерживаемый Mozilla Foundation; есть еще *Seamonkey*, темная лошадка для многих и отличный браузер во всех отношениях. *Seamonkey* — прямое продолжение классического *Netscape Navigator*, который воскрес в виде *Mozilla Suite* и позднее был переименован в *Seamonkey*. Браузер повторяет внешний вид, образ действий и структуру UI классического браузера *Navigator*, но с самым свежим движком Gecko и прочими современными наработками *Firefox* в качестве начинки. Единственная его проблема заключалась в том, что каталог дополнений *Seamonkey* оставлял желать лучшего, насчитывая всего 800 пунктов.

К счастью, у нас есть этот специальный конвертор дополнений, который исправляет и переупаковывает дополнения *Firefox* для работы с *Seamonkey*. Конвертор является web-приложением, доступным онлайн, но может быть установлен и локально (скачайте его с https://github.com/lemon-juice/addonconverter). Вам не придется находить расширение *Firefox* в виде файла — всего лишь введите URL его онлайн-страницы. Настройки в общем и целом приемлемы по умолчанию, так что просто жмите на большую кнопку Convert.



› **Конвертор дополнений выполняет ряд обычных задач, которые вы можете по желанию включить или отключить в *Seamonkey*.**

Исправляет и переупаковывает дополнения Firefox для Seamonkey.

Конвертор создаст заново упакованное дополнение XPI, которое можно загнать в *Seamonkey* (нажав Ctrl+o и выбрав файл XPI). Все конвертированные дополнения хорошо устанавливаются, но если внутри есть код чисто для *Firefox*, они могут не всегда работать нормально. Мы сочли конвертор весьма полезным и всего несколькими щелчками мыши создали потрясающую версию *Seamonkey*.

Оболочка Unix

ZSH

Версия: 5.1 **Сайт:** www.zsh.org

ZSH — интерпретатор команд (оболочка) Unix, который применяется и как интерактивная оболочка входа в систему, и как оболочка обработчика командных скриптов. Из стандартных оболочек Linux *ZSH* больше всего напоминает *KSH*, но включает множество расширений функциональности оболочки: например, в ZSH есть редактирование командной строки; программируемое завершение команд; встроенная проверка орфографии; функции оболочки (с автозагрузкой); механизм журнала, и т.д.

Чтобы использовать *ZSH* после установки, откройте свой любимый эмулятор терминала и введите $ zsh. Появится мастер первичной настройки, который подробно опишет основные функции *ZSH*.

Например, *ZSH* имеет легко настраиваемую историю команд; удобный скрипт *compinstall* с интерактивным редактором автодополнения; преднастроенную работу некоторых клавиш (для пользователей *vi* или *Emacs*, или, если захотите, вообще индивидуальную); случаи подачи звукового сигнала, и т.д.

ZSH предлагает беспредельные возможности для фанатов командной строки, которые обожают инкапсулировать длинные команды в сжатые алиасы. В качестве еще одного примера, вот синтаксис для создания алиасов:

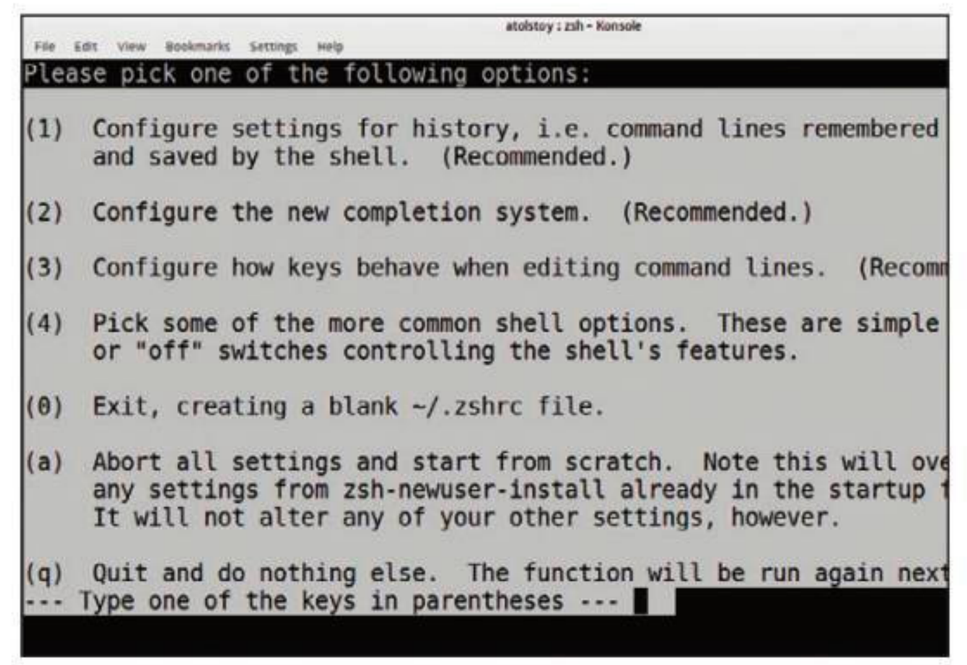
```
$ alias lss='ls -g -s'
```

ZSH также любят за разумное автоисправление команд, и оно вам особенно пригодится, если вы страдаете от частых опечаток. Однако если вы сочтете эту функцию слишком надоедливой для некоторых команд, можете отключить для них механизм исправления с помощью модификатора nocorrect. Например: $ nocorrect git status не даст *ZSH* исправлять status на stats, что, очевидно, совершенно излишне. И снова, используя предыдущий пример, вы можете слегка облегчить себе жизнь, используя следующее:

```
$ alias git status='nocorrect git status'
```

Все опции для *ZSH* можно также изменить вполне традиционным способом, отредактировав файл **~/.zshrc** в текстовом редакторе. Если вам нравится *ZSH* — мы полагаем, когда-нибудь вы все же нырнете в этот океан функций! — вы сможете настроить оболочку по умолчанию для входа в свою систему Linux. Просто запустите $ chsh -s /bin /zsh — и всё.

ZSH также любят за его разумное автоисправление команд.



При самом первом запуске *ZSH* вы можете точно настроить ее с помощью мастера.

Мультиплексор оболочки

Terminator

Версия: 0.97-2 **Сайт:** http://bit.ly/GnomeTerminator

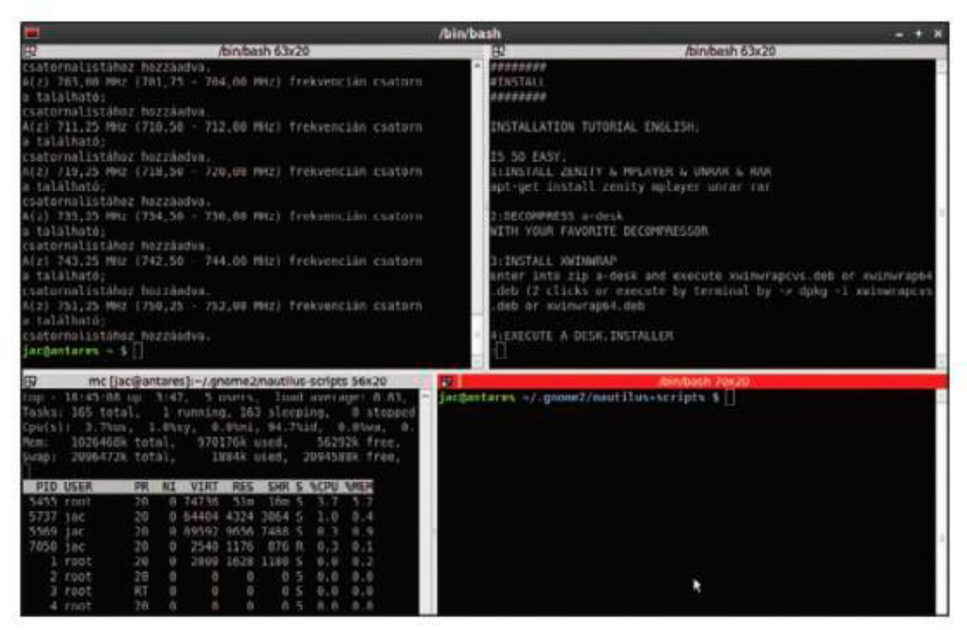
Мы уже делились всякими интересными подсказками, хитростями, обходными путями и спецпрограммами, которые позволяют использовать современные компьютеры без графического пользовательского интерфейса [см. *Побег из GUI*, стр. 46 **LXF197**]. Но если вы не готовы к столь радикальным изменениям, почему бы не оставить GUI, заодно применяя наилучшие способы создания минималистского, быстро реагирующего рабочего пространства? *Terminator*, например, является компактным приложением для эффективного заполнения большой экранной области терминалами. Его можно использовать для открытия нескольких терминалов в одном окне и использовать привязки к клавишам для переключения между ними.

Приложения вроде *Tmux* организуют рабочую область исключительно в текстовом режиме, а *Terminator* делает то же самое, но работает, как обычное приложение *GTK*, заключая терминалы в одно окно. Разумным обоснованием использования *Terminator* является возможность отслеживать несколько интерфейсов командной строки за раз, работая при этом с браузером или иными приложениями рабочего стола, совершенно как обычно.

Terminator работает как надстройка *GTK* над компонентами Gnome GUI и Python-VTE (VTE — это виджет терминала, используемый Gnome Terminal). Свежие версии *Terminator* теперь поддерживают *GTK 3* вместе с наследием *GTK 2*. При запуске *Terminator* выглядит точь-в-точь как еще один эмулятор *Терминала*, только с дополнительной красной верхней панелью и кнопкой меню слева. Управление терминалами привязано к определенным клавишам — их необходимо выучить (или справляться с $ man terminator). Так, например, нажатие клавиш Ctrl+Shift+o распределит ваши терминалы по горизонтали, Ctrl+Shift+e — по вертикали, Ctrl+Shift+f осуществит поиск внутри текущего терминала, а Ctrl+Shift+<стрелка> размещает текущий терминал в соответствующем направлении.

Можно перемещаться между терминалами, щелкая по требуемому мышью. Текущий терминал всегда обозначается красной строкой вверху. Кнопки меню скрывают некоторые расширенные функции, например, группировку терминалов и трансляцию, и можно вводить команды в несколько терминалов одновременно — очень круто!

Можно вводить команды в несколько терминалов одновременно.



Этот *Terminator* не отправлялся в прошлое, но делает применение командной строки удобнее уже сегодня.

Просмотрщик стерео

sView

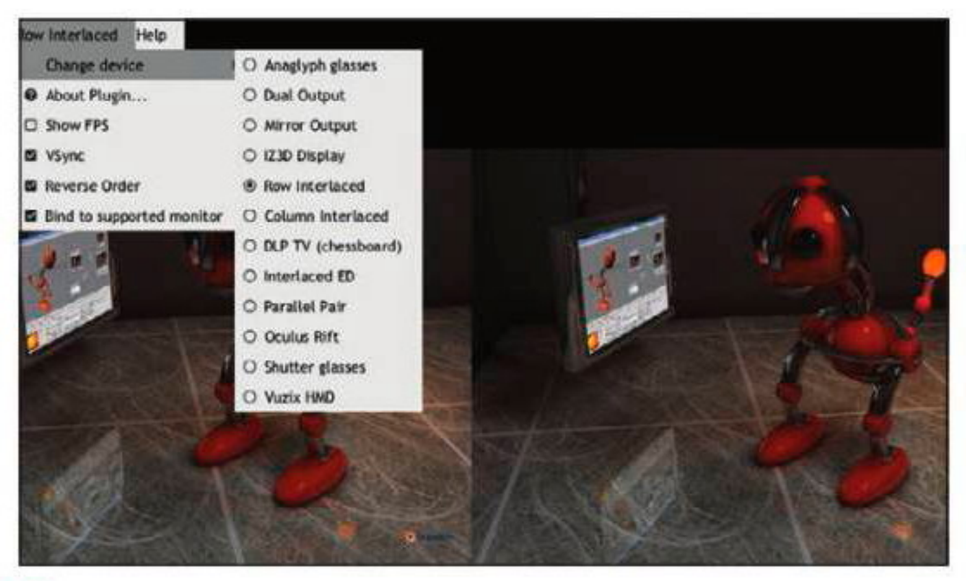
Версия: 15.08 **Сайт:** www.sview.ru

sView — приложение с рядом непривычных функций. Как намекает 's' в его названии, это не только плейер для просмотра фильмов, но также и программа просмотра изображений. Приложение создано для предпросмотра и воспроизведения стереоскопических 3D-данных в разных форматах. Однако самый лакомый кусочек в *sView* — его способность конвертировать обычные видео в стереокартинку, которую можно рассмотреть через 3D-очки. Все это декодируется с помощью среды *FFmpeg* и отображается как вывод OpenGL.

И панель задач, и выпадающие меню представлены в виде полупрозрачных экранных сообщений, как на потребительской электронике. Приложение идет с демо-изображением JPS (JPEG в стерео), с которым можно поиграть — настроить нужный режим отображения и отладить прочие параметры. В *sView* также множество опций настройки отображения, включая богатый выбор поддерживаемых устройств, таких как: анаглифные очки; дисплей IZ3D; стереоочки с активным затвором; Oculus Rift; и прочие шлемы, и очки, и дисплеи.

Вы также можете изменять тип поверхности (плоскость, шар и куб); настраивать фильтр сглаживания; запускать опцию Vsync; выбирать тип и порядок вывода стерео, а также изменять стереоформат посредством индивидуальной композиции изображений и перекрывания.

Эта программа одинаково хорошо работает и для статических изображений (например, фотографий), и для подвижных. В фильмах файл фильма содержит специальные метаданные 3D; *sView* даже умеет автоматически определять стереоформат, но вы всегда можете настроить формат вручную. Приложение даже поддерживает воспроизведение IPTV: вы можете задать ему поток M3U и смотреть его в 3D.



› Помимо необходимости среды *FFmpeg*, *sView* зависит от пары 3D-очков, чтобы вы смогли насладиться стереоформатом.

Для предпросмотра и воспроизведения 3D-стереоданных.

sView — кросс-платформенный продукт, но не слишком широко распространен в мире Linux. Он предлагает только PPA Ubuntu (**ppa:sview/stable**) и tar-архив исходника. Однако процедура компиляции достаточно проста, хотя и не вполне обычна: вы просто вводите `$ make/code` в директории *sView* **sources**, и если у вас есть заголовки *FFMpeg* и некоторые другие основные программы, все должно пройти хорошо.

Утилита PDF

Briss

Версия: 0.9 **Сайт:** http://bit.ly/BrissPDFCropper

Порой натыкаешься на довольно уродливые и практически заброшенные своими хозяевами приложения, которые, однако, иногда умудряются оказаться важным недостающим звеном. Публикация документов — именно та область, где пользователи Linux трудолюбиво копят полезные, и часто забытые, приложения и утилиты, пригодные в определенных случаях. Один из примеров — работа с промышленным стандартным форматом PDF, и многие пользователи привечают *Briss*, небольшое приложение на Java, которое использует библиотеки с открытым кодом *Itext* и *Jpedal* и выполняет простую, но очень важную работу по извлечению страниц из PDF-файлов.

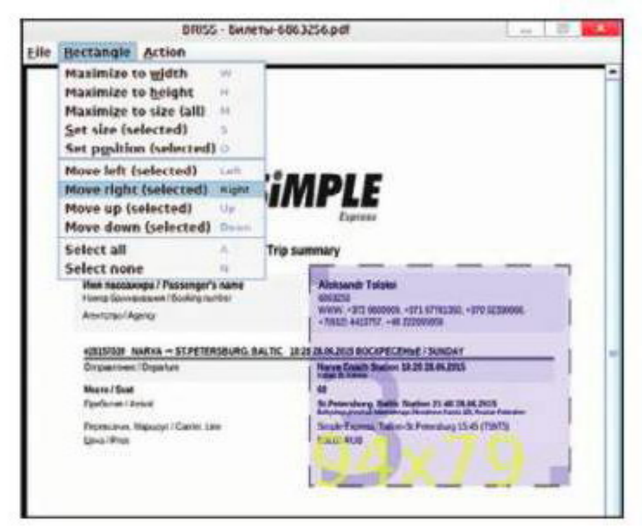
Проблема доступа к приложению решена очень элегантно: *Briss* распространяется в виде одной архивированной директории с кросс-платформенными файлами JAR (для Linux и OS X) и дополнительным EXE-файлом для ленивцев с Windows.

Запуск приложения в Linux предполагает наличие у вас среды выполнения [runtime] Java, например, OpenJDK (есть в любом дистрибутиве Linux). Чтобы запустить *Briss*, откройте терминал в его директории и запустите `java -jar briss-0.9.jar`, или можете сразу открыть требуемый PDF-файл командой

```
$ java -jar briss-0.9.jar yourfile.pdf .
```

С *Briss* очень просто работать; кстати, его название — сокращение от The Bright Snippet Sire [Блестящий Повелитель Фрагментов]. Запускается он открытием PDF-файла и указанием страниц, которые вы хотите убрать из склейки. Когда файл загрузится и будет показан в главном окне, вы сможете рисовать на нем прямоугольники, следуя намеченной структуре; и количество прямоугольников не ограничено, так что каждый фрагмент можно превратить в отдельную страницу целевого файла. Можно перетаскивать прямоугольники мышью, выделять их, выравнивать их и устанавливать точный размер и расположение (в миллиметрах). *Briss* предлагает эти опции щелчком правой кнопки по прямоугольнику или через меню Rectangle. Когда вы решите, что выделили всё, что нужно, можете посмотреть на плоды своего труда, выбрав Action > Preview и проверив тестовый PDF-файл. Чтобы подтвердить результаты, выберите Action > Crop PDF и укажите директорию для целевого файла.

Выполняет работу по извлечению страниц из PDF-файлов.



› Держите это небольшое приложение под рукой на случай, если вам понадобится нарезать PDF.

HotGames Развлекательные приложения

Платформенная аркада

Mega Mario

Версия: 1.7 **Сайт:** http://bit.ly/MegaMario

Mega Mario — очередная инди-игра, использующая очарование ретро-классики. Это классическая реинкарнация оригинальной *Super Mario Bros.*, где Марио [Mario] должен спасти своего брата Луиджи [Luigi], которого захватил злобный Bowser. Игра старательно копирует многие элементы *Super Mario Bros.*, включая фон Mushroom Kingdom [Грибное царство], рассыпанные монеты, кирпичи с вопросительными знаками и т.д. Марио может подобрать оружие Fire Flower [Цветок огня], вырасти вдвое (для каждой трансформации требуется суперцветок), и, конечно, поражать врагов, наскакивая на них.

У *Mega Mario* немного другой набор графики, что придает ей более свежий вид и лучшее разрешение на экране. Еще одна отличительная, и отчасти шокирующая опция, с учетом ее происхождения от Nintendo, это режим Blood, в котором все персонажи (включая Марио) размазывают кровь по всему уровню, если их ранят. Однако графика выглядит так, словно их нарисовали в простом редакторе типа *Paint*, поэтому кровь напоминает темно-красные фонтанчики.

Навигация в игре осуществляется с клавиатуры или геймпада, и аркадное действо длится на протяжении 24 предварительно определенных уровней, но если для настоящего поклонника *Mario* этого мало, игра поддерживает индивидуальные уровни. Благодаря тому, что *Mega Mario* — игра с открытым кодом, содержание доступно для внесения в него собственных изменений, через директорию **/usr/share/megamario**. Уровни можно редактировать, меняя простой текстовый файл; музыка (MID), звуки (WAV) и спрайты/плитки (PNG) также находятся в соответствующих директориях. Разрешение видео в игре по умолчанию всего лишь 640×480, но его можно изменить в файле **~/.megamario/mario.ini**.

› Разрешите себе немного платформенного удовольствия Марио старой школы в *Mega Mario*, но бойтесь летучих врагов!

Копирует оригинальные и интересные элементы Super Mario.

Стрелялка

OilWar

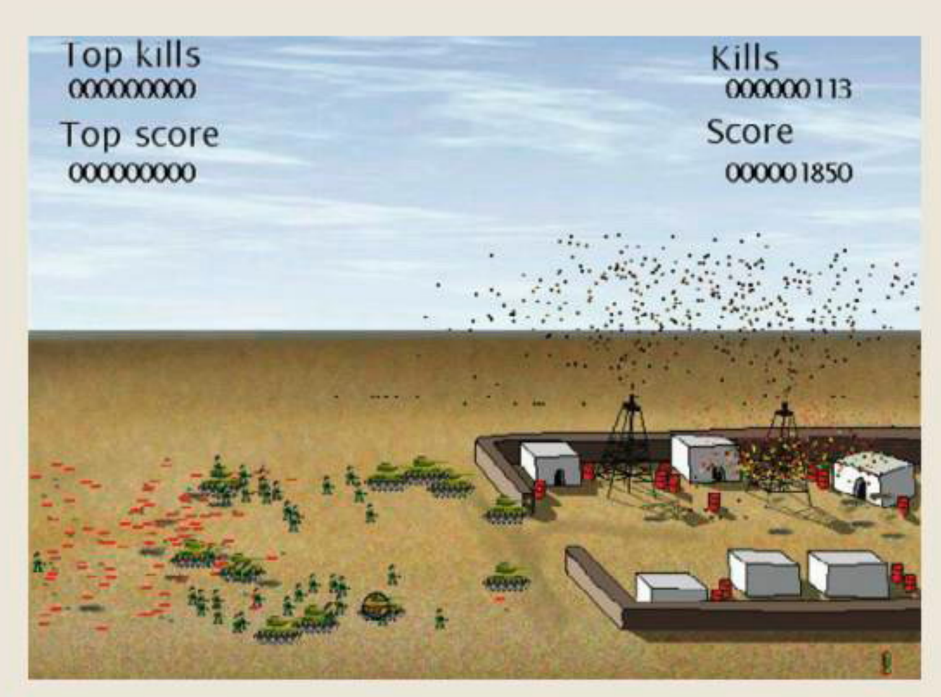
Версия: 1.2.1 **Сайт:** http://bit.ly/OilWar

Хоть *OilWar* и создана много лет назад, но все, кто в курсе мировых новостей, заметят, как современна тема этой небольшой игры. *OilWar* — простая и забавная игра, в которой вам надо защищать нефтепромысел со скважинами и буровыми вышками от наступления армии. Мы не совсем уверены в том, кто здесь хороший, а кто плохой, но двое финских разработчиков, которые и создали эту игру, вероятно, вдохновлялись иракским конфликтом и американской армией в 2000-х. Как бы то ни было, *OilWar* мигом захватила наше внимание и развлекала нас довольно долго. Оказалось, защищать нефтепромысел бывает превесело.

В *OilWar* больше крови, чем в *Mega Mario* (см. выше). У игрока имеется оружие двух видов: пушка, привязанная к левой кнопке мыши, и тактические наземные ракеты, привязанные к правой кнопке. Когда игра начинается, вражеские солдаты выдвигаются слева в штыковую атаку, усиленную танками. Одного щелчка левой кнопки достаточно, чтобы вывести из строя одного солдата, а вот на уничтожение танка уйдет пять щелчков. Когда вам покажется, что врагов стало слишком много, значит, пора запустить ракету, которая уничтожит все вокруг и эффективно очистит площадь от войск.

Рано или поздно нападающие войска достигнут своей цели — нефтепромысла; он сгорит, и на том игра закончится. Цель *OilWar* — продержаться как можно дольше и набрать как можно больше очков. Вам в помощь изредка будут появляться дополнительные ракеты, которыми вы сможете стрелять.

OilWar включена во многие дистрибутивы, такие, как Gentoo, Arch, Rosa и ряд других. Если не найдется бинарник для вашей системы, можете скомпилировать игру из исходника.



› В *OilWar* вы защищаете нефтяное месторождение от надвигающейся армии, используя пушки и ракеты.

OilWar мигом захватила внимание и развлекала нас долго.

Инструмент рисования

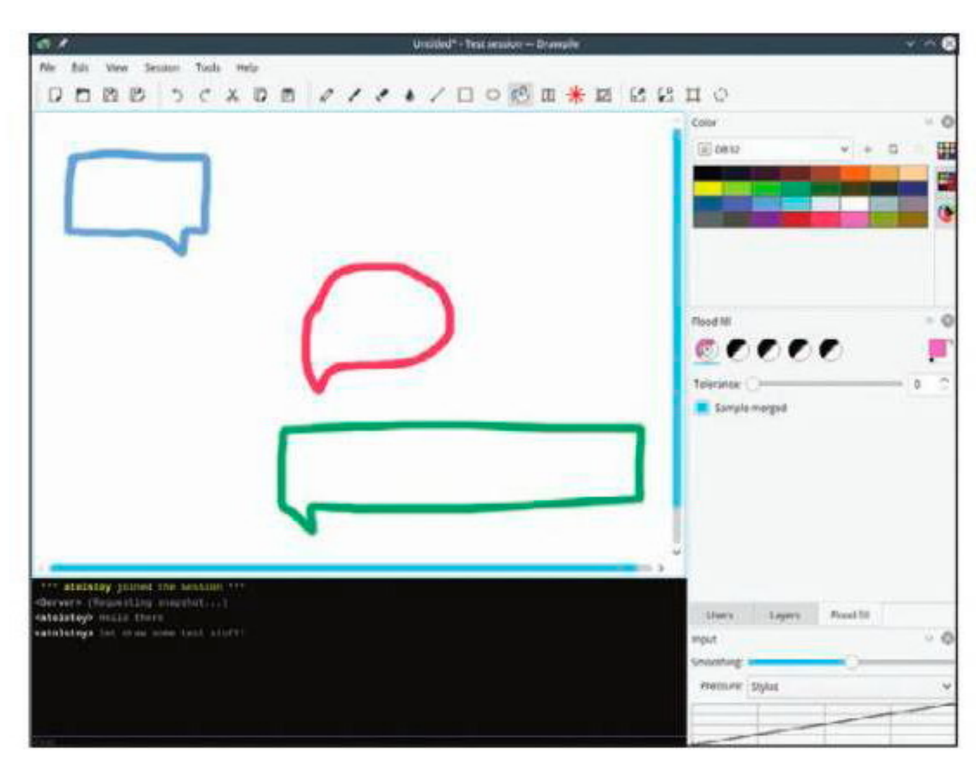
Drawpile

Версия: 1.0.2 **Сайт:** http://drawpile.net

В мире Linux множество графических редакторов; одни мечтают о славе *Adobe Photoshop*, другие хотят быть дружелюбными к пользователю приложениями в стиле *Paint*. Есть также ряд специнструментов редактирования, подобно *CinePaint*, предназначенных для ретуширования фильмов, или *Tux Paint*, разработанного как простой в использовании графический редактор для детей. *Drawpile* — тоже спецприложение: это программа рисования, нацеленная на упрощение коллективной работы.

На первый взгляд, пользовательский интерфейс *Drawpile* выглядит похожим на любой обычный графический редактор, что и неплохо, если вы заодно пытаетесь работать с другими. Панель рисования расположена в верху окна, а область настроек инструментов — справа, и в нее включены обычные элементы, такие, как палитра, кисти, интенсивность и слои, и т.д. Количество инструментов довольно скромное, но их вполне хватает для создания простых схем, эскизов, овалов для текста и даже черновиков деловой графики.

Самая интересная функция в *Drawpile* — это, конечно, коллективная работа. В меню Session вы найдете опции для размещения новой сессии или подключения к существующей. Разместить сессию легко благодаря встроенному в *Drawpile* серверу, который приложение автоматически использует для локальных сессий. Пользователю надо всего лишь предоставить название и логин/пароль, которыми могут пользоваться другие пользователи, чтобы присоединиться к сессии. В стандартной LAN *Drawpile* разместит сессию с IP образца 192.*.*.* или 10.*.*.*, но может также сделать локальные сессии доступными через Интернет. Просто щелкните правой кнопкой по IP-адресу в нижнем правом углу и выберите опцию "Get externally visible IP address". Как только вы разместите сессию или присоединитесь к ней, появится стандартная область чата внизу окна. *Drawpile* также может записывать сессии пользователя в собственном проприетарном формате DPREC. Это не скринкаст, а, скорее, удобный инструмент для воспроизведения ранее выполненных в *Drawpile* действий.



› Сессии в Интернете можно записывать для воспроизведения.

Программа, нацеленная на упрощение коллективной работы.

Утилита ввода

Solaar

Версия: 0.9.2 **Сайт:** https://pwr.github.io/Solaar

Беспроводные мыши и клавиатуры стали исключительно популярны. Стандартный набор батарей может прослужить не один месяц, и вы получаете больше свободы и меньше путаницы проводов. Периферийные устройства Logitech занимают солидную долю на рынке, и велики шансы, что у вас имеется клавиатура или мышь Logitech, или вы видели их в местном компьютерном магазине. Одна из функций Logitech — ресивер Unifying, поставляемый со всеми беспроводными устройствами. Этот ресивер означает, что если у вас есть клавиатура и мышь Logitech, вам не нужны два ресивера одновременно. Вместо этого вы можете присоединить к одному ресиверу до шести устройств Logitech, отключить остальные и высвободить свои порты USB.

До недавнего времени в Linux со всем этим существовала одна проблема: периферийные устройства работали прекрасно, однако у пользователей Linux не было специальной утилиты, способной соединять/отсоединять устройства; отображать их состояние, уровень заряда батарей и т.д. *Solaar* — проект с открытым кодом, который предлагает все эти функции.

Solaar — небольшая утилитка с режимами GUI и командной строки (solaar-cli), которая управляет вашими устройствами из системного лотка или терминала и поддерживает большинство беспроводных продуктов Logitech, использующих ресиверы Unifying и Nano. *Solaar* получает доступ к ресиверу посредством мониторинга устройства **/dev/hidraw0**. Из-за того, как работает в Linux горячее подключение под управлением UDEV, вам, возможно, придется отключить ресивер, а потом снова его подключить — с тем, чтобы *Solaar* его определил. Окно настройки *Solaar* отличается простотой и аскетизмом: слева — дерево устройств, а справа — информация о подключенных устройствах.

› Клавиатуры используют шифрованное соединение, а мыши — нет.

Нажав на значок с лампочкой, вы увидите еще более подробную информацию, в том числе протокол USB HID, настройки частоты и подробности о прошивке. Можете поэкспериментировать с подсоединением и отсоединением, но примите здравый совет: не отключайте все устройства от ресивера, если у вас под рукой нет стороннего периферийного устройства. У *Solaar* также имеется естественное ограничение, не позволяющее подключать одно устройство к более чем одному ресиверу за раз. Если вы попытаетесь это сделать, первый ресивер потеряет соединение. LXF

Управляет устройствами из системного лотка или терминала.

На диске

Дистрибутивы, приложения, игры, книги и всякое-разное...

Лучшее из Интернета, упакованное в 9 ГБ качественного DVD.

Дистрибутивы

Раз мы строим медиа-сервер на Ubuntu, вас может удивить, что в LXFDVD этого месяца не включен ремикс Ubuntu со всеми его пакетами. Ведь мы делали ремиксы с рабочими столами для каждого релиза Ubuntu. Причина в том, что нам больше не разрешают этого делать. Новая политика Ubuntu запрещает распространение измененных версий дистрибутива, если из них не удалены все ссылки на Ubuntu. Нам теперь нельзя даже включить стандартный дистрибутив с обновленными пакетами, не удалив все признаки Ubuntu, что повлекло бы перекомпиляцию многих пакетов, включая ядро. Это непрактично до неосуществимости, особенно когда новая версия выходит незадолго до закрытия номера.

Понятно желание Ubuntu защитить свою торговую марку и репутацию, но наши ремиксы создавались исключительно из пакетов в основных репозиториях Ubuntu и перекомпилировались в Ubuntu Construction Kit — подход испытанный и надежный. Ubuntu немало выиграл в раскрутке благодаря появлениям на дисках нашего журнала. Ubuntu и раньше принимал неоднозначные решения, и хотя это решение не из самых радикальных, все же оно может оказаться непопулярным.

Neil

Основа для нашего медиа-сервера

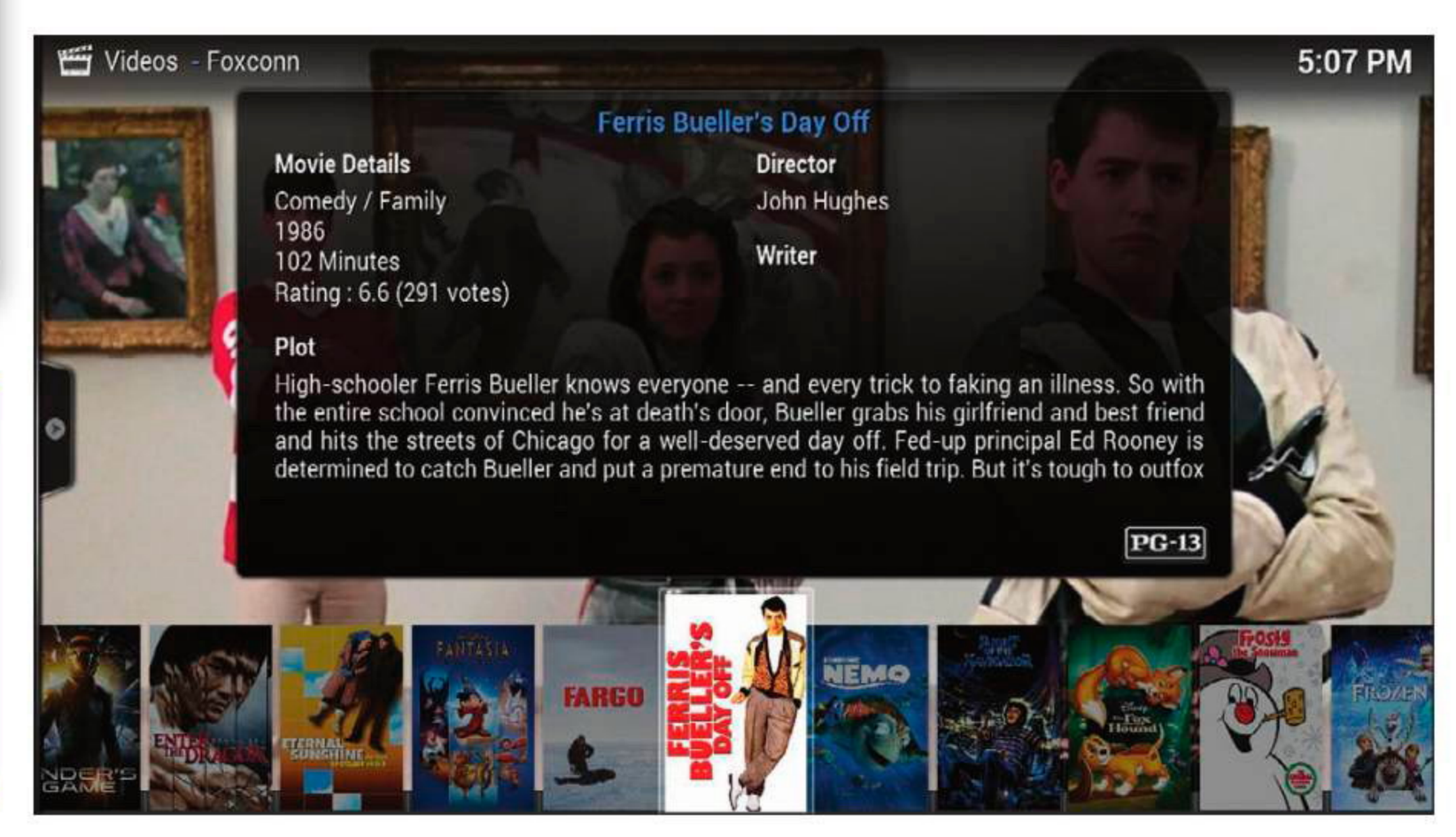
Ubuntu 15.04

У вас наверняка уже есть дистрибутив, в котором можно установить медиа-сервер; а может, вы решили начать с нуля или даже для начала попробовать все на виртуальной машине. Хотя вы можете установить эти программы на любом дистрибутиве Linux, наша основная статья (см. стр. 28) взяла за основу Ubuntu 15.04, так что вот он вам. **LXFDVD** включает 32- и 64-битные базовые версии Ubuntu 15.04. Если у вас 64-битное оборудование, стоит использовать 64-битный дистрибутив. Дни, когда программы лучше работали на 32 битах, давно канули в Лету, и для перекодировки медиа лишняя мощность не помешает.

Король дистрибутивов медиа-плейеров

Kodibuntu 14.0

Kodi — самый популярный медиа-плейер для домашнего кинотеатра; ну как такого не включить! Он доступен в виде live CD как Kodibuntu, и можно в него загрузиться и сразу же начать воспроизведение своих мультимедиа — или установить его на жесткий диск. Правда, Kodibuntu пока не обновлен до самой последней версии *Kodi*. Kodibuntu идеален, если вам нужно удобство загрузки с live CD, чтобы с ним ознакомиться, а установив его, можно потом обновиться до *Kodi 15.1*. Если вы хотите пропустить подход live CD и установить *Kodi* на свой настольный ПК или ноутбук, мы заодно включили пакеты для *Kodi 15.1*. Если вы хотите использовать *Kodi* с *Emby*, есть репозиторий плагинов, устанавливаемых из *Kodi*, но не с live CD. ZIP-файл находится в папке **Emby** на **LXFDVD**, и вы можете установить его прямо изнутри *Kodi*, не распаковывая.



» Важно

ВНИМАНИЕ!

Порченые диски

В маловероятном случае какого-то дефекта вашего **LXFDVD** обращайтесь, пожалуйста, по адресу disks@linuxformat.ru или телефону +7 (812) 309-0686.

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.

Наша подборка медиа-плейеров

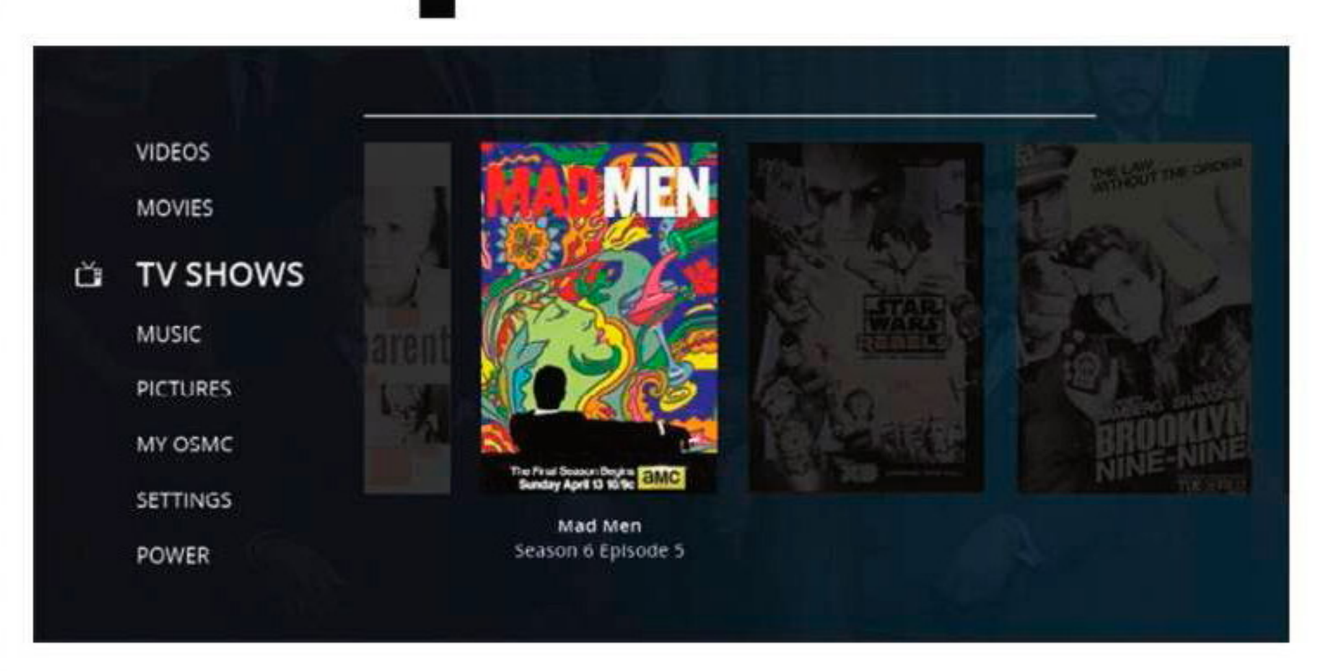
Медиа-стримеры

Чтобы реализовать идеи нашей статьи по созданию медиа-сервера для стриминга, вам понадобятся некоторые программы. Наш DVD включает пакет для *Emby Server* — программы, используемой в статье. Архив называется *MediaBrowser*, это предыдущее название *Emby*, и чтобы запустить его после распаковки, понадобится Mono. Будет лучше установить его из репозиториев своего дистрибутива, если такое возможно, но на крайний случай мы включили и этот пакет.

От медиа-сервера мало проку, если медиа нечем воспроизводить, поэтому мы включили ряд медиа-плейеров для домашнего кинотеатра. Основным выбором является *Kodi*, описанный на предыдущей странице. Еще мы включили OpenELEC и OSMC; у них обоих применяется *Kodi*, и добавлен лишь минимум дистрибутива, чтобы было на чем запускать. Эксплуатируют они старый добрый *Xbox Media Centre* (*XBMC*; это название устарело, и его переименовали в *Kodi*). OpenELEC предлагается в виде gzip-архива ряда файлов образов. Распакуйте тот, что для вашего оборудования, и скопируйте его на устройство флэш памяти — устройство USB для образов ПК и SD-карту для Raspberry Pi. Вы можете и распаковать, и скопировать одной командой в терминале:

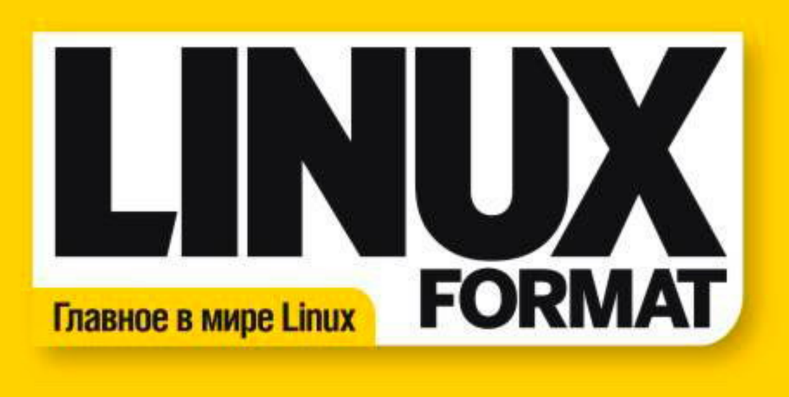
```
$ zcat OpenELEC/OpenELEC-Generic.i386-5.0.8.img.gz | sudo dd
of=/dev/sdX bs=1M
```

где **/dev/sdX** — ваше флэш-устройство. Затем загрузитесь с этого устройства. В случае ПК у вас запустится программа установки, так же, как при загрузке с CD.



OSMC (Open Source Media Centre) тоже идет в виде gzip-архивов образов для Raspberry Pi, и устанавливается так же. В обоих случаях есть отдельные образы для Raspberry Pi Model 2. Улучшенный CPU этой модели делает ее лучшим кандидатом для просмотра контента с высоким разрешением: более старый Raspberry Pi B может испытывать затруднения с видео 1080p. Для установки на ПК имеются пакеты RPM и Debian/Ubuntu, которые скачают и установят для вас последнюю версию — просто установите их с помощью *RPM*, *Yum* или *gdebi* обычным способом. В большинстве дистрибутивов надо просто открыть файл соответствующего пакета в своем менеджере файлов. LXF

OpenELEC и OSMC — минималистские дистрибутивы, предназначенные чисто для запуска *Kodi*, благодаря чему достаточно легковесны для Raspberry Pi, как и для обычного ПК.



Пропустили номер?

Закажите его через сайт www.linuxformat.ru в «ГНУ/Линуксцентре»! Журналы доставляются и в печатной, и в электронной форме, так что с момента открытия браузера до получения нужного вам выпуска **LXF** может пройти всего пара минут!

Прямо сейчас для заказа доступны следующие номера:

LXF201
Октябрь 2015

- **Linux против Windows 10** Пингвин завесит Окна
- **Навязчивый сервер** Дистрибутивы для Web
- **Кто у нас ябеда** Electronic Frontier — о соцсайтах
- **АТС из Кубиков** Cubieboard и Asterisk

LXFDVD: Mageia 5, 4MLinux 13, AntiX 15-V, Tiny Core, Emmabuntüs 3-1.02, GParted Live 0.24.0-2, 11 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_201/
PDF-версия: shop.linuxformat.ru/elxf_201/

LXF202
Ноябрь 2015

- **Академия от Тукса** Учимся кодировать
- **Хромбуки** Коммуникатор стал работягой
- **Раскинем сети** Виртуальные и частные
- **Ищем вакансию** Новостные ленты нам в помощь

LXFDVD: ClearOS, ExTiX, OpenMediaVault, UberStudent, Ubuntu Studio, wattOS, 16 видеоуроков по Linux, 10 книг о Linux, учебники, горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_202/
PDF-версия: shop.linuxformat.ru/elxf_202/

LXF203
Декабрь 2015

- **Дистрибутивы года** Проблема выбора
- **Файловые системы** Их довольно-таки много
- **ОС доверия** Tizen защитит ваши данные
- **Целебный Elixir** Erlang с человеческим лицом

LXFDVD: Bodhi, elementary OS, GALPon MiniNo, Linux Mint, Manjaro, openSUSE, 10 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_203/
PDF-версия: shop.linuxformat.ru/elxf_203/

А чтобы не упустить ничего, оформите подписку! Все, кто подписался на печатную версию журнала через www.linuxformat.ru/subscribe или www.linuxcenter.ru, получают электронную версию в подарок! На сайте shop.linuxformat.ru вы также сможете приобрести предыдущие выпуски **LXF**.

Подписывайтесь на сайте www.linuxformat.ru/subscribe

Телефоны отдела подписки:
- Санкт-Петербург (812) 309-0686
- Москва (499) 271-4954

Linux Format ВКонтакте

Вступайте в нашу группу vk.com/linuxform

На странице LXF ВКонтакте вы найдете:
- Новости о Linux
- Статьи из архива **LXF**
- Анонс свежего выпуска **LXF** и часть статей из него
- Живое общение и онлайн-консультацию по подписке на наше издание

Информация о диске

Внимательно прочтите это перед тем, как использовать LXFDVD!

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

DVD-приложение к журналу Linux Format № 1 (204). Январь 2016

Содержание

LINUX FORMAT

Сторона 1

ДИСТРИБУТИВЫ

Ubuntu 15.04 (32- и 64-битный)
Kodibuntu 14.0 (64-битный)

ПРОГРАММЫ ДЛЯ МЕДИА

Emby
Kodi 15.1
OpenELEC
OSMC

HOTPICKS

Add-on converter (Git) Конвертор расширений
Briss 0.9 Утилита PDF
Drawpile 1.0.2 Инструмент рисования
Mega Mario 1.7 Платформенная аркада
Midori 0.5.11 Web-браузер
Oilwar 1.2.1 Стрелялка
Solaar 0.9.2 Утилита ввода
sView 15.08 Просмотрщик стерео
Terminator0.97-2 Мультиплексор оболочки
Trinity R14.0.1 Среда рабочего стола
ZSH 5.1 Оболочка Unix

Окончание на обороте »

DVD-приложение к журналу Linux Format № 1 (204). Январь 2016

Содержание

ГЛАВНОЕ
CheckInstall
Coreutils
HardInfo
Kernel
Memtest86+
Plop
SBM
WvDial

УЧЕБНИКИ
Astro Pi
Conky
Glade

ПОМОЩЬ
Руководство новичка
Справочники
ЧаВо (FAQ)

Пожалуйста, перед использованием данного диска ознакомьтесь с инструкцией, опубликованной в журнале на стр. 109!

КОММЕНТАРИИ? Присылайте ваши пожелания и предложения по электронной почте: info@linuxformat.ru

ДЕФЕКТНЫЕ ДИСКИ В маловероятном случае обнаружения дефектов на данном диске, обращайтесь, пожалуйста, по адресу disks@linuxformat.ru

Настоящий диск тщательно тестировался и проверялся на всех стадиях производства; однако, как и в случае с любым новым ПО, мы рекомендуем вам использовать антивирусный 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 может повлечь за собой использование этого DVD, представленных на нем программ или данных. Прежде чем устанавливать какое-либо ПО на компьютер, подключенный к сети, проконсультируйтесь с сетевым администратором.

Тираж изготовлен ООО «Маркон», 188652, Россия, Ленинградская область, Всеволожский р-н, дер. Юкки, Школьная ул., 7-а. Лицензия МПТР ВАФ N 77-103.

Сторона 2

ДИСТРИБУТИВЫ
openSUSE-Edu-li-f-e 42.1.1 (64-битный)
Q4OS Live 1.4.6 (64-битный)

Все дистрибутивы на этом диске:
- представлены ISO-образами;
- можно загрузить непосредственно с DVD в live-режиме;
- имеют инсталлятор, запускаемый из этого режима.

Создание установочных дисков при помощи cdrecord

Самый быстрый способ записать ISO-образ на чистую матрицу — это обратиться к программе *cdrecord*. Для всех перечисленных ниже действий потребуются права суперпользователя-root. Сначала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdrecord -scanbus
```

После этого на экране терминала должен отобразиться список устройств, подключенных к вашей системе. SCSI-адрес каждого устройства представляет собой три числа в левой колонке — например, 0,3,0. Теперь вы можете с легкостью записать образ на диск:

```
cdrecord dev=0,3,0 -v /путь к образу/image.iso
```

Чтобы упростить дальнейшее использование *cdrecord*, сохраните некоторые настройки в файле **/etc/default/cdrecord**. Добавьте по одной строке для каждого устройства записи (скорее всего, в вашей системе присутствует только одно такое устройство):

```
Plextor= 0,3,0 12 16M
```

Первое слово в этой строке — метка; затем после адреса SCSI-устройства вы должны указать скорость и размер буфера. Теперь можете заменить SCSI-адрес в командной строке на выбранную вами метку. Все будет еще проще, если вы добавите следующее:

```
CDR_DEVICE=Plextor
```

Для записи ISO-образа вам осталось набрать команду

```
cdrecord -v /path/to/image.iso
```

Если вы не принадлежите к любителям командной строки, в таком случае вам придет на помощь утилита *gcombust*. Запустите ее от имени root и выберите вкладку Burn и ISO 9660 Image в верхней части окна. Введите путь к образу, который вы хотите записать на диск, и смело нажимайте на Combust! Пока ваш образ пишется на диск, можете выпить чашечку кофе.

Другая ОС?

Использовать Linux для записи компакт-диска не обязательно. Программы вроде *cdrecord* просто переносят двоичные данные на чистую матрицу. Все необходимые файлы уже включены в ISO-образ, который распознается любой операционной системой, будь то Linux, Windows, Mac OS X или AmigaOS.

Нет устройства для записи дисков?

Если у вас нет устройства, с помощью которого можно было бы записать образ на диск, можно найти какого-нибудь друга или организацию, у кого есть компьютер с дисководом, и прожечь диск у них. Опять-таки, вам подойдет любая операционная система, способная распознать пишущий привод (см. выше).

Некоторые дистрибутивы умеют монтировать образы дисков и выполнять сетевую установку или даже установку с раздела жесткого диска. Конкретные методы, конечно, зависят от дистрибутива. За дополнительной информацией обращайтесь на web-сайт разработчика дистрибутива.

softline®

Cloud Software Hardware Services

20+ Years in IT

IT-архитектура вашего бизнеса

корпоративные мобильные решения на платформе Microsoft

SaaS

CLOUD

гибридные облака

SOLUTIONS

SAM

поставка аппаратных решений

юридическая поддержка

remote workplace

PROJECTS

VIRTUALIZATION

LINUX

программное обеспечение

BI & BigData

hardware

web-разработка

legal support

education

ERP&CRM

защита персональных данных

SERVICES

WinSpace

Mirapolis

разработка IT-стратегий

CAD CAM CAE

SOFTWARE

публичные облака

САПР и ГИС

информационная безопасность

решения Microsoft

SECURITY

учебный центр

техническая поддержка

search

flexible

CONSULTING

data leak prevention

разработка информационных систем

корпоративный портал DeskWork

open source

PLATFORM

IaaS

protection

hosting

technical support

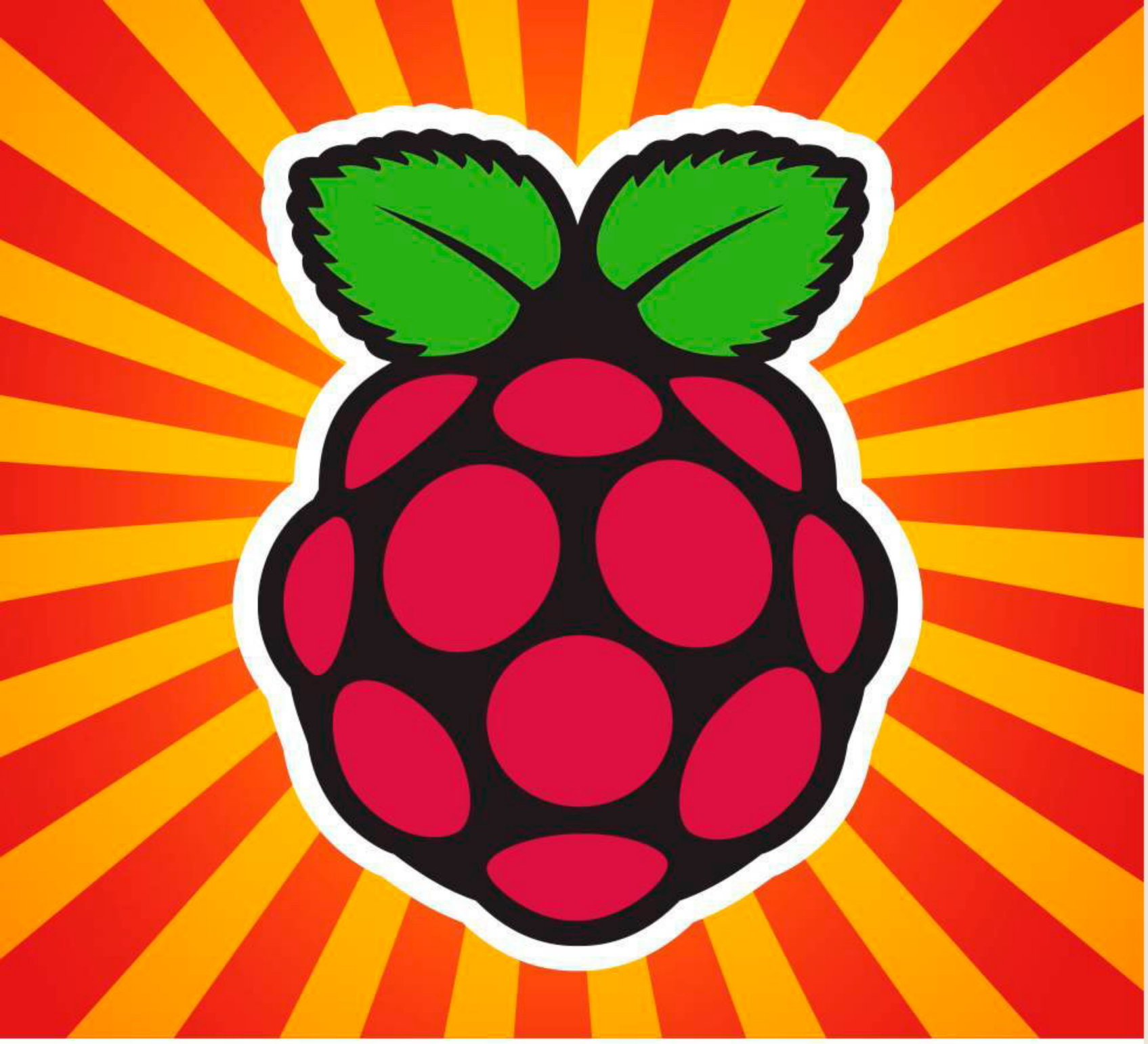
PaaS

HARDWARE

частные облака

16+

8 (800) 232-00-23 | +7 (495) 232-00-23 | www.softline.ru | info@softline.ru



В февральском номере

С Raspberry Pi живется лучше!

Автоматизируйте свой дом — от и до — при помощи Linux и Raspberry Pi.

Видеоплейеры

Желаете наслаждаться самым плавным воспроизведением видео 4K? Мы все перепробовали, но *VLC* по-прежнему лидер.

Крутой KDE 5!

Свечение KDE Plasma позволит оглядеть свои компьютеры и решить, пригодится нам это или нет.

Пишите письма

Тех, кто до сих пор с содроганием вспоминает повинность рассылки новогодних поздравлений, мы вынуждены огорчить: это можно было автоматизировать.

Содержание будущих выпусков может меняться — вдруг при температуре плазмы мы спалимся...

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 2000 экз., распространение электронной версии 30 000 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
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. б/н, лит. Ф
Тел. (812) 462-8383, e-mail: office@ldprint.ru
Заказ 13576

РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ

Редактор Нейл Мор [Neil Mohr] neil.mohr@futurenet.com
Научный редактор Джонни Бидвелл [Jonni Bidwell] jonni.bidwell@futurenet.com
Выпускающий редактор Крис Торнетт [Chris Thornett] chris.thornett@futurenet.com
Художественный редактор Эфраин Эрнандес-Мендоса [Efrain Hernandez-Mendoza] efrain.hernandez-mendoza@futurenet.com

ПОДГОТОВКА МАТЕРИАЛОВ

Джонни Бидвелл [Jonni Bidwell], Нейл Ботвик [Neil Bothwick], Джолион Браун [Jolyon Brown], Крис Браун [Chris Brown], Алекс Кэмпбелл [Alex Campbell], Алекс Эймс [Alex Eames], Мэтт Эллиотт [Matt Elliott], Рич Мак-Кормик [Rich McCormick], Нейл Мор [Neil Mohr], Чуонг Нгуен [Chuong Nguen], Ник Пирс [Nick Peers], Лес Паундер [Les Pounder], Маянк Шарма [Mayank Sharma], Джим Тэккер [Jim Thacker], Александр Толстой [Alexander Tolstoy], Евгений Балдин, Андрей Гондаренков, Сергей Коклеев, Юрий Малолетко, Дмитрий Пантелеичев, Алексей Федорчук, Лада Шерышова, Игорь Штомпель

Иллюстрации Шейн Коллинджа [Shane Collinge]

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.futureplc.com.

© Linux Format 2005
© Future Publishing Ltd 2005

16+

BATH • LONDON • MILAN • NEW YORK • PARIS • SAN DIEGO • SAN FRANCISCO

Добро пожаловать в робототехнику!

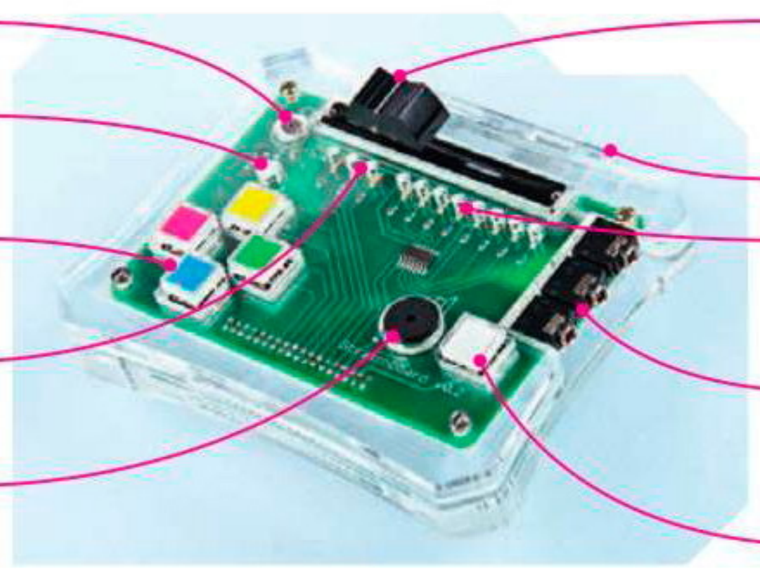
ScratchDuino

Электронный комплекс на основе свободного аппаратного обеспечения для школ и вузов

ScratchDuino.Лаборатория

Плата расширения с датчиками ввода-вывода — как установленными на плате, так и подключаемыми. Предназначена для программирования взаимодействия компьютера с внешними устройствами.

Полная интеграция («из коробки») со средой программирования Scratch, предназначенной для детей.



* РАБОТАЮТ В РЕЖИМЕ ЭМУЛЯЦИИ PICOBOARD, НЕ ТРЕБУЯ ДОПОЛНИТЕЛЬНОГО ПРОГРАММИРОВАНИЯ.

ЦЕНА **9500 руб.**

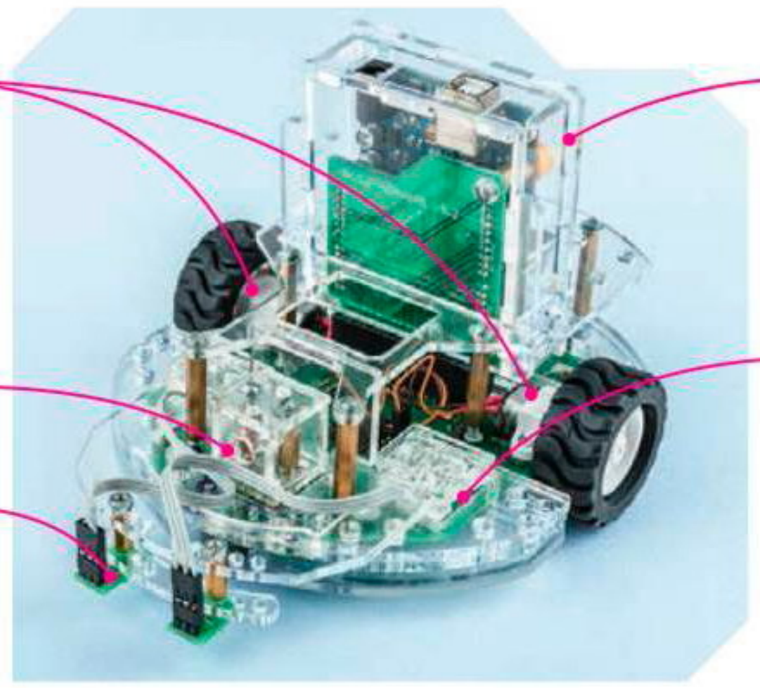
ГАРАНТИЯ 3 ГОДА, ВКЛЮЧАЯ ТЕХНИЧЕСКУЮ И МЕТОДИЧЕСКУЮ ПОДДЕРЖКУ

ScratchDuino.Робоплатформа

Внешний робот-исполнитель, управляемый из среды программирования Scratch. Не требует навыков программирования на языках высокого уровня и может применяться для обучения, начиная с младших классов.

ScratchDuino.Робоплатформа обеспечивает возможность:

- подключения внешних датчиков без механических соединений и пайки;
- расширения через последовательную шину RoboBus;
- установки деталей Lego Technics.



* ВСЕГО НА РОБОПЛАТФОРМЕ 5 РАЗЪЕМОВ. ВОЗМОЖНО ПОДКЛЮЧЕНИЕ ДАТЧИКОВ КАСАНИЙ, ИК-ДАТЧИКОВ РАССТОЯНИЙ/ПРЕПЯТСТВИЙ И ДР.

ЦЕНА **19 500 руб.**

ГАРАНТИЯ 3 ГОДА, ВКЛЮЧАЯ ТЕХНИЧЕСКУЮ И МЕТОДИЧЕСКУЮ ПОДДЕРЖКУ

Назначение ScratchDuino

- Изучение взаимодействия компьютера с внешней средой
- Изучение изменения параметров внешней среды
- Изучение процессов передачи информации и принципов ее построения
- Изучение внешних устройств управления
- Моделирование устройств

Среда разработки

Lazarus (язык Pascal) ■ Scratch ■ Arduino IDE

ScratchDuino адаптирован для учебных заведений. Поставляется с комплектами учебно-методических материалов.

Продукт разработан при финансовой поддержке Фонда содействия развитию малых форм предприятий в научно-технической сфере

WWW.SCRATCHDUINO.RU

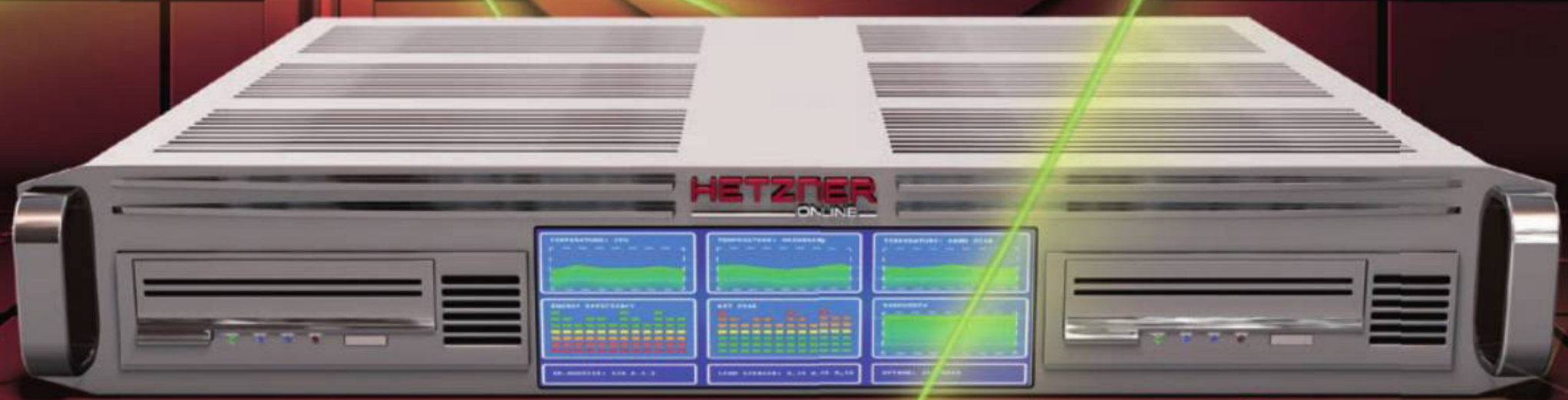
(812) 309-0686 (Санкт-Петербург) ■ **(499) 271-4954** (Москва)

HETZNER DEDICATED SERVERS

HETZNER ONLINE

В ЦЕЛОСТИ И СОХРАННОСТИ!

ХРАНЕНИЕ ДАННЫХ В ГЕРМАНИИ



ВЫДЕЛЕННЫЙ СЕРВЕР PX60

- Intel®Xeon® E3-1270 v3 Quadcore Haswell
- 32 ГБ ECC RAM
- 2 x 2 ТБ 6 Гбит/с SATA 7200 об/м Enterprise класс Software-RAID 1
- 30 ТБ Трафик*
- Операционная система Linux
- Подключение 1 Гбит порт
- Гарантировано 200 Мбит/с
- IPv6 подсеть (/64)
- Без минимального контракта

2600 рублей в месяц
+ Установка 3700 рублей

intel inside XEON

ВЫДЕЛЕННЫЙ СЕРВЕР PX70-SSD

- Intel®Xeon® E3-1270 v3 Quadcore Haswell
- 32 ГБ ECC RAM
- 2 x 480 ГБ 6 Гбит/с SATA SSD Data Center Series
- 50 ТБ Трафик*
- Операционная система Linux
- Подключение 1 Гбит порт
- Гарантировано 200 Мбит/с
- IPv6 подсеть (/64)
- Без минимального контракта

3700 рублей в месяц
+ Установка 3700 рублей

intel inside XEON

* Нет платы за превышение. При превышении 30 ТБ/месяц (PX60), 50 ТБ/месяц (PX70-SSD) скорость соединения ограничивается (подсчёт ведётся по исходящему трафику, входящий и внутренний трафик не учитывается). Опционально можно снять ограничение, подтвердив оплату 80 рублей за каждый дополнительный ТБ.

Безопасное защищённое хранение ваших данных в собственных дата-центрах немецкой компании Hetzner Online, соответствующее высоким требованиям к стандартам защиты данных в Германии.

RU.HETZNER.COM

100% Green Electricity Energy-efficient Hardware

GreenIT 2011 Best Practice Award

Hetzner Online активно поддерживает защиту окружающей среды, используя исключительно 100% возобновляемые источники энергии.Выбирайте более чистое будущее вместе с Hetzner Online!

Все цены указаны без учёта НДС и регулируются условиями предоставления услуг компанией Hetzner Online AG.
Цены могут быть изменены. Все права защищены соответствующими производителями.