

DVD ДАРОМ!



FEDORA 23



# LINUX FORMAT

Главное в мире Linux

Февраль 2016 № 2 (205/206)

# ХАКНЕМ!

## Чудеса в доме на Raspberry Pi

- » Шесть веселых проектов на пробу
- » Автоматизируем освещение
- » Мониторим отопление
- » Обезопасим свой дом

**70** страниц учебников и статей!

- » Открываем KDE Plasma 5
- » Почтовые рассылки в LibreOffice
- » Создаем совместные ресурсы



Академия кодига:  
Perl 6 и Lua

ЕСТЬ ЭЛЕКТРОННАЯ ВЕРСИЯ ДЛЯ ВАШЕГО ПЛАНШЕТА!



## Глубиной в океан

« У нас 8 дистрибутивов — надо много изучать, если хочешь знать их от и до »

Тэмми Бютов — о web-хостинге для всего мира с. 38

Управление дисками

### Освоим LVM

- » Добьемся полного контроля над своими дисками



Управление системой

### Настроим Conky

- » Скомбинируем Lua и Conky, создав монитор системы



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

Linux center  
www.linuxcenter.ru



## Плюс: Для Pi!

10-страничный гид по Raspberry Pi

- » Astro Pi: пользуемся AstroCam
- » Raspbian Jessie — Debian 8
- » Цепляем контроллер Xbox





# iTeleRadio

ИНТЕРАКТИВНОЕ РАДИО&ТЕЛЕВИДЕНИЕ

iteleradio.ru

## Что мы делаем

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



## Кто мы

Вот от чем мы спросили наших экспертов: Linux и его разработчики очень умны, но что в мире Linux и в их доме могло бы быть умнее?



### Джонни Бидвелл

Дуглас Адамс говорил, что он редко «бывал счастливее, чем когда тратил целый день, программируя свой ПК на автоматизацию задачи, иначе отобравшей бы добрые 10 секунд на выполнение вручную». Его чувства я вполне разделяю, и это мое обычное оправдание при задержках статей для LXF.



### Нейл Ботвик

Львиной долей моей жизни движут скрипты Python и оболочки. Моя цель в автоматизации дома — догнать Уоллеса из мультика «Неправильные штаны». Правда, не пришлось бы управлять через Windows, ведь один из уроков этого фильма — то, что впуск пингинов в дом чреват катастрофами...



### Мэтью Хансон

Я люблю мастерить и собирать дома персональные компьютеры, что неизбежно требует устранения неполадок и выяснения, где я накосячил, всё поломал. А это бесит; вот и нужен робот, который трудился бы за меня. Конечно, для починки этого робота надо бы построить другого... И еще одного...



### Лес Паундер

Я люблю пройтись с моим псом, но холодная погода все-таки ужасна, хотя борода меня и греет. GPS-контролируемый робот для выгуливания собак был бы кстати. Я бы задавал маршрут и планировал сроки с телефона, посылая обновления через Twitter. А по приходу домой угощал бы его хлецем.



### Ник Пирс

Охотно бы автоматизировал дома освещение: и для окружающей среды хорошо, и для моего кошелька. И смягчило бы эффект явной аллергии моего семейства на выключение света при выходе из комнаты. А в сложной ситуации я мог бы инициировать «Режим диско» для семейного конкурса по танцам.



### Валентин Синицын

Хорошо, что в Linux всегда можно залезть в исходники и посмотреть, как они работают. Плохо, что в Linux иногда приходится залезать в исходники, чтобы понять, как это вообще должно работать. В общем, даешь больше хорошей документации без ошибок!



## Сеть для всего

» Похоже, «Интернет вещей (IoT, Internet of Things)» становится главным «трендом ИТ-шной моды» этого сезона. Одна фирма даже выпустила IoT-редакцию своей операционной системы. А уж слово smart к названию своего изделия не добавил только ленивый.

Всё это было вполне предсказуемо. Изобретатель Ethernet Роберт Мэткалф [Robert Metcalfe] еще тридцать лет назад предположил, что полезность сети растет пропорционально квадрату количества узлов («Закон Мэткалфа»). А технические системы, как учил основатель ТРИЗ Генрих Саулович Альтшуллер, всегда развиваются в сторону увеличения их полезности. По этой логике «сеть для людей» неизбежно станет «сетью для всего».

Новая технология часто порождает слишком радужные надежды. В 1950-е годы атомные поезда, самолеты и автомобили казались делом ближайшего будущего, а сегодня гражданские атомные суда можно пересчитать по пальцам одной руки. Восторг от новизны идеи с неизбежностью переходит в оценку рисков.

В современной реализации IoT мне видятся две проблемы: безопасность интерфейса управления и, в некоторых случаях, сама его архитектура. Почему я могу управлять кофеваркой только из мобильного приложения и через облако? Неплохо бы опубликовать интерфейс, а еще лучше — придумать открытый протокол управления «сетевыми» вещами. Нечто вроде SNMP для слишком умных кофе-машин.

### Кирилл Степанов

Главный редактор

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

## Как с нами связаться

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

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

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

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

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

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

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

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

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

# ПОДПИШИСЬ НА LINUX FORMAT!

Оформи подписку на печатную версию журнала Linux Format на 2016 год и получи в подарок диск с архивом номеров 2005—2014, а также подписку на электронную версию в формате PDF



## SHOP.LINUXFORMAT.RU

### Варианты доставки

- » Почтой по России простой или заказной бандеролью
- » Курьерской службой СПСР или IML
- » В виде PDF-файлов для подписчиков электронной версии

### Редакция

- » Санкт-Петербург: пр. Медиков, 5, корп. 7, (812) 309-0686
- » Москва: (499) 271-4954

# Доверяйте свой контент только профессионалам

Сеть доставки контента CDNvideo с 2010 года обеспечивает качественным, разнообразным и бесперебойным сервисом клиентов в России, СНГ, Европе и Юго-Восточной Азии.

## Бесперебойные онлайн-трансляции



Высокая скорость и качество видео  
Вещание на неограниченную аудиторию  
Корректное отображение на любых устройствах  
Множество дополнительных опций

## Мультимедиа по запросу



Стабильно высокая скорость загрузки  
Без буферизаций и зависаний  
Не требует капитальных затрат  
Снижение нагрузки на каналы связи

## Ускорение интернет-сайтов



Улучшает позиции в поисковой выдаче  
Положительно влияет на конверсию  
Повышает лояльность пользователей  
Улучшает пользовательский опыт в регионах

Подробности на сайте <http://cdnvideo.ru> и по телефону +7 495 212-02-46

CDNvideo – ведущий провайдер услуг сети доставки контента (CDN) в России и СНГ.  
Ёмкость сети составляет более 500 Гбит/с, в том числе в Москве – более 300 Гбит/с.  
Доступность сети в 2013-2015 гг. составила более 99,99%.

# Содержание

«Преодоление препятствий выковывает сильный характер». Арнольд Шварценеггер

## Обзоры

### openSUSE Leap 42.1 ..... 12

Напрыгивать на него или отказаться от геккона полностью? Наш вердикт дистрибутиву без Live-версии.

### BakAndImgCD 14.0 ..... 13

Может, оно и идеально для резервного копирования, создания образов системы и восстановления при сбоях, но... придется полюбить терминал: это все, что вы получите.

### RockStor 3.8.7 ..... 14

Решение NAS на базе Linux, которое пообещало ПО управления хранилищем из ваших сладких снов. Правда, нам показалось, что сны могли быть и получше.

### MotionEyeOS 20151011 ..... 15

Превратите ваш Raspberry Pi в систему CCTV уровня Pro, сочетая множество камер и данное программное обеспечение. Разберемся, как там Большой Брат это делает...

### MySQL Router 2.0.2 ..... 16

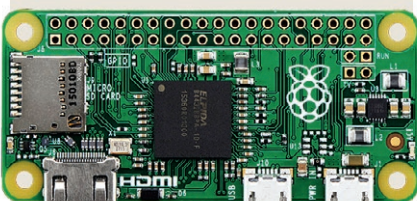
Высокоскоростной кластер серверов нуждается в обслуживании связей между этими серверами, и наш программный маршрутизатор неплохо справляется с задачей.



› Модульная структура позволяет пристыковывать сторонние плагины.

### Raspberry Pi Zero ..... 18

Никто не ожидал такого, что учудили в Raspberry Pi Foundation! А они выкатили новый Pi, самый маленький и самый дешевый из всех!



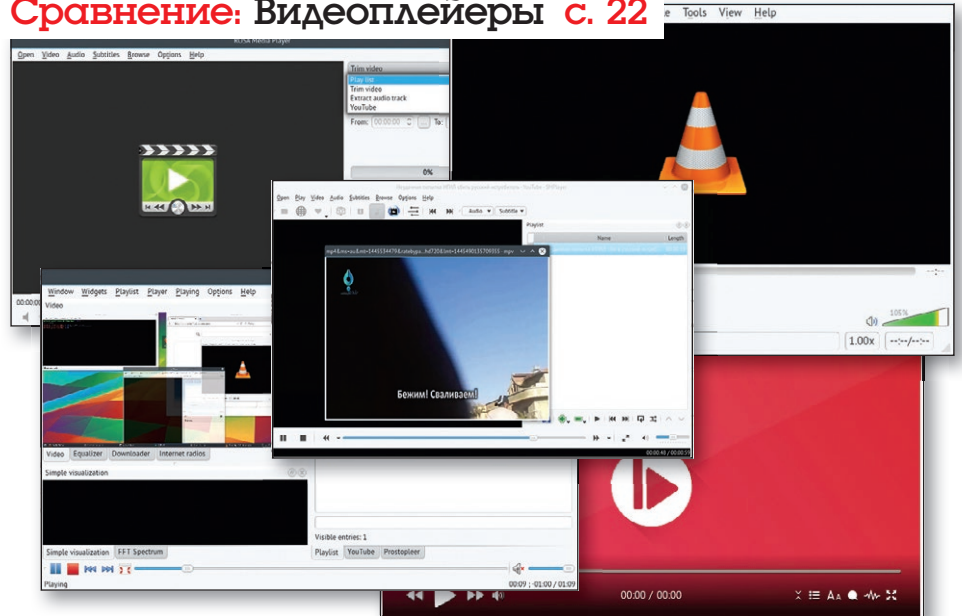
› Вы думали, таких крошечных ПК не бывает? А вот вам Pi Zero.

## ХАКНЕМ!

Превратите свое жилище в умный дом благодаря проектам на Pi и инструментам Linux с. 28



## Сравнение: Видеоплееры с. 22



## Интервью



«Навыки критического мышления и решения задач помогают во всем.»

Тэмми Бютов — о втором в мире хостинг-провайдере с. 38

# На вашем бесплатном DVD

**НАЧИНАЕМ РАБОТУ В LINUX**

- Вставьте диск
- Загрузитесь
- Работайте в Linux!

**UBUNTU 15.10**

Свежий релиз с 8 рабочими столами, готовыми к запуску и установке

64-битная версия и 32-битная стандартная

**KUBUNTU**  
Для рабочего стола Plasma 5

**Ubuntu 15.10** 64-битный с **Unity, Mate, LXDE, Xfce,**  
**Ubuntu 15.10** 32-битный,  
**Kubuntu** 64-битный, **Fedora 23** 64-битный

» Каждый месяц — только лучшие дистрибутивы

Побалуйте себя и любимых подпиской на LXF!



Доступно в AppStore!



[www.linuxformat.ru/subscribe](http://www.linuxformat.ru/subscribe)

## Пользователям Raspberry Pi



### Новости Pi ..... 88

Что творится в Pi Foundation, включая новое партнерство с Code Club.

### Raspbian Jessie ..... 89

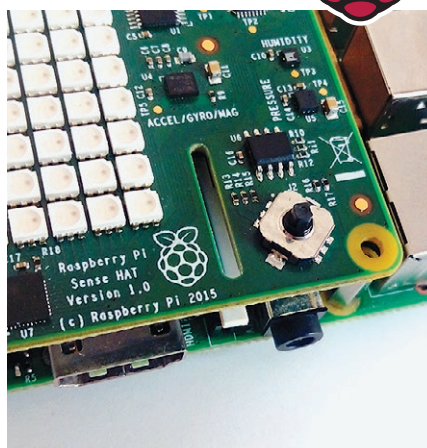
Откройте для себя новые функции повзрослевшей официальной ОС для Raspberry Pi.

### AstroCam ..... 90

Наворотим на Astro Pi плату расширения и камеру — и сможем делать фантастические фотографии.

### Игровой контроллер Xbox .... 92

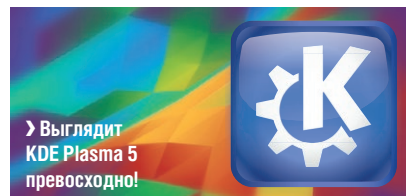
Подцепим его к своему Raspberry Pi, чтобы наслаждаться консольными играми — как современными, так и «ретро».



## Ищите в номере

### Открытие KDE Plasma 5 ..... 42

Рабочий стол KDE 5 отменяется: вместо него встречайте KDE Plasma 5! Разбираемся, что такое KDE Framework 5 и прочие прибамбасы этого замечательного окружения.



» Выглядит KDE Plasma 5 превосходно!

### Samsung вступает в Тайзен.Ру .... 54

Российская ассоциация, продвигающая ОС Tizen, заручилась поддержкой гиганта ИТ-индустрии,



## Академия кодига

### Lua и Conky ..... 80

Лучший способ изучить Lua — приняться писать скрипты, а в случае с Conky скрипты будут еще и небесполезны для контроля за системой.

### Perl 6 ..... 84

Не прошло и 14 лет, как мы наконец-то дождались этой новой версии. Кое-кто всё равно заявит «Да уж лучше Python», но прогресс оказался существенным.



## Постоянные рубрики

### Новости ..... 6

В Штатах боятся роботов, аккумулятор будущего не перегреть, автомагнатов тянет в Linux, вирусологические учли критику, диск на 8 ТБ сделан для NAS, web-сервер влез в карман, Oracle побил рекорд по исправлениям, а Google раскрыл зонтик над Fuel.

### Новости мобильных ОС ..... 20

Sortana временно оглохла, Lenovo танцует Tapco, рынок мобильных ОС диверсифицируется, а Android братается с мышами и прорывается на ПК.

### Сравнение ..... 22

Наполняйте свой взор живыми картинками только при посредстве лучших видеоплееров: *Bomi, QMplay2, Romp, SMPlayer, VLC*.

### Интервью LXF ..... 38

**Тэмми Бютов** выработала характер, занимаясь скейтбордингом, и махнула за океан, чтобы заняться интернет-хостингом.

### Рубрика сисадмина .... 46

Жизнь м-ра **Джюлиона Брауна** очень украсил *Kubernetes*, но он не остановился на достигнутом и открыл охоту на ELK, что по-английски — «лось». Этот лось умеет управлять логами!

### Ответы ..... 96

#### ВАШИ ПРОБЛЕМЫ РЕШЕНЫ!

**Нейл Ботвик** — про слитком чуткий таппад, общение со старыми серверами SSH, монтирование устройств Android, проблемы с беспроводным подключением, медленный запуск рабочего стола, уничтожение данных.

### HotPicks ..... 100

Отвечайте горяченько! Лучшие в мире новинки свободного ПО: *Blender, Bomi, eXeLearning, FLIF, Liri, PulseAudio, QMMP, SANE, Solar Wolf, SweetHome3D, TORCS*.

### Диск Linux Format ..... 106

Содержимое двустороннего DVD этого месяца.

### Пропустили номер? ... 108

Вам не достался наш предыдущий выпуск? Так жить нельзя! Вы ж не попадете в рай потокового вещания медиа!

### Через месяц ..... 112

Пора делать ноги из мира Windows 10 — установите Linux на своем ПК и ПК друзей.

## Учебники

### Libre Office

#### Слияние почты и БД ..... 56

Если вам требуется массовая рассылка писем, почему бы не автоматизировать эту операцию?

### Управление томами

#### LVM подробно ..... 60

Обретаем полный контроль над своими дисками.

### RAID

#### Организуем диски ..... 62

Всё необходимое для создания и поддержки своих собственных RAID-массивов.

### GParted

#### Создаем разделы ..... 64

Добавим и поизменяем разделы на дисках.

### Развертывание систем

#### Fog без тумана ..... 68

Развернем клон удачной системы на всех ПК.

### ZuluCrypt

#### Шифрование диска ..... 70

Зароем важную информацию под шифром.

### IP-телефония

#### Городские — не такие ..... 72

Подключимся через Asterisk к городской телефонной сети общего назначения.

### MySQL Fabric

#### Первое знакомство ..... 76

Освоим ПО для управления фермами серверов с высокой доступностью.



**В ЭТОМ НОМЕРЕ:** Боязнь бунта машин » Неопалимая батарея » Автомобильные магнаты и Linux » Вирус исправился » Диск на 8 ТБ » Web-сервер в кармане » Рекорд по патчам » Зонт для Fuel

## ДОМАШНИЕ РОБОТЫ

# В США не доверяют роботам

Но к 2020 г. робот появится в каждом десятом американском доме.



» Рубрику готовил  
**АНДРЕЙ  
ГОНДАРЕНКОВ**

**J**uniper Research прогнозирует резкий рост популярности домашних роботов в ближайшие 4 года. Согласно представленному компанией исследованию, к 2020 г. потребительский робот будет, по крайней мере, в каждом 10-м американском доме (сейчас в Америке один робот приходится на 25 домов).

Конечно, по своим возможностям эти устройства пока не дотягивают до Rosie из мультсериала *The Jetsons*, и не могут выполнять обязанности горничных и поваров; однако с такими домашними делами, как, например, стрижка газона или уборка пылесосом, первые домашние роботы справляются прекрасно. Некоторые отели уже используют роботов в качестве

разносчиков или гидов, домашние же пользователи начинают с более простых машин, таких как робот-пылесос iRobot Roomba.

«Состояние потребительской робототехники сравнимо с положением персональных компьютеров в конце 1970-х, — заявил Штеффен Зоррель [Steffen Sorrell], аналитик Juniper Research. — Венчурный капитализм и корпоративные инвестиции пошли здесь совсем недавно. Инвесторы знают, что стартовала новая парадигма использования машин и взаимодействия с ними. Появление в наших домах более сложных и умелых роботов приведет к прорыву в технологиях, а также к падению цен».

Ключевым моментом на пути к размножению роботов в американских домах

исследователи считают наличие доверия. Выходящие на рынок новые роботы — это большие, сильные машины, и людям необходима достаточная уверенность, что роботы не причинят вреда им либо их имуществу. Формирование такого доверия является большой проблемой.

» Используя визуальную систему оперативного ориентирования и составления карты (vSLAM), робот-пылесос строит карту помещения с предметами интерьера и знает свое местонахождение.



## НАНОТЕХНОЛОГИИ

# Аккумулятор будущего

Он не боится перегрева: наноразмерные никелевые шипы его спасут.

**Л**итий-ионные аккумуляторы широко применяются в портативной электронике. Их главные достоинства — малый вес и большая энергоемкость, они легко перезаряжаются. А основной недостаток — склонность к перегреву при повреждениях: короткое замыкание в такой батарее часто приводит к пожару. Исследователи из Стэнфордского университета создали литий-ионный аккумулятор, способный отключаться при перегреве.

В новой батарее использовали полимерную пленку со встроенными частицами никеля с шипами наноразмеров. Шипы покрыты графеном — проводящим материалом, по поверхности которого протекает

электрический ток. От нагревания пленка расширяется, и при температуре около 70°C контакт между проводящими шипами пропадает, отключая батарею. Неуправляемая тепловая реакция пресекается, аккумулятор остывает, и между шипами снова образуется контакт, возобновляя ток.

«Выбором количества частиц или типа полимерных материалов можно повысить или снизить температуру отключения», говорит Чжэньань Бао [Zhenan Bao], участница исследовательской группы и преподаватель химического машиностроения в Стэнфорде. В работе также участвовали инженер И Ци [Yi Cui] и ученый Чжэнь Чэн [Zheng Chen]. «Наш проект предоставляет

» Исследователи из Стэнфорда используют для отключения литий-ионной батареи при сильном нагреве полимерную пленку.



надежную, быструю, обратимую стратегию, которая обеспечивает и высокую производительность батареи, и повышенный уровень безопасности», считает И Ци.

Бао уже применяла никель-полимеры в носимых датчиках температуры тела.



ВСТРАИВАЕМЫЙ LINUX

# Linux в автомобиле

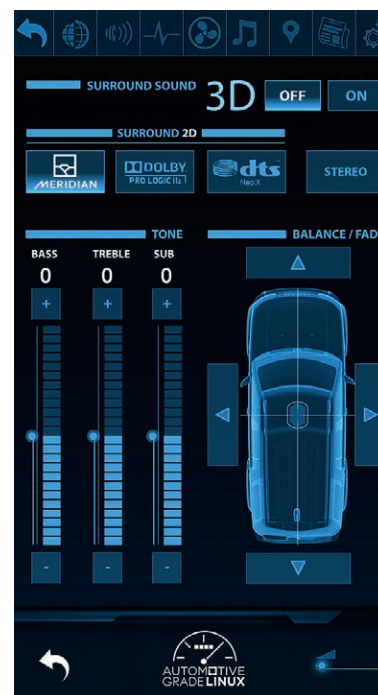
Расширен состав рабочей группы AGL и представлен новый дистрибутив.

**В** начале января в состав Automotive Grade Linux (AGL) — специальной рабочей группы при Linux Foundation, разрабатывающей открытый программный стек на Linux для автомобилей — вошли Subaru (Fuji Heavy Industries, Ltd.), Mitsubishi Motors, Mazda Motor Corporation и Ford Motor Company (ставшая первым производителем автомобилей в США, присоединившись к AGL). В составе группы уже присутствуют такие гранды, как Toyota Motor Corporation и Nissan Motor Company Ltd, а также Jaguar Land Rover (последний входит в совет Linux Foundation).

Одновременно был представлен релиз AGL Unified Code Base — универсальной платформы для работы со всеми системами автомобилей. Новый дистрибутив AGL, основанный на наработках Yocto Project (совместного открытого проекта по разработке шаблонов, инструментов и методов для создания специализированных Linux под встраиваемые системы на базе различных аппаратных архитектур), включает:

- » полноценный дистрибутив Linux на основе Yocto Project;
- » типовой слой для автомобильных информационно-развлекательных систем (IVI) для проектов вроде AGL, GENIVI и др.;
- » открытую инфраструктуру разработки, в т.ч. репозитории Git, систему рецензирования кода Gerrit, систему отслеживания ошибок Jira;
- » систему непрерывной интеграции Jenkins;
- » автоматизированную инфраструктуру для тестирования;
- » оболочку Westin IVI shell с расширением Wayland IVI (от GENIVI);
- » поддержку Qt для мультимедиа и QML для приложений;
- » Demo-версии приложений Home Screen, Media Browser, HVAC Control and Display, AM/FM Radio and Navigation;
- » первый открытый драйвер для устройств MOST, от Microchip Technology;
- » поддержку нативных и HTML5-приложений.

» Дистрибутив AGL служит общей основой для создания различными производителями авто собственных конечных решений.



БЕЗОПАСНОСТЬ

# Новый шифровальщик для Linux

Создатели Linux.Encoder.3 проделали «работу над ошибками».

**В** начале наступившего года вирусописатели преподнесли пользователям Linux очередной «подарок» — новую версию троянца-шифровальщика для данной ОС. Исследованный специалистами компании «Доктор Веб» образец энкодера, получивший наименование Linux.Encoder.3, имеет несколько заметных отличий от своих предшественников.

Судя по всему, вирусописатели вняли советам одной западной антивирусной компании, подробно расписавшей в своих публикациях ошибки, допущенные авторами в коде Linux.Encoder.1, и оперативно устранили эти ошибки. Как и предыдущие версии Linux.Encoder, этот троянец проникает в домашнюю папку сайтов с использованием шелл-скрипта, который злоумышленники внедряют в различные системы управления контентом с использованием неуставленных уязвимостей. Linux.Encoder.3 не потребует для своей работы

привилегий суперпользователя Linux — троянец запускается с правами web-сервера, которых ему вполне достаточно для того, чтобы зашифровать все файлы в домашней директории сайта.

Вирусописатели изменили применяемый троянцем алгоритм шифрования, однако расширение зашифрованных файлов

## Вирусописатели вняли советам одной антивирусной компании.

оставлено прежним — **.encrypted**. Существенным отличием от предыдущих версий шифровальщика является то обстоятельство, что Linux.Encoder.3 способен запоминать дату создания и изменения исходного файла и для испорченных им файлов подставлять вместо них те значения,

что были установлены до шифрования. Каждый экземпляр вредоносной программы использует собственный уникальный ключ шифрования, создаваемый на основе характеристик шифруемых файлов и значений, сгенерированных случайным образом.

Ряд архитектурных особенностей Linux.Encoder.3 позволяет успешно расшифровывать файлы, поврежденные в результате действия данной вредоносной программы. Однако в связи с тем, что упоминавшаяся ранее антивирусная компания вновь опубликовала исследование троянца, содержащее подробную информацию о его «слабых местах», этими сведениями могут воспользоваться злоумышленники с целью модернизации шифровальщика. В ближайшее время с большой долей вероятности можно ожидать появления очередной версии Linux.Encoder, модифицированной с целью затруднить расшифровку поврежденных данной вредоносной программой файлов.

ХРАНИЕНИЕ ДАННЫХ

# Для локальных серверов

Появился HDD Seagate емкостью 8 ТБ для NAS-решений SMB.

**К**омпания Seagate объявила о выпуске нового HDD емкостью 8 ТБ, ориентированного на применение в системах NAS малых предприятий, частных облаках, хранилищах мультимедиа. Событие, как говорится в пресс-релизе, «отражает неизменную приверженность компании Seagate рынку SMB [среднего и малого бизнеса]». Новинка может являться основой для формирования восьмидисковой NAS-стойки общей емкостью 64 ТБ.

Заявленный максимум рабочей нагрузки составляет 180 ТБ/год — наибольший в этой категории дисков для NAS. Что касается стабильности, то минимальное время наработки на отказ — 1 млн часов. Каждый диск сопровождается трехлетней гарантией, а для предотвращения потери данных доступен фирменный сервис Rescue Data Recovery. Помимо этого, самого емкого диска в линейке HDD Seagate для NAS, выпускаются модели на 1, 2, 3, 4 и 6 ТБ. Дополнительно Seagate также предлагает широкий выбор настольных устройств NAS, работающих под управлением фирменной Linux-подобной операционной системы.

Два основных производителя NAS положительно оценили новинку. Дэвид Чейнг [David Chaing] (QNAP): «Новый HDD на 8 ТБ для NAS от Seagate предоставляет специалистам огромный потенциал для легкого управления растущим количеством данных». Чед Чейнг [Chad Chaing] (Synology): «Наши клиенты получили невероятную прежде возможность обрабатывать огромные объемы создаваемых и совместно используемых данных».

» У оптимизированных для работы в NAS HDD меньшая вибрация и пониженное энергопотребление.



## Максимум рабочей нагрузки — наибольший в этой категории.

Seagate приводит результаты своего исследования, которые объясняют выбор предприятиями малого и среднего бизнеса решений NAS среди таких альтернатив как SAN (storage area networks) или DAS (direct attached storage): в 18% причиной является цена, в 16% простота использования, в 10% — качество предлагаемого решения.

Устройство протестировано на резервирование и восстановление, работу в составе print- и file-серверов, хранилища мультимедиа, системы архивации, совместного доступа к файлам и виртуализации. Цена новинки — \$385, поставки начнутся в конце первого квартала. Производители NAS, включая ASUSTOR, QNAP, Synology и Thecus, квалифицировали диск на совместимость со своими продуктами. Напомним также, что летом прошлого года Seagate объявила о намерении ликвидировать свой облачный сервис Wulua.

АППАРАТНАЯ МИНИАТЮРИЗАЦИЯ

# Web-сервер на Linux — в кармане

Портативное устройство Ocean управляется Debian 8.1 Jessie.

**В**нешне устройство похоже на смартфон, но на самом-то деле это полнофункциональный web-сервер под управлением Linux, который пользователь может уместить в своем кармане (по размерам устройство примерно соответствует iPhone 6: 150×78×12 мм, масса 170 г). С самого начала разработки Ocean (<https://getocean.io>) упор делался на портативность и мобильность: интегрированная батарея позволяет запускать web-приложения и Bluetooth в местах, где возможность подключения сетевого питания отсутствует. А поскольку внутри этого малыша скрыта мощь полнофункциональной версии ОС GNU/Linux (по умолчанию это Debian 8.1 Jessie с ядром 3.4.105+, но пользователь имеет возможность заменить его любым другим дистрибутивом), Ocean способен решать вполне серьезные задачи:

- » создание и развертывание web-приложений с использованием таких платформ как Node.js или Ruby-on-Rails;
- » организация персонализированного маршрутизатора;
- » выполнение роли хаба в системе Интернета вещей (IoT hub);
- » прототип маяков iBeacon или Eddystone.

Кроме того, устройство может быть использовано в качестве портативного аккумулятора для iPhone и устройств на Android: энергии его батареи достаточно, чтобы 1,3 раза перезарядить, например, iPhone 6.

Вот технические характеристики Ocean:

- » двоядерный процессор ARM Cortex-A7 на 1 ГГц;
- » 1 ГБ DDR3 480 МГц ОЗУ;
- » поддержка карт micro-SD до 16 ГБ;
- » внутренний flash chip 4 ГБ;
- » порт USB 2.0/3.0;



» Web-сервер Ocean выполнен на SoC и управляется полноценным Linux.

- » 802.11 b/g/n Wi-Fi и Bluetooth 4.0+;
- » поддержка беспроводного зарядного устройства Qi;
- » батарея емкостью 4200 mAh, обеспечивающая непрерывную работу процессора в течение 2 дней.

Продажи начались в феврале по цене \$149. Позднее разработчики обещают выпустить модели с большей емкостью накопителей по цене \$180 и \$200.

ИСПРАВЛЕНИЕ ОШИБОК

# Oracle ставит рекорд

Январское обновление устраняет 248 уязвимостей.

Ежеквартальный выпуск пакета обновлений для продуктов Oracle в январе устранил рекордное количество уязвимостей — 248 (на 62% больше, чем в октябре 2015 г., когда было закрыто 154 уязвимости, и в 2,5 раза больше среднего показателя — 100), в более чем 50 линейках продуктов компании. Причем, как отмечает компания ERPScan, специализирующаяся на безопасности систем SAP и Oracle, приложение GoldenGate содержит 2 критических уязвимости, допускающих удаленное использование, а 3 уязвимости, связанные с Java SE, платформой Oracle для запуска приложений Java на серверах и рабочих станциях, по общей системе оценки уязвимости (CVSS Score 10) имеют максимальный уровень опасности.

Директор Oracle по безопасности ПО Эрик Морис [Eric Maurice] сказал, что исправления касаются популярных продуктов



По количеству исправленных уязвимостей январское обновление является рекордным не только для Oracle, но и для любого поставщика ПО.

компании, в т.ч. MySQL, OC Solaris и пакета E-Business suite — видимо, благодаря плодотворной деятельности исследователей из Google, ERPScan, HP Zero Day Initiative, Salesforce.com, Onapsis, да и самой Oracle. Особое внимание уделено бизнес-приложениям Oracle и SAP — пока атаки на такие системы особого распространения не получили, но ситуация может измениться в связи с ростом интереса хакеров к хранящимся там конфиденциальным данным.

ЭКОСИСТЕМА ТЕХНОЛОГИЙ

# Fuel «прикрыли зонтом»

Fuel признан частью OpenStack Big Tent, а компания Mirantis опередила всех по вкладу кода в OpenStack.

В конце 2015 г. проект Fuel — инструмент развертывания OpenStack-облака и последующего управления им — был официально признан частью экосистемы OpenStack и включен в список проектов Big Tent. Эта инициатива компании Mirantis быстро собрала большое сообщество разработчиков и получила признание пользователей. И те, и другие поняли, что простота этого инструмента существенно повышает «качество жизни» клиентов, а значит, стимулирует развитие бизнеса создателей собственных дистрибутивов OpenStack.

По объему кода Fuel на 50% больше чем Nova (контроллер) и имеет на 75% больше коммитов в месяц, чем Neutron (Networking-as-a-Service). Вот как прокомментировал новость о включении Fuel в Big Tent сооснователь и глава маркетинга компании Mirantis Борис Ренский: «Я надеюсь, что официальное признание Fuel частью экосистемы OpenStack привлечет новых контрибуторов и подтолкнет уже имеющих

членов сообщества приумножить усилия по развитию проекта, для того, чтобы пользователи платформы OpenStack могли получать большее удовлетворение от работы с облаком при минимуме усилий по развертыванию и конфигурированию».

## Внутренняя конкуренция пойдет на пользу экосистеме.

Big Tent (англ. «Шатер» или «Большой зонт») — подход к управлению проектами внутри экосистемы OpenStack, принятый на саммите в Ванкувере весной 2015 г. Это «презумпция полезности» новых инициатив. Если раньше считалось, что для каждой конкретной задачи в OpenStack может быть только одно правильное решение, то после принятия Big Tent сообщество согласилось с тем, что внутренняя конкуренция пойдет на пользу экосистеме. LXF

Новости короткой строкой

Уязвимость ядра Linux (CVE-2015-8660), проявляющаяся при использовании пространств имен, позволяет обычному локальному пользователю получить права root. Источник: [seclists.org](http://seclists.org)

С 30.11.2016 Mozilla закрывает распределенный сервис идентификации Persona: отключат сайт [persona.org](http://persona.org), прекратят обработку запросов идентификации, удалят данные пользователей. Источник: [mail.mozilla.org](http://mail.mozilla.org)

Из новинок ядра Linux 4.4: Direct I/O и AIO для примонтированных в loop-режиме ФС, сбоеустойчивый RAID5 с добавочным журналированием, драйвер для SSD-накопителей Open-Channel на основе памяти NVM, поддержка 3D в виртуальном GPU virtio-gpu, KMS-драйвер для Raspberry Pi. Источник: [lkm1.org](http://lkm1.org)

Весной в продажу поступит телевизор Panasonic DX900 UHD TV на Firefox OS с диагональю экрана 58" и 65". Источник: [blog.mozilla.org](http://blog.mozilla.org)

Уязвимости в движке WebKit и в драйвере уровня ядра GPU позволили запустить на PlayStation 4 дистрибутив Linux вместо штатной Orbis OS. Источник: [venturebeat.com](http://venturebeat.com)

Проект Zcash развивает новую криптовалюту на технологиях Bitcoin. Протокол zerocash гарантирует для транзакций невозможность узнать отправителя, получателя и сумму перевода. Источник: [z.cash/blog/helloworld.html](http://z.cash/blog/helloworld.html)

Компания BQ Mobile (Испания), производитель смартфона BQ Aquaris с Ubuntu Touch, выпустит первый официально поставляемый с Ubuntu планшет. Источник: [www.omgubuntu.co.uk](http://www.omgubuntu.co.uk)

Из-за уязвимости в подсистеме keyrings (ключи аутентификации и сертификаты для шифрования) ядра Linux версий 3.8–4.5 обычный локальный пользователь может получить права root. Источник: [openwall.com](http://openwall.com)

Герман Грэф объявил о замене неконкурентоспособной новой IT-инфраструктуры Сбербанка на открытую платформу от компании GridGain. Источник: [www.rbc.ru](http://www.rbc.ru)

Ubuntu стала платформой облачных окружений и серверных и корпоративных приложений компании AT&T. Источник: [insights.ubuntu.com](http://insights.ubuntu.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 бактериологической 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».



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

## Ubuntu и ZFS: взгляд в будущее

О несравненных достоинствах файловой системы ZFS было написано много — в том числе и на страницах **LXF** (**LXF164**, **165/166**, **167**). Говорилось и о главном ее достоинстве — а оно в том, что ZFS файловой системой не является. Ибо объединяет в себе функции как ФС, так и системы управления томами. Что ныне не столь уж уникально — сходная функциональность имеется и в Btrfs, и в Hammer. Однако первую по сей день до ума так и не довели, а вторая поддерживается только родной операционкой, то есть DragonFly BSD. ZFS же, во-первых, прошла испытание временем и, во-вторых, тем или иным образом поддерживается везде, в том числе и в Linux'е во всем его многообразии.

Так-то оно так, да не совсем так. Потому что, несмотря на все достижения проекта ZFS on Linux, поддержка первой во всех дистрибутивах второго выполняется через устройство, родительское для **/dev/feet**. И причины тому не технические, а чисто юридические, то есть несовместимость лицензий. Так что, казалось бы, путей разрешения этого классового противоречия нет.

Тем интересней выглядит заявление Марка Шаттлворта, что в Snappy Ubuntu 16.04 ZFS поддерживаться будет. Как — пока не ясно. Но, поскольку Snappy, по определению, система атомарная, можно предполагать, что поддержка эта будет «из коробки». Что, очевидно, противоречит лицензиям GPL и CDDL. Как будет решено это противоречие — пока можно только гадать. Благо, гадать остается не очень долго.  
[alv@posix.ru](mailto:alv@posix.ru)

## Сегодня мы рассматриваем:

**openSUSE 42.1** ..... 12  
Маститый дистрибутив совершил большой скачок и сменил ориентацию: отныне он переходит к возобновляемым релизам и отказывается от Live-версии. К тому же это первый релиз с KDE Plasma 5.

**BakAndImgCD 14.0** .... 13  
Официальная ветка минималистского дистрибутива 4MLinux предназначена для максимального облегчения операций резервного копирования и создания образов и клонов системы. Удалось ли переиграть Clonezilla?

**RockStor 3.8.7** ..... 14  
Впечатляющий дистрибутив для сетевого хранилища, который внедрил файловую систему Btrfs. А если

вы снабдите его своей электронной почтой, он будет предупреждать вас об определенной активности в его поле.

**MotionEyeOS** ..... 15  
Программная реализация всепроникающего глаза Саурана — организуем профессиональное видеонаблюдение, управляя множеством камер Raspberry Pi. Это проект одиночки, но для начала вам документации хватит.

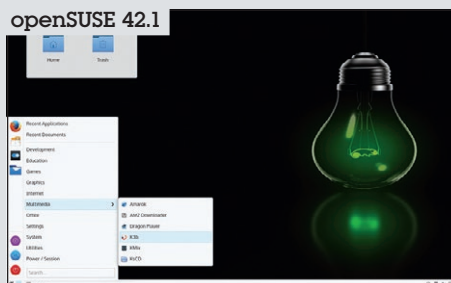
**MySQL Router 2.0.2** ..... 16  
Ферме серверов требуется пастух... Этот продукт промежуточного звена обеспечит прозрачную и эффективную маршрутизацию трафика между приложением и серверами MySQL.



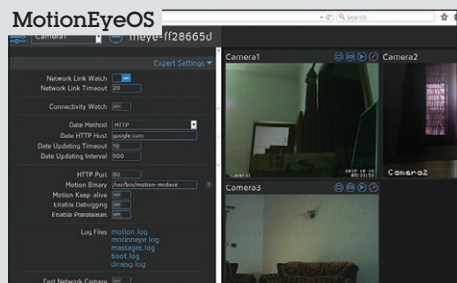
➤ Вот такой он, Raspberry Pi Zero.

Расширяемая архитектура легко подстраивается под любые нужды.

**Raspberry Pi Zero** ..... 18  
«Небываемое бывает»: в Raspberry Pi Foundation разработали одноплатный компьютер ценой всего \$5! Этот прорыв проложит дорогу к образованию массам детей и превратит в экономически обоснованные массу проектов.



➤ Традиционный зеленый цвет — в новой ипостаси: свечение плазмы?



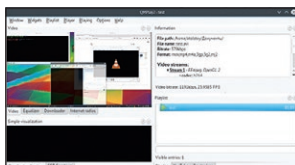
➤ Вы можете присматривать за своими детьми, кошками, собаками и визитерами...

## Сравнение: Видеоплееры с. 22

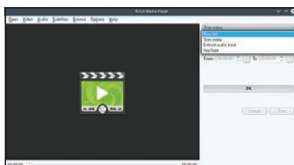
### Bomi



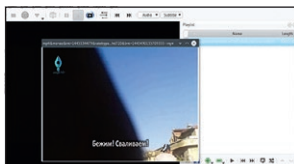
### QMplay2



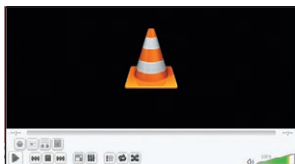
### Romp



### SMPlayer



### VLC



Если чего в Linux имеется великое множество, так это медиа-плееров. Чтобы облегчить вам проблему выбора, мы стравили пяток самых лучших. Результат получился предсказуемый...

# openSUSE 42.1

Дистрибутив на базе SUSE Linux Enterprise совершает прыжок вперед, а неустрашимый **Шашанк Шарма** выясняет, геккон это или хамелеон.

## Вкратце

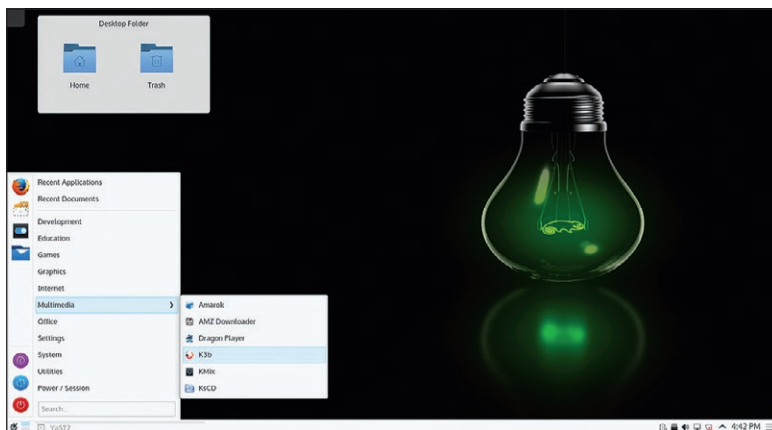
» Один из ведущих настольных дистрибутивов на базе RPM. Его поддержкой занимается международная корпорация, использующая openSUSE в качестве испытательной площадки для версии Enterprise. С этого релиза, openSUSE прекращает 9-месячный цикл обновления и будет опираться на релизы SLE. См. также: Fedora и Mageia.

**П**роjekt openSUSE встал на путь обобщения и систематизации того, что предложил в прошлом году, объединив новейший релиз Tumbleweed и экспериментальную ветку Factory для создания непрерывно обновляющейся кодовой базы для нового дистрибутива. Дополняет этот релиз новая серия стабильных релизов под названием Leap [англ. скачок].

Основой Leap является исходный код SUSE Linux Enterprise (SLE), и все исправления ошибок и обновления безопасности дистрибутив будет получать из релизов SLE. Согласно описанию релиза, первая версия, 42.1, выполнена на базе первого пакета обновления SLE12. Забегая вперед, Leap 42.2 будет основываться на SP2, а 42.3 на SP3.

Главной целью нового дистрибутива является стабильность. Для этого в Leap используются только тщательно протестированные компоненты, в которых уверены разработчики SUSE, обслуживающие клиентов Enterprise. В этом смысле openSUSE Leap для SUSE — то же, что и производный от RHEL CentOS для Red Hat.

Другая существенная переменная в связи с переходом на пакеты SLE касается цикла релизов. Помимо стабильной поддержки, Leap теперь перенимает и цикл обновления SLE. Планируется синхронизировать выпуск новых основных версий с выходом релизов и пакетов SLE. Согласно примечаниям к релизу, ожидается, что пользователям Leap должны будут обновляться до последней промежуточной версии в течение шести месяцев со дня ее выхода. Это гарантирует им получение обновлений программ и безопасности в течение 18 месяцев для



» Номер версии Leap — еще один отклик на «Главный вопрос жизни, Вселенной и всего такого» из *Автостоп по Галактике*.

каждого из таких релизов. Основной релиз, такой как Leap 42, будет поддерживаться не менее 36 месяцев, так что уже не имеет смысла использовать openSUSE Evergreen.

Поскольку Leap ориентирован на стабильность, в составе дистрибутива проверенные пакеты, на одну-две версии старше последнего релиза.

## Большой скачок

По умолчанию дистрибутив использует файловую систему Btrfs для корневого раздела и XFS для каталога **home**. Однако лучше, если это будет целиком Btrfs, поскольку тогда вы сможете применять инструмент *Snapper* для управления моментальными снимками [snapshot] файловой системы. Помимо ежечасного выполнения этой операции по умолчанию, *Snapper* также создаст снимки до и после внесения изменений в систему, используя *YaST* или менеджер пакетов. В Leap 42.1 можно загрузиться прямо в снимок.

*Snapper* был интегрирован в *YaST*, главный инструмент настройки дистрибутива, чтобы создавать снимки на уровне файловой системы. Кроме того, *YaST* содержит некоторые новые модули, такие как *YaST Docker* для управления демоном *Docker* и управления контейнерами.

В соответствии со своим стремлением к стабильности, релиз основывается на ядре Linux 4.1. LTS. Вдобавок это первый стабильный релиз openSUSE с использованием KDE Plasma 5 (5.4.2, для пущей точности). Приложения в составе дистрибутива взяты из наборов KDE Applications 15.08 и 15.04. Помимо этого, в дистрибутиве есть

файловый менеджер *Dolphin* на базе версии Frameworks 5. Команда разработчиков KDE из openSUSE до сих пор работает над планом внедрения обновленного релиза KDE Applications в жизненный цикл Leap 42.1.

Еще одним отходом от традиции стала доступность openSUSE Leap только в виде устанавливаемых DVD для 64-битных архитектур. Разработчики сочли, что устанавливаемые Live CD не вполне раскрывают возможности установщика *YaST*.

openSUSE Leap — интересное пополнение ассортимента дистрибутивов. Он не столь дряхлый, как Debian Stable, обещает больше стабильности, чем обычные настольные дистрибутивы, и гораздо совершеннее, чем CentOS. Поэтому он подходит как для любого ПК, так и даже для сервера. **LXF**

## Свойства навскидку

**Надежная база**  
Основывается на стабильных и проверенных открытых компонентах из релиза SLE.

**Установка с DVD**  
В отличие от других дистрибутивов, openSUSE доступен только как установочный DVD для 64-битных машин.

## LINUX FORMAT Вердикт

**openSUSE 42.1**  
**Разработчик:** openSUSE Project  
**Сайт:** GPL и другие  
**Лицензия:** Разные

<b>Функциональность</b>	9/10
<b>Производительность</b>	9/10
<b>Удобство в работе</b>	8/10
<b>Документация</b>	8/10

» Отлично продуманный и безупречно сделанный дистрибутив — обязателен для пользователей openSUSE.

## Рейтинг 9/10

# BackAndImgCD 14.0

Хорошую лысину шапка красит, вот и **Шашанк Шарма** напялил свой колпак сисадмина и взялся за BackAndImgCD.

## Вкратце

Основанный на 4MLinux, BackAndImgCD является специальным дистрибутивом для бэкапа файлов, клонирования дисков и разделов. См. также: Clonezilla Live CD, Redo Backup и System-RescueCD.

Создание резервных копий, запись образов и разметка диска принадлежат к числу тех админских задач, с которыми должны спокойно справляться даже рядовые пользователи ПК. Однако, в отличие от стандартного резервирования файлов и папок, выполнение тех же процедур на уровне системы требует специальных инструментов и умений, о чем люди часто забывают. Дистрибутив BackAndImgCD, официальное ответвление минималистичного 4MLinux, предназначен для упрощения процессов резервного копирования, создания образов и клонирования файлов, разделов и дисков.

Дистрибутив доступен в виде ISO-образа объемом всего 28 МВ и без лишних проволочек предоставляет доступ к популярным открытым инструментам и сценариям клонирования, таким как *Partclone*, *Partimage* и *ddrescue*. Благодаря этим инструментам поддерживается широкий спектр файловых систем, в том числе таких, что часто используются в открытых и проприетарных ПК и серверах — ext3, ext4, ReiserFS, Btrfs, HFS+, FAT, NTFS и т.д.

Благодаря текстовому интерфейсу, дистрибутив быстро берется за дело. При загрузке вам предлагается выбрать одну из трех поддерживаемых опций. Первая — резервное копирование вручную, при котором жесткий диск сканируется, позволяя быстро скопировать файлы на съемные USB-диски, оптические приводы или удаленный FTP/SFTP сервер. Собрав информацию о том, какие файлы и куда следует скопировать, дистрибутив запускает многофункциональную утилиту *Midnight Commander* для резервного копирования.

```
Welcome to BakAndImgCD !

The most important commands are:
'backup' (manual backup of files),
'fsbackup' (automatic backup of files),
'image' (creation of disk images).

Backup files (and images) can be compressed and sent to a remote FTP server.
By default BackAndImgCD uses dynamic IP for the 'eth0' interface to connect
to the Internet. Run the 'netconfig' script if you need to reconfigure these
default settings.
There are the following two commands to (re)start the Internet connection:
'connect' for Ethernet (including WiFi),
'modem' for Dial-up (including fast USB modems).

The 'cfdisk' utility may help you to view/manage available MBR partitions,
while the 'cgdisk' tool should be used if you have GPT partitions.
You can execute 'mc' to open Midnight Commander if, for any reason, you want
to use a file manager. Run 'links' command if you need a simple web browser.
The 'rsync' tool is also included.

Good luck :-)
```

Дистрибутив BackAndImgCD в плане интерфейса неказист, зато проявляет достаточную гибкость, чтобы встроиться в вашу сеть.

Вторая опция — автоматическое резервирование. В отличие от предыдущей, она применяется не к отдельным файлам, а к целым разделам. Для начала отображается таблица разделов, и вас спрашивают, который из них вы хотите копировать. (Было бы неплохо, если бы это можно было делать и по нескольку за раз.) Когда вы укажете раздел, вам предложат выбрать тип архива, перечислив достоинства каждого из них. Можно сжать содержимое раздела в TAR.GZ, это быстрее всего, а можно медленнее, но эффективнее; в TAR.BZ.2, или TAR.XZ, это будет дольше всего.

## Быстрый, но скучный

Как уже отмечалось, в BackAndImgCD нет ни диалоговых окон, сообщающих о ходе выполнения, ни примерного времени ожидания. По завершении копирования появляется опция отправить на удаленный FTP-сервер. В этом случае устанавливается соединение с удаленным компьютером и управление вновь передается *Midnight Commander* для передачи данных.

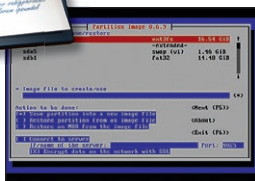
Третья и последняя задача, которую вы можете выполнить с помощью дистрибутива — это создание образов дисков. Для начала вам предлагается выбрать один из трех доступных инструментов клонирования: *partimage*, *partclone* или *ddrescue*. Есть и четвертая опция, специально для новичков, уместно названная «Понятия не имею [I have no idea]!»: при ее выборе запускается инструмент *partclone*. *Partclone* может клонировать разделы/диски только если место назначения по объему равно источнику или больше него; в противном случае процесс завершится, и вам предложат

обратиться к файлу журнала программы для решения проблем.

Можете написать это на излишнюю осведомленность или предвзятость, но нам не показалось, что BackAndImgCD настолько хорош, чтобы сманить нас с Clonezilla. Помимо вариантов с локальным диском и USB, нам очень не хватило возможности клонирования разделов напрямую в удаленное расположение, например, при работе с образами дисков, тогда как Clonezilla позволяет смонтировать удаленное расположение (через *Samba*, *NFS*, *WebDAV*) в локальной файловой системе и сохранить образ прямо туда. Аналогично, при клонировании дисков целиком, Clonezilla умеет копировать весь диск или отдельные разделы по сети. А BackAndImgCD перемещает образ на удаленный ресурс только после сохранения копии на локально доступном диске или носителе. **LXF**

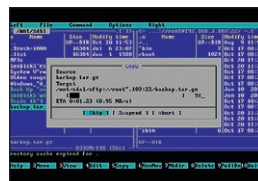


## Свойства навскидку



### Все по делу

Благодаря простому интерфейсу, с помощью этого дистрибутива можно быстро клонировать диски и разделы на большинстве систем.



### Midnight Commander

Может, на вид он неказист, но в mc есть практически все те опции, что и в любом современном файловом менеджере.

## LINUX FORMAT Вердикт

### BackAndImgCD 14.0

Разработчик: 4MLinux  
Сайт: <http://bakandimgcd.4mlinux.com>  
Лицензия: GNU GPL v3

Функциональность	6/10
Производительность	7/10
Удобство в работе	5/10
Документация	5/10

Инструмент для резервирования и клонирования без излишества, для тех, кто знает, что делает.

## Рейтинг 6/10

# RockStor 3.8.7

В нескончаемом поиске новых способов эффективного управления данными, Маяк Шарма тестирует NAS на базе Linux.

## Вкратце

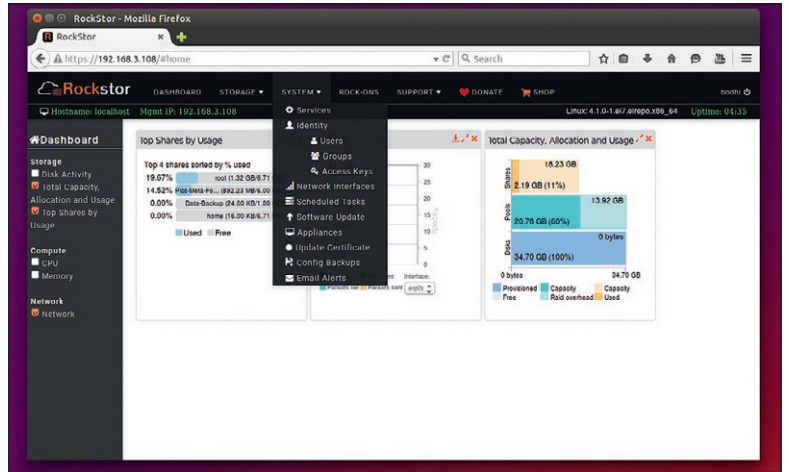
» Дистрибутив, превращающий запасной компьютер в сетевое хранилище — Network Attached Storage (NAS). Может связать имеющиеся диски в RAID разного уровня и управляется через web-интерфейс. См. также: FreeNAS и OpenMediaVault.

**Х**отя в основе FreeNAS лежит FreeBSD, этот дистрибутив давно полюбился пользователям Linux. Однако OpenMediaVault на базе Debian будет достойной альтернативой для тех, кому надоело ломать голову над сложностями FreeNAS; вот и RockStor — еще один дистрибутив с NAS-сервером на Linux — также обещает полную функциональность FreeNAS в удобном для использования и управления пакете.

Лучшее в этом дистрибутиве на основе CentOS 7 — фокусирование на единой файловой системе, Btrfs. Это упрощает процесс создания хранилища и обеспечивает однородность свойств файловой системы во всех элементах интерфейса. Дистрибутив поддерживает несколько профилей RAID, обеспечивая избыточность данных. Кроме того, можно менять размер пула памяти, добавляя или удаляя диски, и даже изменить его профиль RAID без потери данных и нарушения доступа.

При добавлении диска в пул дистрибутив автоматически запускает процесс балансировки Btrfs, чтобы равномерно распределить данные по всем дискам пула. Для обеспечения согласованности данных дистрибутив также позволяет инициировать операцию чистки Btrfs, которая считывает данные дисков в пуле, проверяет их контрольные суммы и исправляет искажения. Можно также задать расписание для выполнения этой операции.

Отдельные диски в пуле являются подтомами Btrfs и доступны через любой из популярных протоколов, включая Samba/



» Имея несколько установок RockStor, легко дублировать данные из одной на другую.

CIFS, NFS, AFP и SFTP. Кроме того, легко менять размер, удалять или клонировать хранилище.

## Рок-н-ролл

Еще одной интересной функцией Btrfs являются моментальные снимки [snapshot], то есть создание копии хранилища. Снимок не дублирует данные, а хранит только ссылки на исходное содержимое хранилища. Он занимает место только тогда, когда вы удаляете файл из хранилища, а в снимке он продолжает существовать. Возможен откат к одному из предыдущих снимков хранилища.

Базовая функциональность RockStor способна расширяться за счет применения плагинов, называемых Rock-ons. Они выполнены в виде контейнеров Docker и находятся в активной разработке, но в текущей версии уже есть плагины для BTSync, медиа-сервера Plex, ownCloud, OpenVPN и т.д.

Управлять NAS можно через сетевой интерфейс, над которым разработчики RockStor изрядно потрудились, чтобы сделать его достаточно интуитивным. Как мы обнаружили, и раздел управления, и экраны снабжены полезными всплывающими подсказками, тогда как опытные пользователи оценят функции контроля доступа, а также возможность сжатия данных с применением алгоритмов zlib или lzо. Помимо этого, RockStor можно настроить на работу со службами каталога Active Directory, NIS или LDAP.

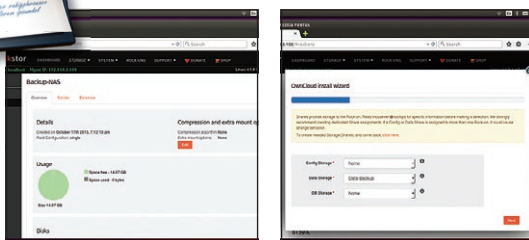
Если разрешить доступ RockStor к учетной записи электронной почты, можно

настроить дистрибутив на отправку уведомлений об определенных действиях. По завершении настройки RockStor NAS можно с помощью встроенной функции создать резервную копию текущей конфигурации, чтобы использовать ее на другой системе.

Сервер выпускает стабильные обновления приблизительно раз или два в месяц, и вы можете обновиться через интерфейс управления с помощью одного щелчка. Впечатляющий список функций подкрепляется подробной документацией с исчерпывающими описанием каждой. Справочники хорошо иллюстрированы, дабы помочь начинающим пользователям установить RockStor без особых усилий. Кроме того, на сайте проекта есть онлайн-демо-версия. Так что пробуйте. **LXF**



## Свойства навскидку



### Лучшее от Btrfs

Все самые продвинутые функции файловой системы Btrfs, такие как управление томами, моментальные снимки и очистка системы.

### Дополнения

Расширьте возможности NAS через плагины, например, для доступа с разных носителей и добавления личного облачного сервера.

## LINUX FORMAT Вердикт

### RockStor 3.8.7

**Разработчик:** RockStor Inc.  
**Сайт:** <http://rockstor.com>  
**Лицензия:** GPLv2

<b>Функциональность</b>	8/10
<b>Производительность</b>	8/10
<b>Удобство в работе</b>	8/10
<b>Документация</b>	8/10

» Прекрасное и удобное NAS, взявшее лучшее от файловой системы Btrfs ради создания впечатляющего дистрибутива.

## Рейтинг 8/10



# MotionEyeOS

Видеонаблюдение не столь эффективно и всепроникающе, как всевидящее око Саурана, но есть ПО, очень близкое к этому оку, обнаружил **Маянк Шарма**.

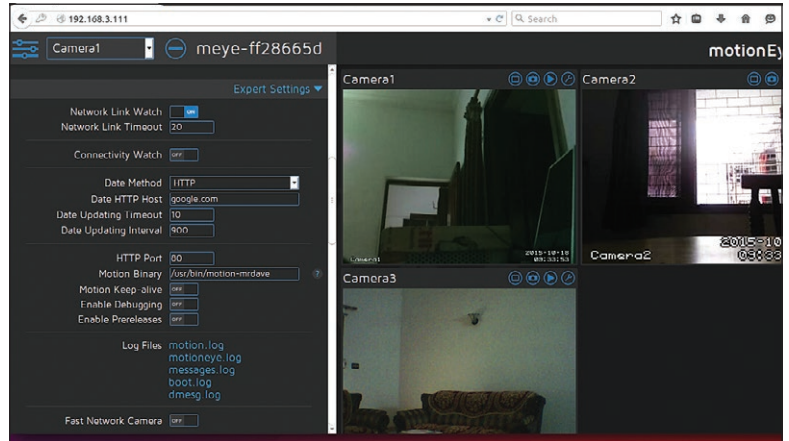
## Вкратце

» Существуют различные варианты использования Pi как системы обнаружения движения, в том числе почтенный CLI-инструмент *motion*. В прошлом этот инструмент был обернут скриптами от пользователей, которые захватывают и отправляют электронную почту, но наш маленький дистрибутив обещает сделать все это за вас.

Помимо образования, Raspberry Pi широко используется в проектах, требующих энергоэффективности и постоянной готовности устройства. Один из таких случаев — система видеонаблюдения с обнаружением движения, и MotionEyeOS — грамотный маленький дистрибутив для этой задачи. Он создан специально для одноплатных компьютеров и работает как на оригинале Pi и его обновлениях, так и на Banana Pi, Cubietruck и Odroid C1.

Мы тестировали последнюю версию MotionEyeOS на Raspberry Pi 2; наша тестовая система видеонаблюдения состояла из трех камер — официальной камеры Raspberry Pi, дешевой USB-камеры и сетевой камеры, управляемой приложением *IP Webcam* на старом Android-смартфоне. Хотя для переноса образа MotionEyeOS на карту microSD сойдет и dd, мы бы предпочли использовать скрипт установки, созданный разработчиком дистрибутива. С помощью этого скрипта можно передать SSID и пароль для беспроводной сети (и, возможно, статический IP-адрес) перед резервным копированием образа на карту памяти. Можно вставить подготовленную карту памяти в Pi без монитора, и дистрибутив автоматически запустит SSH сервер, так что вы сможете войти удаленно, получив доступ к командной строке. Впрочем, виртуально управлять всеми аспектами системы можно с удаленного браузерного интерфейса администратора MotionEyeOS, который работает через порт 80.

Интерфейс администратора разбит на несколько разделов. По умолчанию он



» Интерфейс усеян подсказками, которые четко объясняют различные опции.

отображает простые функции управления подключенными устройствами. Можно, например, включить такие функции, как потоковое вещание видео, фотосъемки или запись видео при обнаружении движения, с заданием срока хранения видео. Можно также определить график работы камеры: например, активировать ее и обнаружение движения в период с 22:30 до 06:00 по будням и круглосуточно по выходным.

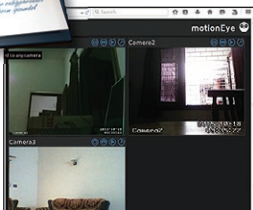
Помимо элементов управления, интерфейс отображает потоки от всех камер. Когда камера обнаруживает движение, рамка ее изображения становится красной, камера начинает фотографировать и записывать видео. Записанные видео и кадры можно просматривать и загружать по кнопкам в отдельных окнах. Замечательная особенность — преобразование фотографий в замедленное видео. Собранный материал также виден в локальной сети и может быть доступен через предварительно настроенный сервер *Samba*, запущенный на дистрибутиве.

HTTP. Правда, этот движок поддерживает только официальные камеры Pi. Кроме того, он не обучен распознавать движение, то есть подходит только для потокового видео со значительно более высокой частотой кадров и высоким разрешением без надсады для CPU Pi.

Другой дополнительный параметр, который следует отметить, это возможность при обнаружении движения отправлять уведомления по электронной почте, в том числе отснятые кадры в виде вложения. Как и все прочие параметры, этот можно индивидуально настроить для каждой из подключенных камер.

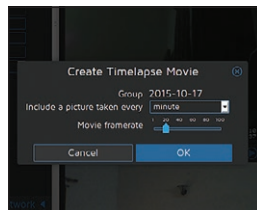
Не забывайте, что данный проект — пока что шоу одного актера; но он предоставляет достаточно документации для начала работы, даже если в нем и отсутствуют традиционные средства поддержки. **LXF**

## Свойства навскидку



### Поддержка камеры

Работает с большинством камер и даже с камерами, подключенными к другому экземпляру MotionEyeOS в сети.



### Интуитивный интерфейс

Простое управление для новичков и расширенные настройки, дающие экспертам больше контроля над системой.

## Расширенные возможности

Кроме конфигурирования самого дистрибутива, можно также разогнать Raspberry Pi, придав ему силы справляться с возросшей нагрузкой управления несколькими камерами. По умолчанию для захвата и потоковой передачи видеокладов дистрибутив использует демон движения, но не пользуется преимуществами GPU Pi. Вместо этого можно использовать движок *streamEye*, который может переложить обработку на GPU Pi и направить поток видео как MJPEG через

## LINUX FORMAT Вердикт

### MotionEyeOS 20151011

Разработчик: Calin Crisan  
Сайт: <https://github.com/ccrisan/>  
Лицензия: GNU GPLv3

Функциональность	8/10
Быстродействие	7/10
Удобство в работе	7/10
Документация	6/10

» Отличный дистрибутив для превращения Pi и других одноплатных компьютеров в специальную систему видеонаблюдения.

## Рейтинг 7/10

# MySQL Router 2.0.2

Лада Шерышова представляет один из структурных элементов решений высокой доступности, обеспечивающий связь между приложениями и серверами MySQL.

## Вкратце

» Используя *MySQL Router*, создавайте более надежные приложения, а также стройте высокодоступные решения в связке с *Fabric*.

**М**ysql Router — продукт промежуточного звена, обеспечивающий прозрачную и эффективную маршрутизацию между приложением и серверами MySQL. Router предоставляет широкий спектр вариантов использования, включающий реализацию высокой доступности (ВД) и масштабирования за счет эффективной маршрутизации трафика к целевым серверам MySQL.

Как правило, высокодоступная ферма серверов MySQL состоит из одного главного сервера и нескольких подчиненных. Задача приложения — поддерживать бесперебойность работы и отказоустойчивость в случае, если вдруг главный сервер выйдет из строя. С *MySQL Router* маршрутизация подключений приложения к серверам MySQL станет абсолютно прозрачной. Также Router позволяет получить дополнительную масштабируемость и производительность за счет балансировки нагрузки, распределяя подключения в циклическом [карусельном, round-robin] режиме.

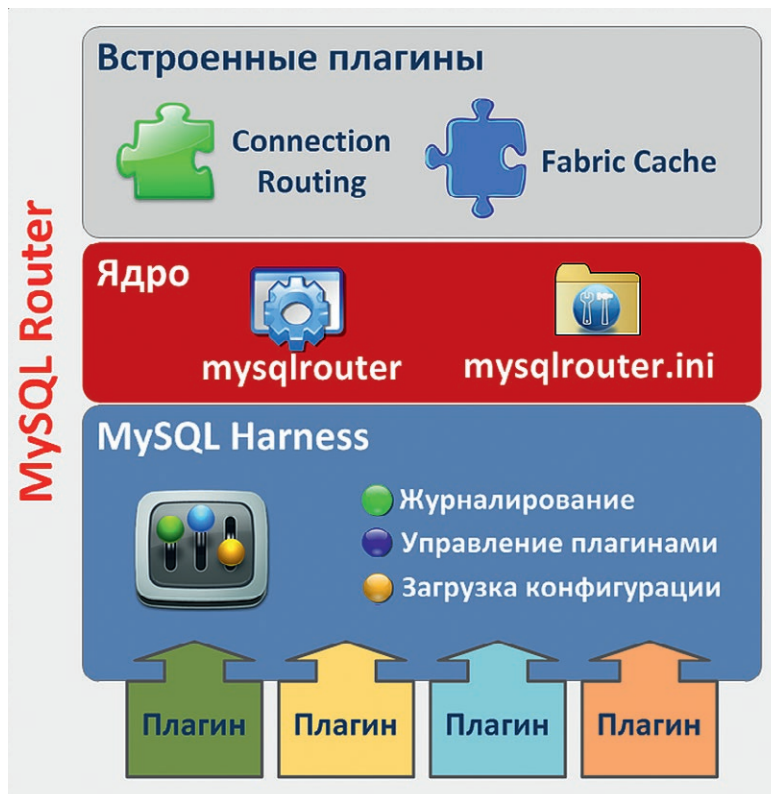
Приятной особенностью является еще и то, что архитектура продукта обладает расширяемостью через плагины, позволяя разработчикам легко подстраивать Router под свои нужды и тем самым получать безграничные функциональные возможности. Кроме того, продукт предлагает бесшовную интеграцию с *MySQL Fabric* (см. стр. 14 LXF204), предоставляя последнему возможность управлять высокодоступными группами серверов на основе интеллектуальной маршрутизации, что упрощает управление группами в рамках выполнения задач резервирования и непрерывного функционирования.

*MySQL Router* — это структурный элемент для решений высокой доступности. Продукт предоставляет 3 основные возможности: простую маршрутизацию соединений, маршрутизацию соединений, интегрированную с *Fabric*, и поддержку плагинов посредством MySQL Harness.

Рассмотрим возможности *MySQL Router* подробнее.

## Архитектура на плагилах

*MySQL Router* — это простое приложение, состоящее из ядра, встроенных плагинов и узла подключения дополнительной функциональности посредством MySQL Harness. Вся архитектура Router имеет модульную структуру — основные и дополнительные



» Архитектура *MySQL Router*: модульная и расширяемая.

функции *MySQL Router* загружаются как плагины.

Ядро *MySQL Router* — это отдельный исполняемый файл **mysqlrouter**, который запускается на той же платформе, что и приложение. Его можно запустить в фоновом режиме, и, включив журналирование, выводить логи в файл. Аналогично MySQL-серверу, Router имеет конфигурационный файл (**mysqlrouter.ini**), хранящий информацию о том, как выполнять маршрутизацию. Этот файл может содержать несколько разделов, позволяя нескольким приложениям использовать один и тот же экземпляр Router. Приложения подключаются к Router через хост и порт, указанные в конфигурационном файле.

В текущей версии *MySQL Router* поставляется со следующим набором встроенных плагинов:

- » **Плагин Connection Routing** Выполняет простую маршрутизацию подключений.
- » **Плагин Fabric cache** Осуществляет прозрачную маршрутизацию подключений на основе информации, полученной от *MySQL Fabric*.

## Маршрутизация подключений

Встроенный в *MySQL Router* плагин **Connection Routing** — маршрутизатор подключений — предоставляет автономную маршрутизацию, разрешая выполнять перенаправление соединения, отправленного через Router к доступному серверу MySQL. Маршрутизация подключения — это попросту перенаправление пакетов MySQL, но без какой-либо проверки и анализа. Если текущее соединение от Router к серверу MySQL прерывается (например, при отключении сервера), приложение выдает ошибку подключения и инициирует новый запрос на подключение. Выбрав из списка другой сервер, Router просто перенаправит соединение. Такая стратегия маршрутизации довольно проста и имеет максимальную пропускную способность. Отметим, что списки серверов для подключения в данном случае формируются вручную на уровне конфигурационного файла Router.

Ниже приведен пример соответствующего раздела конфигурационного файла: `[routing:simple_redirect]`

```
bind_port = 7002
mode = read-write
destinations = localhost:3310,localhost:3311,localhost:3312
```

Такая конфигурация предписывает маршрутизатору прослушивать порт 7002 на локальной машине и перенаправлять подключения к одному из серверов, указанному в списке в опции destinations, устанавливая режим подключения read-write, заданный параметром mode.

## Интеграция с Fabric

Интеграция *MySQL Fabric* с *Router* обеспечивает другую форму маршрутизации соединения, где вместо ручной настройки подключений используется список серверов, предоставляемый *MySQL Fabric*. Напомним, что *Fabric* позволяет управлять фермой MySQL-серверов, объединяя их в высоко-доступные группы для обеспечения резервирования (позволяя автоматически переходить на другой ресурс в случае отказов), масштабирования и облегчения управления. *MySQL Fabric* поддерживает список серверов в ферме, включая их роли и конфигурационную информацию. *Fabric* передает эту информацию маршрутизатору через специальный кэш.

В этом случае также потребуются внешние изменения в конфигурационный файл *Router*, но только для настройки самого плагина:

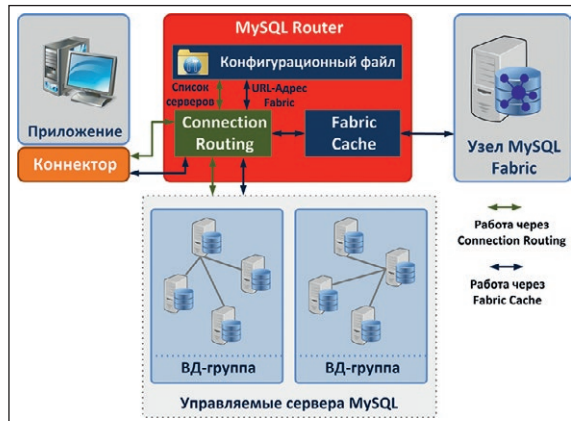
```
[fabric_cache:webfarm]
address = fabric.example.com
user = fabricuser

[routing:webfarm]
bind_port = 19906
destinations = fabric+cache://webfarm/group/homepage/
mode = read-only
```

В первом разделе устанавливаются адрес и пользователь для подключения к *Fabric*. Второй раздел определяет стратегию маршрутизации, которая предписывает *Router* прослушивать порт 19906 на локальной машине, а список серверов для подключения брать из кэша *Fabric* по указанному URI.

## Плагины и MySQL Harness

Как уже говорилось, *MySQL Router* имеет модульную архитектуру, которая позволяет без труда подключать плагины для расширения функциональности. Здесь приходит на помощь технология плагинов *MySQL Harness*. *MySQL Harness* предоставляет возможность добавлять новые функции без внесения изменений в исходное приложение *mysqlrouter*, что значительно ускоряет разработку. Также *Harness* обеспечивает журналирование и позволяет включать логи событий, тем самым



» **MySQL Router** служит посредником между приложением и фермой серверов.

избавляя разработчиков от необходимости написания какого-либо специализированного кода.

Основные возможности *MySQL Harness*:

- » Отслеживание зависимостей между плагинами.
- » Поддержка версионирования.
- » Загрузка, запуск и остановка плагинов.
- » Разрешение на создание сторонних плагинов.
- » Загрузка и чтение конфигурационных файлов.
- » Поддержка платформо-независимых файлов.
- » Надежное ведение логов.

## Как работает MySQL Router

Клиентское приложение через коннектор подключается к *MySQL Router* (например, через порт 8500). Используя выбранную стратегию маршрутизации, *Router* проверяет, какой сервер *MySQL* доступен, и выполняет соединение.

При использовании простой маршрутизации подключений плагин *Connection Routing* считывает направления из конфигурационного файла и перенаправляет соединение на один из серверов в списке. При использовании стратегии маршрутизации на основе плагина *Fabric cache Router* сначала определяет URL-адрес, по которому установлен *Fabric*, затем получает список направлений, но уже не из конфигурационного файла *Router*, а от *Fabric*, и далее передает его маршрутизатору подключений для перенаправления соединения на один из серверов в этом списке.

Рекомендации Oracle по использованию продукта:

- » Устанавливайте и запускайте *MySQL Router* на той же машине, что и приложение.
- » Привязывайте *Router* к localhost, используя `bind_port = 127.0.0.1:<порт>` в конфигурационном файле.
- » *MySQL Router* лучше всего работает в связке с *MySQL Fabric*, хотя может использоваться и без него.

## Ограничения

При работе с *MySQL Router* надо учитывать следующее.

» В настоящее время приложение *mysqlrouter* не снабжено функциями высокой доступности, такими как мониторинг статуса *Router* и его автоматический перезапуск в случае возникновения ошибок. Вы должны дописывать их сами, например, используя скрипт.

» *Router* не анализирует пакеты, а просто перенаправляет соединение.

» *Router* нельзя привязать к нескольким IP-адресам. Параметр `bind_address` в конфигурационном файле допускает только один адрес. Однако можно привязать все порты на локальном компьютере (localhost), используя `bind_address=0.0.0.0`.

## Интеграция с приложениями

Разработка приложений совместно с *MySQL Router* не требует каких-либо специальных библиотек или интерфейса. Единственное, что следует учитывать — то, что *Router* только перенаправляет соединение при попытках подключения без чтения пакетов и какого-либо анализа. Если сервер выходит из строя, *Router* просто вернет приложению ошибку подключения. Поэтому приложение необходимо дополнять проверкой на ошибки и, при их возникновении, заставляя приложение повторять подключение.

Поскольку *Router* рекомендуется устанавливать на той же машине, что и приложение, можно написать приложение так, чтобы оно осуществляло мониторинг за исполняемым файлом *mysqlrouter* и перезапускало его в случае необходимости. Например, если список серверов в разделе направлений будет исчерпан, можно перезапустить *Router* с новым списком направлений маршрутизации, или же перезапустить его со старым списком серверов.

И не забывайте, что, используя *MySQL Harness*, можно дополнить функциональность вашего приложения. **LXF**

## LINUX FORMAT Вердикт

### MySQL Router 2.0.2

Разработчик: Oracle  
Сайт: dev.mysql.com/downloads/router/  
Лицензия: GPL

Функциональность	7/10
Производительность	10/10
Простота в работе	8/10
Документация	10/10

» Продукт интересный, но чтобы его эффективно использовать, нужны уверенные навыки в разработке плагинов.

## Рейтинг 8/10

# Raspberry Pi Zero

**Лес Паундер** налетает на свеженький Raspberry Pi. На сей раз в нем ноль калорий, но сохранил ли он свою сладость?

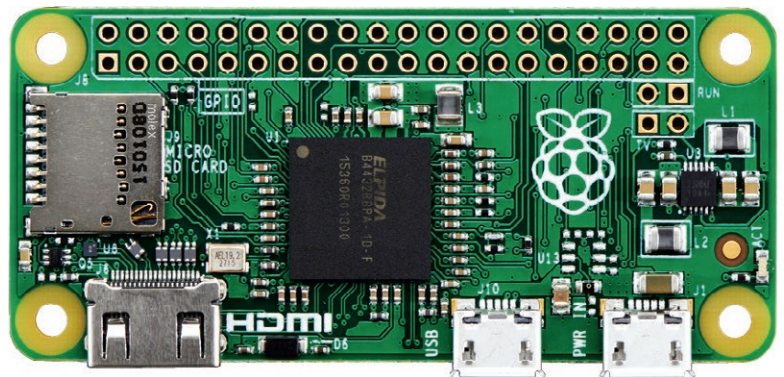
## Спецификации

- » **Процессор** Broadcom 1 ГГц BCM2835
- » **ОЗУ** 512 МБ
- » **Внешняя память** Slot для Micro SD
- » **Порты** Mini HDMI (1080p 60) Micro USB для питания и передачи данных
- » **Прочие функции** Свободный 40-контактный порт GPIO, свободный составной видеопорт.
- » **Габариты** 65 × 30 × 5 мм

Это второй Raspberry Pi, выпущенный в 2015 г., и на сей раз фокус Foundation сместился от мощности в сторону цены. Предыдущие модели Raspberry Pi стоили от 30\$, совершенствуя свои характеристики с каждым релизом. Но после встречи с Эриком Шмидтом из Google Эбен Аптон решил, что следующий Raspberry Pi будет скорее дешевле, чем более производительным.

Raspberry Pi Zero — это компьютер за \$5. Это не опечатка: теперь компьютер стоит как завтрак. По характеристикам Pi Zero ближе к исходному Raspberry Pi, на той же плате (SoC) BCM2835, с процессором ARM11 CPU с частотой 1 ГГц, и на 40% мощнее, чем Pi 1. В Pi Zero есть карта micro-SD, но без привычного механизма защелкивания. Портов по периметру платы мало, только micro-USB для питания и периферийных устройств и mini-HDMI для аудио/видео.

Для периферийного порта micro-USB, как и для mini-HDMI, требуется адаптер, но многие дистрибьюторы, как водится, уже успели устранить этот пробел. На Pi Zero нет DSL- и CSI-коннекторов, а значит, нет совместимости с фирменным сенсорным экраном и видеочамерой Pi. Коннекторы убрали ради снижения цены. Зато в Pi Zero имеется, теперь уже стандартный,



» Raspberry Pi Zero мал, но имеет 40-контактный GPIO для совместимости с огромным количеством расширений Raspberry Pi.

40-контактный GPIO (General Purpose Input/Output), но без штырьковых разъемов — тут вам предлагается шанс проверить свои навыки пайки. Мы протестировали GPIO с Python 3 и Scratch и можем смело сообщить, что все работало как положено. Также мы испытали типичную плату расширения — в данном случае, Unicorn Hat от Pimoroni; с ней после установки тоже всё было в порядке. Pi Zero совместим со многими платами.

## Опять этот IoT

Мы протестировали PiZero с последней версией Raspbian Jessie, который обновился как раз перед выходом Pi Zero, и загружалась она на 52 секунды дольше, чем на стандартном ПК — примерно как в исходном Pi.

Для кого же предназначается Pi Zero? Во-первых, для производителей, которым выгодно обладать недорогой платформой со столь широкой пользовательской базой. Pi Zero — встраиваемая платформа, которая хорошо вписывается в какой-нибудь проект IoT (Internet of Things, Интернет вещей) или другую подобную стационарную систему.

Хотя изначально в Pi Zero нет возможности Wi-Fi подключения, ее легко добавить. На самом деле, народ уже придумал прицеплять USB-модем Wi-Fi к свободным разъемам под Zero. Вторая категория пользователей, кому будет полезен Zero — это те, кто не может позволить себе компьютер. С Pi Zero цена снижается до предела, делая это удовольствие доступным практически любой семье.

А вам зачем покупать Raspberry Pi Zero? Если вы любите робототехнику, проекты, связанные с метеорологией, и ковыряние в железе, то Pi Zero — идеальная платформа для дешевых экспериментов. Использовать в проекте Zero столь же выгодно, как и платы, подобные ESP8266 и многочисленным клонам Arduino. Удалив некоторые компоненты и сохранив удобство Pi в чистом виде, мы получаем дешевую встраиваемую платформу, которая легко вписывается в экосистему расширений Raspberry-Pi. А совместимость с платами расширения и использование той же ОС дает доступ к богатейшей библиотеке ресурсов Raspberry Pi.

Raspberry Pi Zero пополнил семейство данных плат и предлагает сделать невероятно выгодный первый шаг в мир вычислений, кодирования и электроники. **LXF**

## Сравнительные тесты

Test	Pi 2	B+	Zero
SunSpider (мс)	2476	9477	10507
3D	499	1657	1672
Access	190	482	1258
Crypto	194	647	837
Math	141	431	872
String	930	4281	2968

Sysbench	Pi 2	B+	Zero
Prime avg (мс)	29	50	35
Prime min (мс)	29	50	35
Prime max (мс)	54	85	103

## LINUX FORMAT Вердикт

### Raspberry Pi Zero

**Разработчик:** Raspberry Pi Foundation  
**Сайт:** [www.raspberrypi.org](http://www.raspberrypi.org)  
**Цена:** £4/\$5

<b>Функциональность</b>	7/10
<b>Производительность</b>	5/10
<b>Удобство в работе</b>	8/10
<b>Оправданность цены</b>	10/10

» Плата от Raspberry Pi Foundation восхитит и вдохновит целое поколение кодеров. В основном благодаря дешевизне и огромному интересу сообщества.

**Рейтинг 9/10**

# Делите на ноль

Лес Паундер узнал у Эбена Аптона о том, как появился Zero Pi и как добрый совет помог создать компьютер за 5\$.

**В** связи с выходом нового удивительного Raspberry Pi Zero, мы решили встретиться с Эбеном Аптоном [Eben Upton], главой Raspberry Pi Trading, чтобы поговорить о рождении и развитии этого замысла.

**Linux Format:** Зачем появился Pi Zero и как он пришел на рынок?

**Эбен Аптон:** Отчасти по логике «а почему нет» — если ты можешь это сделать, то почему нет? Мы разработали Pi Zero потому, что Raspberry Pi все еще дороговат, не всем по карману, но есть и те, кто пока не может решить, их ли это дело — кодирование, и для них это «первый шаг».

Идея состояла в том, чтобы дать людям своего рода мост в мир компьютеров, по низкой цене. Если он им понравится, они перейдут на Raspberry Pi 2, а Pi Zero используют в каком-нибудь другом проекте. Когда мы впервые задумались о создании дешевого устройства, то думали: а не создать ли



## ЗБЕН АПТОН ПРО LINUX

### Мы верим, что Raspberry Pi приведет в Linux целое поколение.

Pi или нечто под именем Pi, но на базе микроконтроллера? Но очень скоро мы отказались от этой идеи, потому что нам стало ясно, что это должен быть именно Raspberry Pi как таковой, с Raspbian и GPIO, который позволяет стать частью сообщества; ведь сообщество — самое важное в Raspberry Pi. Сегодня, когда я гуглю самые стандартные вопросы по Linux, я получаю ответы, связанные с Raspberry Pi, и это приятно. Мы верим, что Raspberry Pi действительно приведет в Linux целое поколение и совсем другой тип людей, которых мы могли бы представить пользователями Linux.

**LXF:** Изначально была идея сделать из Pi Zero нечто более мощное?

**ЭА:** Да, в 2013-м мы задумывали то, что в итоге превратилось в Pi 2, и это должна была быть более мощная плата, но мы общались с Эриком Шмидтом [Eric Schmidt] из Google, а тот сказал: «глупая идея, надо попытаться сделать что-то менее мощное

и более дешевое», и мы решили, что это отличный совет. Мы тогда приостановили работу над тем Pi 2, который вышел бы в конце 2013/начале 2014 г., выбрав более долгий путь, чтобы создать Pi 2 по той же цене, что и исходный Pi. И это позволило нам поработать в другом направлении, откуда в итоге вышел Raspberry Pi A+ за 20\$ и, наконец, Pi Zero за 5\$.

**LXF:** Кому адресован Pi Zero?

**ЭА:** Главным образом, тем, кому нужен экономичный способ войти в мир компьютеров. Кому еще? Ну, энтузиасты, такие как я, купят его для IoT, робототехнических и домашних встраиваемых проектов, где нужно что-то маленькое и энергоэффективное.

**LXF:** Похоже, это какой-то всплеск интереса к «дешевым» компьютерам, но у CHIP

компьютер за \$9 еще в разработке, а Pi Zero уже здесь.

**ЭА:** Pi Zero не является корыстной попыткой разрушить чьи-то бизнес-модели; это просто лучший Pi, который мы могли сделать по самой низкой цене. Чем мы гордимся, так это тем, что до появления Pi вообще не было подобных машин дешевле 100\$. А теперь у нас есть весь этот огромный мир дешевых компьютеров на Linux, и незачем расстраиваться, когда кто-то другой создает еще один.

**LXF:** На Pi Zero есть 40-контактный GPIO — значит ли это, что он совместим с существующими платами расширения?

**ЭА:** На все 100%: если она подходит для предыдущей модели, то будет нормально работать и с Pi Zero. Мы не выпускали официальных спецификаций для будущих дополнений к Zero, но другие производители уже выпускают платы, довольно близкие по характеристикам.

**LXF:** Заполучив компьютер за \$5, думаете, люди захотят покупать плату расширения, которая стоит дороже?

**ЭА:** Производители уже добились прекрасных результатов в разработке более маленьких плат расширения, и раз там меньше компонентов, они обходятся дешевле; я думаю, платы стоимостью от £10 до £15 будут пользоваться большим успехом, потому что людям захочется иметь и сопутствующее оборудование к этим устройствам. **LXF**



» Pi Zero — компьютер за £4 — порадует всех.



## ПРОБЛЕМЫ СОВМЕСТИМОСТИ

# Поговори со мной, Cortana!

Microsoft отключила функцию голосовой активации Cortana for Android.

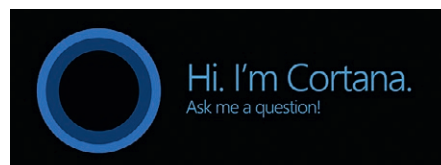
В декабре минувшего года пользователи Cortana for Android в отзывах на странице приложения в Google Play выразили недовольство тем, что Microsoft отключила функцию голосовой активации приложения командой “Hey Cortana”, однако такой шаг был вызван выявленным конфликтом между приложением и нативной системой распознавания голосовых команд мобильной платформы Google (Cortana могла отключить микрофон устройства). При этом установка напоминаний и другие возможности приложения по-прежнему доступны.

Microsoft признала, что при включении “Hey Cortana” пользователи были не в состоянии задействовать общесистемную функцию голосового поиска Android — другими словами, «убить двух зайцев», используя для голосового общения

со смартфоном сразу и “Ok Google”, и “Hey Cortana”, невозможно.

Да, факт одновременного запуска Google Now и Cortana может показаться странным, однако активируемый голосом поиск Google работает в любом приложении, а Cortana по умолчанию использует в качестве поисковой системы Bing. При этом даже у пользователей, предпочитающих искать информацию в Сети с помощью Google, может возникнуть желание использовать обоих ассистентов.

Учитывая крайне негативную реакцию пользователей на ограничение функциональности приложения, было бы удивительно, если бы Microsoft не приняла надлежащих мер. Кроме того, Microsoft планирует сделать Cortana полноценным конкурентом Google Now, а для этого необходимость наличия функции голосовой активации более чем очевидна.



➤ Выявленная еще в процессе бета-тестирования Cortana for Android проблема с работой микрофона на некоторых устройствах сохранилась и в финальной версии.

В любом случае, точные сроки восстановления работоспособности команды “Hey Cortana” названы не были — оставалось предполагать, что это произойдет только по завершении рождественских каникул, после окончания отпусков у инженеров Microsoft.

## ТЕХНОЛОГИИ 3D

# Танго с Lenovo

Нынешним летом Lenovo представит смартфон проекта Tango.

В январе, на пресс-конференции в ходе Международной выставки потребительской электроники (CES) вице-президент Lenovo по разработкам Джефф Меридит [Jeff Meredith] анонсировал смартфон проекта Tango, который появится в продаже уже нынешним летом. Детальной информации очень мало, но известно, что цена новинки будет ниже \$500.

Проект Tango от Google — это технология построения 3D-модели окружающего пространства с помощью мобильных устройств, в частности — смартфонов. В свою новинку Lenovo также предполагает добавить приложения дополненной реальности, которые можно будет использовать, например, в играх, или при покупке мебели и оценке того, как мебель впишется в интерьер конкретной комнаты. В своей работе приложения опираются на данные о физическом пространстве вокруг пользователя смартфона, полученными по 3D-технологии Tango.

У нового смартфона будет дисплей с диагональю менее 6,5” и очень тонкий корпус, однако

название для него в Lenovo еще не придумали. По словам Меридита, в новинке задействован процессор Snapdragon, а работать она будет под управлением Android — что неудивительно, учитывая тесную связь с Google и ее технологиями. По мнению Меридита, добавление возможностей проекта Tango в смартфон являет собой «фундаментальный сдвиг» и «значительный потенциал изменения нашего взаимодействия со смартфонами».

Помимо используемых на современных коммуникаторах двух традиционных камер, устройство будет иметь три дополнительные. Это датчик глубины, камера-рыбий глаз для широкоугольных представлений и RGB-камера для получения очень точных цветных изображений.

Тем не менее, реакцией некоторых аналитиков на анонс от Lenovo стал вопрос о практической ценности проекта Tango. По их мнению, многое будет зависеть от приложений, которые станут его использовать в своих интересах. Напомним, что некоторые производители аппаратных средств (в т.ч. Stanley) уже выпускают небольшие устройства,



➤ Вице-президент Lenovo Джефф Меридит представляет смартфон проекта Tango, сопровождая анонс демонстрацией слайдов.

использующие лазерные технологии и способные быстро измерить размеры помещения с последующим расчетом потребного для обработки его стен количества краски. «3D-визуализация действительно имеет интересные возможности, но приходит от них в восторг прямо сейчас преждевременно», отметил Джек Голд [Jack Gold], аналитик J. Gold Associates. Он напомнил, что Intel разработала технологию RealSense для измерений и 3D-вычислений, применяемую в ряде планшетов, в т.ч. Dell Venue. Но, по его мнению, это не мейнстрим, и он не уверен, что такое нужно ему в телефоне.

С другой стороны, аналитик Moor Insights & Strategy Патрик Мурхед [Patrick Moorhead] уверен, что новый телефон Lenovo «имеет множество достоинств», особенно когда речь идет о дополненной реальности.

## СОБЫТИЯ

# Мобильных ОС много не бывает

Глава Минкомсвязи России обсудил перспективы развития открытой ОС Tizen.

**М**инистр связи и массовых коммуникаций Российской Федерации Николай Никифоров провел рабочую встречу с исполнительным вице-президентом компании Samsung Electronics Хойгуном Ли. Участники встречи обсудили возможности сотрудничества в сфере информационных технологий (ИТ), телекоммуникаций и рынка Интернета вещей. Глава Минкомсвязи России также рассказал о взаимодействии стран БРИКС в сфере инфокоммуникационных технологий (ИКТ) и потенциале расширения глобальной кооперации в этой области.

Стороны обсудили вопросы использования открытой независимой операционной системы Tizen, а также возможность интеграции российских приложений в экосистему Tizen.

«В рамках первой в истории встречи министров телекоммуникаций и ИТ стран БРИКС, которая состоялась в октябре прошлого года в Москве при участии ведущих ИКТ-компаний этих стран, мы обсудили ключевые вопросы, связанные

с возможностью диверсификации глобального рынка информационных технологий, — сказал глава Минкомсвязи. — Считаем, что в перспективе возможно достигнуть баланса, при котором доля одного игрока либо одной страны на этом рынке не будет превышать 50%».

Николай Никифоров напомнил, что вопрос развития разработки ОС в России имеет высокую актуальность, равно как и достижение технологической независимости, а также создание отечественных центров компетенции по разработке системного программного обеспечения.

«В российской экосистеме разработки у нас уже много партнеров», сообщил президент Ассоциации разработчиков и пользователей Tizen «Тайзен.Ру» Андрей Тихонов.

«Мы открываем центр разработки и компетенций нашей ОС в Сколково, — сообщил Хойгун Ли. — Сейчас в России стартует специальная программа по обучению преподавателей, которые смогут готовить российских разработчиков для Tizen».



➤ Глава Минкомсвязи обсудил в Москве перспективы развития открытой ОС Tizen с исполнительным вице-президентом Samsung 4 февраля 2016 г.

Samsung занимается разработкой платформы Tizen для самых разнообразных устройств: уже сегодня Tizen используется в телевизорах, мобильных устройствах, смарт-часах и камерах, а в ближайшее время корейский гигант намерен выпустить и другие девайсы на этой платформе — как для дома, так и для офиса.

## ВЕЗДЕСУЩИЙ ANDROID

## Remix OS

Появилась разновидность Android с поддержкой мыши и окон для ПК.

**G**oogle никогда особо не заботилась об оптимизации Android для ноутбуков и настольных ПК, поскольку усилия компании в этом сегменте были сосредоточены на Chrome OS. Не исключено, что в связи со слухами о возможном слиянии Android и Chrome OS ситуация начнет меняться. Однако уже сейчас существует общедоступное удобное решение.

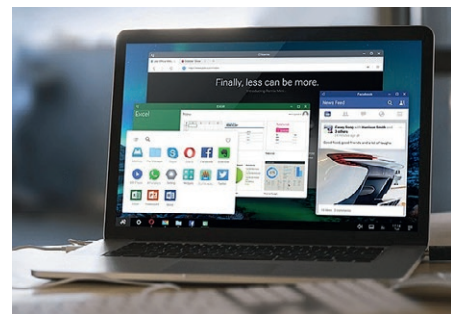
Образовавшие Jide Technology бывшие сотрудники Google всегда позиционировали себя как софтверную компанию. Теперь же они представили версию Android, ориентированную на работу, а не на развлечения, которая может быть запущена на любом ПК.

Remix OS является глубоко модифицированной разновидностью Android ROM, которая обладает функциональностью, более ожидаемой от ноутбука или настольного компьютера. Пользователи могут запускать приложения в полноэкранном режиме

или небольших окнах, и все открытые программы отображаются в панели задач в нижней части экрана. Remix также оптимизирован для работы с мышью или сенсорной панелью — правый щелчок в приложениях вызывает всплывающее контекстное меню.

Дебют Remix OS состоялся в прошлом году на планшете с подключаемой клавиатурной док-станцией, напоминающем устройства Microsoft Surface, а также на легковесном десктопе за \$70. Теперь же Remix OS будет доступна на любом устройстве, поддерживающем загрузку с USB flash. Однако версия для ПК находится пока в альфа-стадии, и ориентирована на разработчиков и ранних тестеров.

Поскольку большинство приложений для Android предназначены для сенсорных экранов, некоторые из них могут не работать должным образом с мышью и клавиатурой. Для установки



➤ Оконные приложения, панель задач, поддержка мыши — возможности у Remix OS больше, чем у стоковой версии Android.

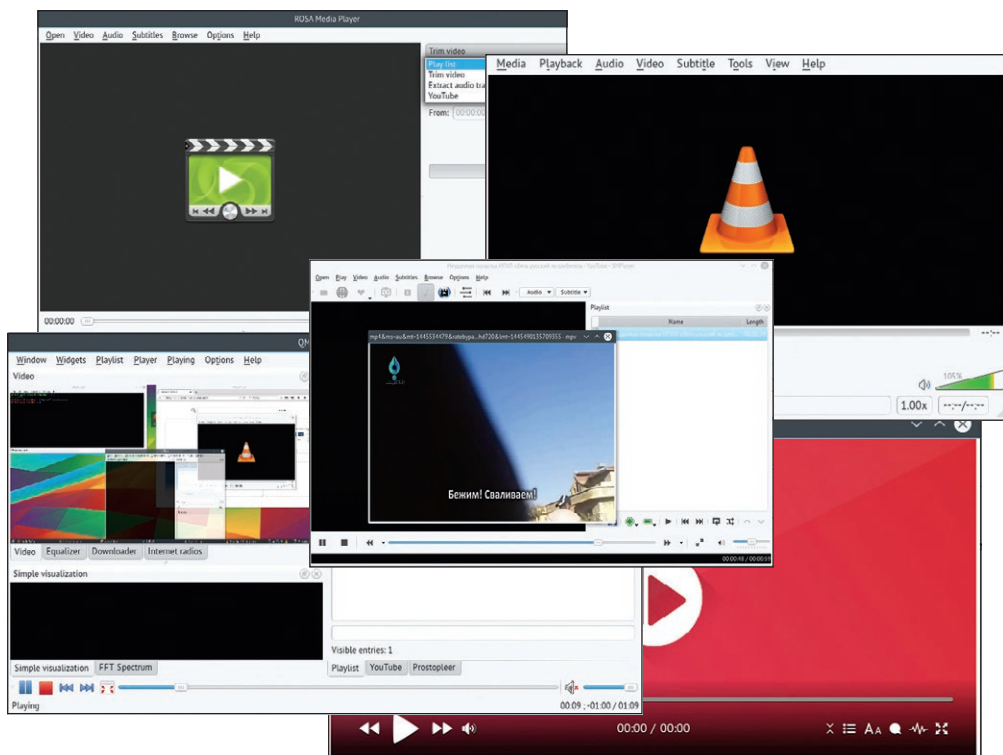
необходим диск USB 3.0 емкостью не менее 8 ГБ и рекомендуемой скоростью записи 20 МБ/сек (попытка установить Remix OS с диска USB 2.0 привела к невозможности загрузки). LXF

# Сравнение

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

## Видеоплееры

Александр Толстой ставит самые передовые видеоплееры в Linux, предлагая им побороться за честь воспроизвести его коллекцию фильмов.



### Про наш тест...

Выбор видеоплеера больше не означает выбор между *VLC* и *Mplayer*. Два эти умелых плеера по-прежнему в деле, но выбор и разносторонность медиа-плееров значительно выросли, и многие из них пытаются быть уникальными, предлагая расширенные функции и более привлекательный интерфейс. В нашем Сравнении мы рассмотрим плееры с разными движками и разными наборами функций. Движки крайне важны для работы плееров и для поддержки видеоформатов, что приводит к естественному различию между, скажем, приложениями на основе *Mplayer* и на основе *FFmpeg*. Помимо этих ключевых элементов, мы также сравниваем наших участников по удобству в работе, доступности, производительности, поддержке разгона оборудования и дополнительным функциям, которые могут — или не могут — добыть лишние очки.

**Е**сть несколько причин знать, какой именно плеер следует использовать на своем настольном Linux. Несмотря на тенденцию переводить все в Сеть и облачные сервисы, многие из нас по-прежнему предпочитают управлять своими мультимедиа и воспроизводить видео как локальные файлы.

Примеры применения тоже весьма многочисленны: видео, скачанные с YouTube или Vimeo; оцифрованное домашнее видео с кассет; клипы с вашего смартфона и DVD; и т.д. Не исключено, что вас вполне устраивает то приложение, которое запускается при двойном щелчке по файлу или когда вы вставляете диск, но существует масса

### Мы выбрали самые функциональные и привлекательные видеоплееры.

плееров, которые достойны знакомства и которые способны предложить гораздо больше. Некоторые из них предлагают более простой интерфейс, лучшее использование оборудования и даже восстановление качества видео с низким разрешением, и мы выбрали наиболее функциональные и привлекательные из видеоплееров, пригодных для установки в среднестатистическом настольном Linux.

Это *VomI*, *VLC*, *QMPlayer2*, *SMPlayer* и *Romp*. Кроме *VLC*, все плееры основаны или на *FFmpeg*, или на *Mplayer*, что в большой мере и определяет возможности плеера. Но, как мы увидим через минуту, движок плеера не обязательно гарантирует наилучшую работу, да и в плеере имеется не только движок. Чтобы определить победителя, давайте посмотрим, как наши конкуренты работают в разных категориях.

### Наша подборка

- »» *Bomi*
- »» *QMplay2*
- »» *Romp*
- »» *SMPlayer*
- »» *VLC*



# Доступность

Найдите и установите их!

Если у вашего дистрибутива Linux хорошие репозитории, установить все пять видеоплееров будет легко. Дистрибутивы вроде Arch Linux или ROSA Desktop Fresh работают хорошо, а Ubuntu и openSUSE продемонстрируют силу своих дополнительных репозиториях — PPA и OBS соответственно. Но вряд ли придется ожидать, что кто-то откажется от своего любимого дистрибутива просто ради пробы видеоплеера. (Потому мы и тестируем их для вас. Всегда пожалуйста.)

Компиляция плеера из исходного кода не является особо серьезным препятствием, но требует времени и определенных навыков, поэтому мы бы предпочли двоичные пакеты. *Bomi* — относительно новый видеоплеер, но у него уже готовы прекомпилированные пакеты для многих дистрибутивов Linux, включая популярные и часть молодых, наподобие KaOS и Chakra (см. <http://bit.ly/BomiPlayer>).

*VLC*, вероятно, наиболее известный кросс-платформенный видеоплеер —

это подлинный ветеран и популярное настольное приложение. Как и следует ожидать, страница загрузки *VLC* (<http://bit.ly/VLCDownloads>) предлагает десятки опций для многих дистрибутивов Linux, и даже если вашей системы нет в списке, есть шансы без проблем найти *VLC* в своем менеджере пакетов.

*QMplay2* отнюдь не так известен; это воистину персональный проект одиночки, польского разработчика Блажея Щигела [Błażej Szczygieł], который приступил к своей разработке в 2012 г. Тот факт, что вы, вероятно, никогда и не слыхивали об этом плеере (если только не читали HotPicks, стр. 100 LXF191/192), как раз и не позволяет ему обзавестись сборками *QMplay2* для Ubuntu, openSUSE, ROSA, Mageia и прочих дистрибутивов.

*SMPlayer* — знакомое название для многих в мире Linux с 2006 г., когда он появился как ответвление *Mplayer*. Официальный сайт *SMPlayer* предлагает только исходники и PPA Ubuntu ([ppa:rvm/smplayer](http://ppa:rvm/smplayer)).



» Компиляция плеера из исходника не особо сложна, однако требует времени и определенных навыков, поэтому мы рассчитываем увидеть двоичные пакеты.

Конечно, пакеты *SMPlayer* есть и для других дистрибутивов, но количество опций очень ограничено — возможно, из-за того, что взаимоотношения *Smplayer* с исходным плеером *Mplayer* можно скорее охарактеризовать как раскол, нежели как дружескую смену направления.

Практически то же самое можно сказать о *Romp*, что означает *Rosa Media Player*, это еще одно ответвление *Mplayer*. *Romp* по умолчанию включен в ROSA Fresh и есть в OpenMandriva. Были созданы сторонние пакеты для Ubuntu ([ppa:nilarimogard/webupd8](http://ppa:nilarimogard/webupd8)) и Arch (см. AUR), но это, вероятно, и все, так что для многих пользователей *Romp*, вероятно, останется неоткрытым.

## Вердикт

- VLC ★★★★★
- Bomi ★★★★★
- QMplay2 ★★★★★
- Romp ★★★★★
- SMPlayer ★★★★★

» VLC доступен практически на всех платформах и дистрибутивах. *Bomi* — следующий.

# Интерфейс пользователя

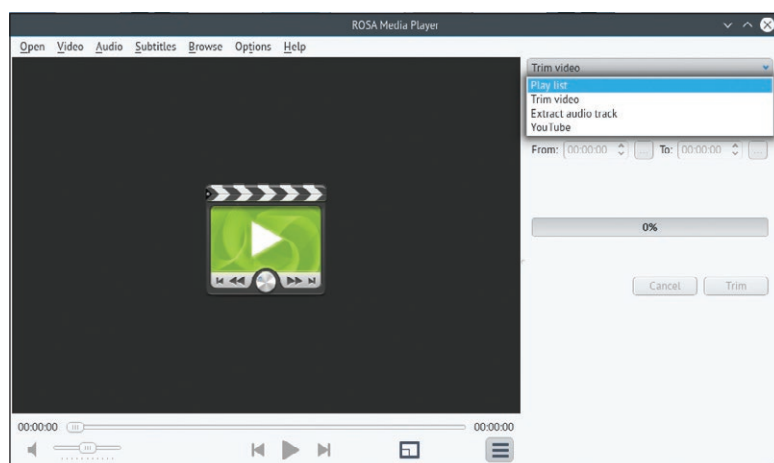
Насколько они дружелюбны и удобны в использовании?

Отсутствие интуитивно понятного интерфейса может отпугнуть новичков от знакомства с новым видеоплеером, даже если во всем остальном он буквально блистает, так что мы заранее рассчитываем на приличный интерфейс пользователя.

*Bomi* — очень аккуратное минималистское приложение, почти все функции которого находятся в ветви его контекстного меню. По умолчанию отображается только всплывающий экран, и кнопки управления воспроизведением и звуком появляются только при наведении курсора на нижнюю часть плеера. Мы сочли это сомнительным решением.

Главное окно *VLC*, возможно, выглядит не слишком стильно, но оно хорошо разработано и предлагает все часто используемые опции и даже некоторые дополнительные функции, например, эквалайзер.

Расположение панелей инструментов и кнопок в *SMPlayer* не слишком отличается от *VLC*, но этот плеер выглядит дешево из-за топорной темы значков и чрезмерно крикливых цветов.



*QMplay2* тоже не лучше — из-за своего перегруженного основного окна с переизбытком элементов, включая визуализацию (для аудиофайлов) и кучу вкладок. Любому, кто впервые применит *QMplay2* для воспроизведения видео, явно захочется закрыть все не относящиеся к делу элементы. Цель дизайна Rosa Labs — быть по возможности дружелюбнее к пользователю.

*И Romp* придерживается чистого, незамусоренного интерфейса с расположенными в центре кнопками управления воспроизведением и такими штрихами, как отдельная кнопка для полноэкранного режима. Верхнее меню также включает наиболее часто используемые функции для быстрого доступа, среди них — воспроизведение DVD и выбор субтитров.

» *Romp* предлагает функции, редкие у настольных видеоплееров.

## Вердикт

- Romp ★★★★★
- VLC ★★★★★
- Bomi ★★★★★
- SMPlayer ★★★★★
- QMplay2 ★★★★★

» *Romp* делает часто используемое приложение проще для новичков.

# Функциональность

Награда за самый всесторонний видеоплеер присуждается...

**П**ричина существования такого большого количества хороших плееров (причем постоянно появляются все новые и новые) в том, что каждый из них старается предложить уникальный набор функций.

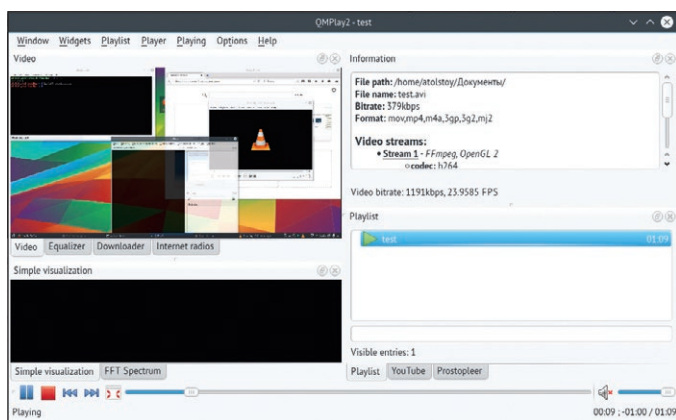
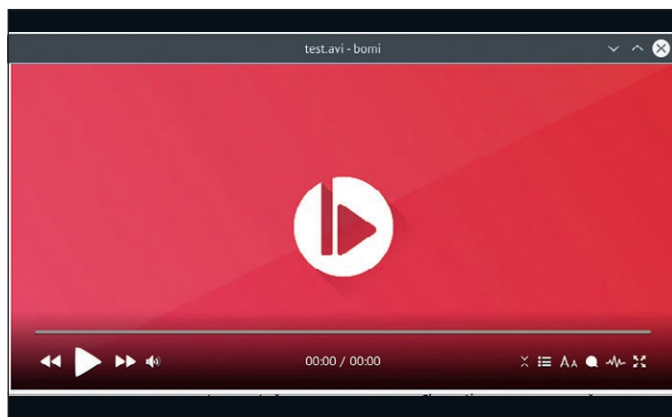
Ответвления появляются из-за нередких разногласий среди разработчиков по вопросу о том,

что же превращает плеер в «отличный» (в частности, именно так произошло с *SMPlayer*). Хотя многие пользователи Linux вообще не обращают внимания на функции плеера — им просто хочется посмотреть кино — есть и такие, кого интересует наличие всяческих функций, и это особенно очевидно, когда речь заходит о решении проблем,

например, при необходимости воспроизвести потоковое видео, настроить коэффициент изображения или подключить внешние субтитры. У каждого имеется собственный список требуемых функций, но мы постараемся выделить усредненный набор, чтобы все увидели, что скрывают в своих меню и панелях плееры нашего Сравнения.

## Bomi ★★★★★

У *Bomi* обширный набор функций, в том числе неограниченная история воспроизведений, автоматическое создание плей-листа и восстановление и расширенная работа с субтитрами. Кроме того, имеется много удобных функций, таких как настройка звука клавишами **Ctrl++** and **Ctrl+-**; перемещение субтитров; автоматическое сокрытие плей-листа, который выпадает из правой части основного окна, и эквалайзер с десятками готовых настроек. Традиционная нижняя панель с кнопками управления воспроизведением и громкостью в *Bomi* тоже обогатилась несколькими дополнительными кнопками для переключения между аудиотреками и масштабирования субтитров. Разработчики *Bomi* уделили особое внимание удобству любителей фильмов с высоким разрешением и сериалов, добавив простое меню для просмотра дисков Blu-ray и возможности автоматического анализа URL и воспроизведения ссылок, скопированных в буфер.



## QMplay2 ★★★★★

После пары дней использования *QMPlay2* нам не показалось, что каких-то функций в нем не хватает. Фактически, в этом приложении тонны дополнительных функций практически на любой случай. Вот некоторые: программа скачивания видео; 8-канальный эквалайзер; интеграция с YouTube и Ploor.com; огромный список predetermined онлайн-радиостанций; интерфейс с высокой степенью индивидуальной настройки (правда, немного перегруженный по умолчанию) и простой доступ к основным опциям *FFmpeg*, на котором основан *QMplay2*. Работать с *QMplay2* весьма приятно; мы смогли без проблем смотреть видео с YouTube (даже через корпоративный прокси), сохранять видео на жесткий диск через *youtube-dl* и даже делать весьма необычные вещи, например, производить многоканальный звук с разной частотой, используя встроенный генератор тона. Полноэкранный режим также предлагает автоматически скрывать панель плей-листа с левой стороны окна.

# Гибкость GUI

Есть ли у них возможность простой индивидуальной настройки?

**В**идеоплеер не должен заставлять приравливаться к его интерфейсу по умолчанию. Вместо этого нужна возможность менять настройки, чтобы добиться более приятной внешности или работы либо отключить ненужные элементы.

У *Bomi* очень привлекательный дизайн интерфейса пользователя, так что, по нашему мнению, в особых изменениях он не нуждается. Однако вам дается возможность выбора: другого скина из списка из 15 вариациями; другого шрифта; поведения плеера при остановке воспроизведения; настройки

индивидуальных шрифтов меню; изменения режима интеграции системного лотка; клавиш быстрого запуска и действий мыши и т. д. Интерфейс *VLC* имеет два режима: «родной» и со скинами. Кроме этого, есть специальный Toolbars Editor, позволяющий перераспределить кнопки для встроенного стиля и даже изменить кнопки для слайдера, и наоборот. *QMPlay2* спасает свой менее чем неидеальный интерфейс по умолчанию, предоставляя возможность изменить практически все. Интерфейс полностью модульный, и его можно оптимизировать

по своему вкусу. То же касается и других элементов плеера: количество настроек огромно. Вы можете управлять каждой частью *QMplay2*. *SMplayer* предлагает очень удобные возможности управления в *Preferences > Interface*, где можно выбрать один из четырех графических режимов (*Basic*, *Mini*, *MPC* и *Skinnable*) и настроить индивидуальный набор значков и стилей виджетов. В данном тесте *Romp* стал аутсайдером: его список настроек очень скромен. Предполагается, что по большей части вы будете принимать его как он есть.

## Вердикт

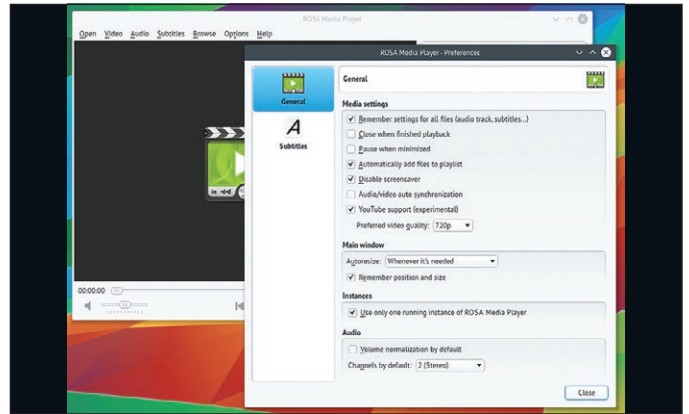
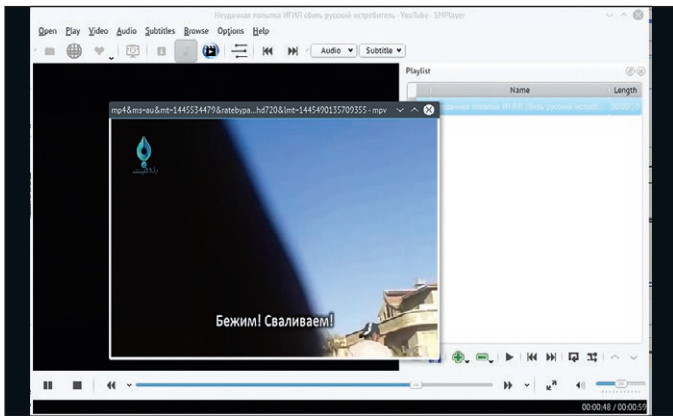
- Bomi ★★★★★
- SMPlayer ★★★★★
- VLC ★★★★★
- QMPlay2 ★★★★★
- Romp ★★★★★

» И *VLC*, и *SMPlayer* идеальны в создании множества стилей GUI.

## Romp ★★★★★

Изначально целью разработчиков *Romp* было создание простого, но и одновременно богатого функциями приложения; но в конечном итоге баланс сместился в сторону простоты, и многие редко используемые функции были ликвидированы. Вот почему *Romp* не может конкурировать по этой части с остальными плеерами нашего Сравнения.

Однако ему тем не менее удается заполучить «всего» четыре звезды в данной номинации, поскольку те функции, которые у него имеются, очень хороши и выгодно отличают его от *SMPlayer*. *Romp* способен делать съемку видео с рабочего стола; на лету настраивать видеоизображение с помощью временной шкалы; применять фильтры для улучшения качества видео; загружать субтитры с [opensubtitles.org](http://opensubtitles.org) и сохранять аудиотреки в форматах MP3 и OGG. Не так давно в *Romp* была еще и поддержка YouTube, но из-за изменений в API YouTube она перестала работать.



## SMPlayer ★★★★★

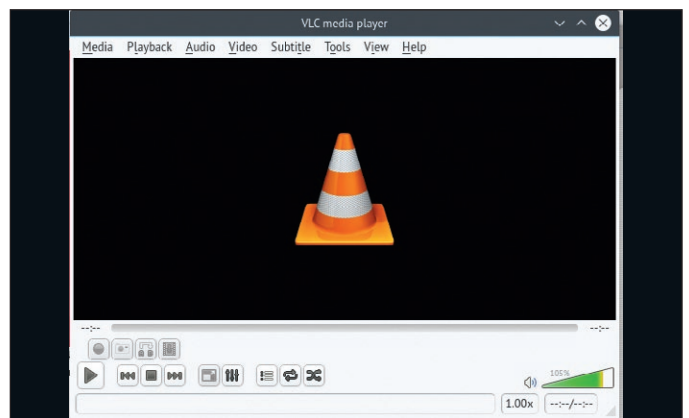
*SMPlayer* предлагает настройку для выбора движка, который влияет на производительность, поддержку форматов и прочие аспекты приложения. Из-за всей этой шумихи и ветвлений *MPlayer* в прошедшие годы *SMPlayer* очень мудро предлагает вам выбрать версию *MPlayer* или *MPV*, которую вы бы предпочли использовать.

Исследование *SMPlayer* обнаруживает интересный набор функций, в том числе: улучшенная поддержка MKV (например, заказанные части, 3D-метаданные); настройки качества для YouTube и Vevo; опция отображения сразу двух субтитров и точный поиск, и т.д. Многие мощные функции *SMPlayer*, такие, как работа с URL, работают нормально только с движком *MPV*, так что позаботьтесь об этом. В *SMPlayer* есть также отдельное приложение для сетевых видео под названием *SMTube* — это симпатичная комбинация браузера и видеоплеера.

## VLC ★★★★★

Большинству пользователей *VLC* знакома лишь малая часть его возможностей. Например, *VLC* пригоден для конвертирования мультимедиа (*Media > Convert/Save*) — очень удобная функция для многих случаев, например, для сохранения потоковых видео и шоу прямого эфира на ваш жесткий диск или извлечение аудиотреков из музыкальных видео и перекодирование их в MP3 или OGG. Та же функция превращает *VLC* в программу записи рабочего стола, если вы выберете целью съемки свой дисплей.

Еще одна превосходная функция *VLC* — возможность усиливать звук более чем на 100%. Не пытайтесь, впрочем, использовать ее со своим любимым треком — возникнет сильное искажение, и вдобавок это небезопасно для вашего слуха; однако подобное усиление звука — отличный шанс прослушать слишком тихие треки, например, запись разговора. Те, кто занимается слежкой, чрезвычайно любят *VLC*!



# Аппаратное ускорение

Эффективно ли оно, и какой плеер снижает нагрузку на CPU?

Эта тема порой сбивает с толку, потому что надо учитывать два фактора: один — это декодирование видео с аппаратным ускорением (которое зависит от используемого видекодека), а другой — устройство вывода с аппаратным ускорением (если оно поддерживается видеодрайвером ядра и представлено с подходящим API). Чтобы разгрузить CPU, нужно реализовать оба.

У *Vomi* хорошая поддержка ускорения видекодека, но не предлагается настройка для выбора вывода видео. Хотя даже

*MPV* — на котором основан *Vomi* — поддерживает VAAPL и VDPAU, *Vomi* не позволяет явно такое задать, так что во многих системах (в основном это касается Radeon) ускоренное перекодирование оказывается бесполезным.

*VLC* также использует системное аппаратное ускорение, и имеет только общее окно-флажок для 'Accelerated video output' в настройках *Video*, которое во многих случаях просто ничего не делает. Наилучшая поддержка аппаратного ускорения — у *QMPay2*; его *Settings* управляют

всеми функциями плеера. Например, раздел *Playing settings* содержит список методов вывода видео. Вы можете сменить приоритет метода, перетаскивая его мышью вниз или вверх. У *SMPlayer* есть еще и рабочие опции для VAAPL и VDPAU, которые работают на поддерживаемом оборудовании, например, на ноутбуке со старым мобильным чипом Radeon HD. *QMPay2* и *SMPlayer* были единственными, которые сумели использовать свое оборудование и компьютер с графикой Intel Ivy Bridge. Ну, а *Romp* не справился ни с чем.

## Вердикт

- QMPay2 ★★★★★
- SMPlayer ★★★★★
- Vomi ★★☆☆☆
- Romp ★★☆☆☆
- VLC ★★☆☆☆

» И в *QMPay2*, и в *SMPlayer* имеются настройки для ускорения.

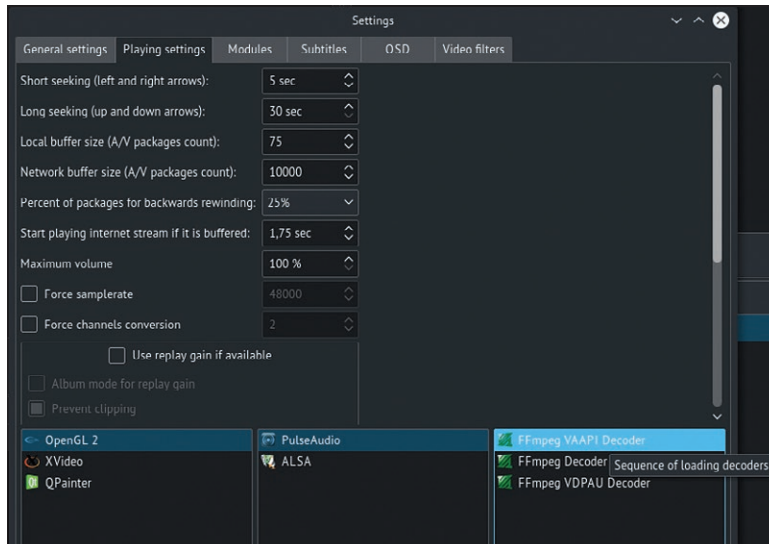
# Состояние разработки

Активно ли они поддерживаются?

**С**остояние разработки видеоплеера — важный фактор, который следует учитывать в первую очередь потому, что постоянно появляются новые кодеки и новое оборудование, и плеер обязан идти в ногу со временем, иначе он становится бесполезным.

Хорошим примером того, что способно произойти, когда разработка не ведется, служит небезызвестная история с изменением API YouTube, которое произошло в начале 2015 г. и привело к тому, что интеграция кода во многих сторонних приложениях перестала работать, и разработчикам пришлось выпускать отладки. Сейчас интеграция YouTube использует отдельную утилиту под названием *youtube-dl*, которая позволяет скачивать видео и затем воспроизводить его локально — и пока это единственный приемлемый обходной путь.

*Bomi*, *QMPlay2* и *SMPlayer* очень хорошо сотрудничают с *youtube-dl*, а вот в *VLC* данная функция не реализована. В целом, все эти четверо получают неплохую поддержку и достаточно часто выпускают обновления. *Bomi* вдохнул новую жизнь в старый *SMPlayer* благодаря энергично обновленному репозиторию Git. *QMPlay2* обновляется каждые две недели, и с 2012 г. прошел



большой путь, тогда как у *SMPlayer* более долгие перерывы между релизами, но тем не менее он продолжает прогрессировать. *VLC* — очень популярный кросс-платформенный плеер; в частности, он является единственным простым решением для скачивания фильмов на iPad, поскольку в этом случае вы не располагаете доступом к файловой системе. Кроме того, вокруг проекта

VideoLAN образовалось большое сообщество, что гарантирует регулярное обновление его нового сайта (<http://www.videolan.org/news.html>). Есть множество и других причин, по которым поддержка *VLC* надолго останется незабываемой. И напротив, будущее *Romp* выглядит весьма неопределенным, поскольку команда разработки уже пару лет не обновляла свой плеер.

**Вердикт**

**Bomi** ★★★★★

**QMPlay2** ★★★★★

**VLC** ★★★★★

**SMPlayer** ★★★★★

**Romp** ★★★★★

» Romp нужно больше внимания, однако другие плееры обновляются регулярно.

# Документация и поддержка

Сможете ли вы получить консультации по каждому плееру онлайн?

**Р**уководства пользователя, справочники, статьи, FAQ и wiki очень важны. Допустим, вам захочется по больше разузнать о плеере, которым вы наслаждаетесь уже давно, или изучить, как использовать некую функцию. Независимо от причины, документация является удобным инструментом для решения проблем и освещения важных функций плеера.

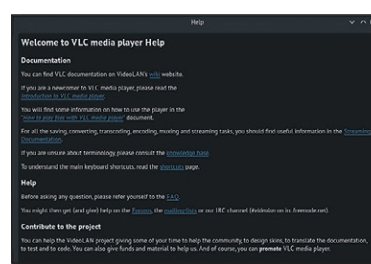
У *Bomi* очень привлекательный сайт с раздельной поддержкой, но он сравнительно скуден и имеет только краткие инструкции по таким вопросам, как создание журналов отладки или создание и отправка вопросов разработчикам проекта. Мы полагаем, что, поскольку *Bomi* — молодой проект, в будущем ситуация улучшится.

*VLC*, будучи хорошо известным плеером, обладает наилучшей документацией, включая подробные руководства пользователя для разных платформ, активные форумы (на <https://forum.videolan.org>) и массу

сторонних материалов (таких, как знаменитая Arch Wiki) со всяческими подсказками, хитростями и настройками. Документация *VLC* переведена на множество языков и доступна не только в формате HTML, но также и в популярном PDF.

*SMPlayer* старается предоставить хорошую поддержку и предлагает форумы, FAQ, публичный баг-трекер и страницу запроса помощи. Главный сайт очень удобен и позволяет новичкам быстро выяснить особенности *SMPlayer* и познакомиться с его специальными функциями. В разделе Help имеются полезные ссылки на руководство пользователя, FAQ и даже опции командной строки — короче, весьма достойный набор.

*QMPlay2* — отличный плеер, но о нем мало где можно узнать подробнее, помимо Readme на сайте <https://github.com/zaps166/QMPlay2>. Отсутствие документации сейчас является препятствием, не позволяющим *QMPlay2* добиться популярности.



У *Romp* когда-то имелся очень сильный уровень поддержки, и хотя его разработка ныне затихла, существующие материалы по-прежнему доступны и актуальны. Разработчики создали неплохую wiki с удобным руководством для делающих первые шаги и даже с промо-видео на Youtube (<http://bit.ly/1LpStSz>). Конечно, этого недостаточно, чтобы превзойти обширную документацию *VLC*, но дает хороший пример того, как можно продвигать и поддерживать ответвление видеоплеера.

» Мы воздаем должное усилиям команды VideoLAN и преданному сообществу плеера.

**Вердикт**

**VLC** ★★★★★

**Romp** ★★★★★

**SMPlayer** ★★★★★

**Bomi** ★★★★★

**QMPlay2** ★★★★★

» VLC в данном тесте лидирует, тогда как SMPlayer и Romp — середнячки.

Видеоплееры

# Вердикт

**Н**есмотря на преизрядное количество многообещающих и активно развивающихся проектов, ветеран этой стаи, *VLC*, по-прежнему остается в ранге альфа.

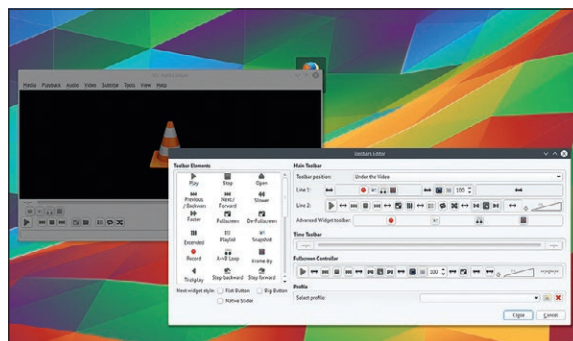
*VLC* — одним из лучших примеров мира ПО с открытым кодом, корни которого уходят в 2001 г., когда появился проект VideoLAN. С тех пор этот видеоплеер с его логотипом в виде дорожного конуса стал одним из наиболее значимых, и мы гордимся тем, что это — приложение с лицензией GPL.

В нашем Сравнении *VLC* набрал максимальное число баллов во всех категориях, кроме аппаратного ускорения. Мы обязаны признать, что это не универсальная проблема, а скорее частный случай с нашей тестовой конфигурацией. Сообщают, что *VLC* на многих компьютерах использует и VA-API, и VDPAU (попробуйте проверить это сами с помощью команды `$vlc --ffmpeg-hw -v`). Отсутствие прямой поддержки YouTube хоть и досадно, но не критично.

А вот за второе место развернулась ожесточенная потасовка между *SMPlayer*, *QMplay2* и *Bomi*. Это плееры очень сильные, и нам пришлось чрезвычайно по душе, что *QMplay2* показывает, влияет ли VA-API на нагрузку CPU — перетащите пункт и выставьте ему высший приоритет, и вы мгновенно увидите результат. *QMplay2* по сути является набором инструментов с сотнями способов пригодиться аспектов (однако для многих из них отсутствует руководство пользователя).

*SMPlayer* тоже хорошо работает со всеми нужными настройками в тщательно продуманном интерфейсе. Да и *Bomi* — весьма многообещающий проект с большим количеством функций и минималистским дизайном; однако этот плеер выглядит, как устройство инопланетян, поскольку скрывает все свои инструменты.

*Romp* — превосходное положение для повседневного



» С помощью Toolbars Editor можно переделать всё, придав *VLC* уникальный вид.

пользования, но уж очень надолго его оставили без внимания — до такой степени, что некоторые функции перестали работать (например, поддержка YouTube). Тем не менее, *Romp* хорош для создания скринкастов, их отладки и извлечения аудиотреков из фильмов.

**VLC — один из лучших примеров мира ПО с открытым кодом.**

**I VLC** ★★★★★  
 Сайт: [www.videolan.org/vlc](http://www.videolan.org/vlc) Лицензия: GPL и LGPL Версия: 2.2.1  
 » По праву остается самым сбалансированным и функциональным плеером.

**IV Bomi** ★★★★★★  
 Сайт: <https://bomi-player.github.io> Лицензия: GPLv2 Версия: 0.9.11  
 » Элегантный интерфейс и много настроек, но пока слишком молод.

**II SMPlayer** ★★★★★★  
 Сайт: <http://smplayer.sourceforge.net> Лицензия: GPL Версия: 15.9  
 » Благодаря поддержке движка MPV это по-прежнему передовой плеер.

**V ROMP** ★★★☆☆  
 Сайт: <http://bit.ly/RompPlayer> Лицензия: GPLv3 Версия: 1.6  
 » Возраст сказывается, но для просмотра фильмов и съемки рабочего стола неплох.

**III QMplay2** ★★★★★★  
 Сайт: <http://bit.ly/QMPlay2> Лицензия: LGPL Версия: 15.10.18  
 » Его подвело отсутствие ресурсов, но это очень многообещающий проект.

**Обратная связь**  
 Какой ваш любимый видеоплеер для Linux? Мы бы хотели услышать о вашем опыте. Пишите нам на [lx.f.letters@futurenet.com](mailto:lx.f.letters@futurenet.com).

## Рассмотрите также...

**О**бщее число видеоплееров для Linux отследить трудно, поскольку это популярная область разработки программ с открытым кодом, и незачем ограничиваться пятеркой из нашего Сравнения. Так, видеоплеер *Rage* ([www.enlightenment.org/about-rage](http://www.enlightenment.org/about-rage)), являющийся частью рабочего стола *Enlightenment*, предлагает довольно продвинутые функции — например,

предпросмотр во время поиска (как у онлайн-плееров); *Gnome Videos* (ранее — *Totem*), пусть с виду и примитивный, очень хорош для воспроизведения файлов через общие сетевые ресурсы. Или, возможно, вы предпочитаете *Miro* ([www.getmiro.com/download](http://www.getmiro.com/download)), полное iTunes-подобное замещение, где можно делать все, включая приобретение видео с Amazon и т. д. Несмотря на то, что сейчас

век облачных сервисов и Интернета Вещей, кое-что остается неизменным: ваши файлы. Философия Unix, где все является файлом, никуда не исчезнет, а значит, всё равно нужно будет хорошее приложение для воспроизведения локальных медиа-файлов с вашего жесткого диска. Откиньтесь в кресле и расслабьтесь, да позаботьтесь, чтобы ваш скринсейвер не испортил впечатления от просмотра! **LXF**

# ХАКНЕМ!

Лес Паундер ведет нас в мир, где холодильники могут размещать твиты, сады — поливать себя сами, и нечто приглядывает за вашим почтальоном...

**Н**аши дома преобразуются технологиями и меняются в течение нашей жизни. Во многих домах сейчас стоят умные телевизоры и системы центрального отопления, которые знают, когда вы приезжаете домой и до какой температуры надо нагреть каждую комнату в доме; и даже холодильники умеют отправлять сообщения, чтобы напомнить вам купить молока. А легко ли это — создать себе уголок технологии XXI века?

Домашняя автоматизация становится популярным проектом для многих хакеров-любителей, по большей части благодаря росту популярности Raspberry Pi и, в частности, выходу Pi 2, который является мощной и экономичной платформой. Контакты Pi GPIO (General Purpose Input/Output) могут взаимодействовать с массой обычных электронных

компонентов, таких, как датчики, реле и транзисторы. Всех их можно запрограммировать с помощью Python и других языков. Pi также предлагает соединение Ethernet, позволяющее поддерживать надежную связь с внешним миром и удаленно управлять

## Попробуем окунуться в океан возможностей проектов, использующих Pi.

проектами. У нас также есть Raspbian Linux — стабильная и надежная операционная система с растущей базой поддержки. Домашняя автоматизация охватывает множество областей, например, управление средой и безопасность.

Но насколько это легко — приступить к автоматизации своего дома? Можно ли создать проект

домашней автоматизации на Raspberry Pi? В нашей статье мы попробуем окунуться в океан возможностей проектов, использующих Pi и ряд готовых компонентов с добавлением кода Python и данных, находящихся в свободном доступе во внешних источниках.

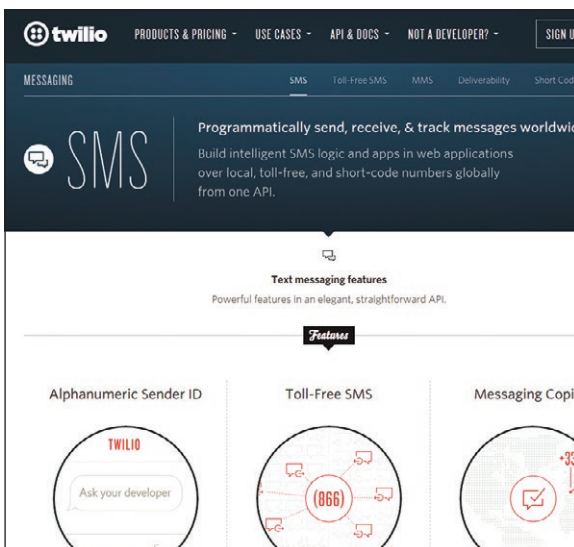
Проработав каждый из этих проектов, вы узнаете много нового: как использовать датчики для определения движения и включения света; как хакнуть скромный дверной звонок, чтобы он обзавелся функцией отправки SMS; как добавить датчик для определения наличия почты и отправки нам фото по электронной почте, чтобы мы больше не теряли свои посылки и письма. И всё это становится возможным благодаря компьютеру за £30, парочке датчиков и кое-какому волшебству Python.

# Письма от двери

Получайте SMS-ки при появлении посетителей.

Скромный дверной звонок справляется с задачей предупредить нас о появлении гостей, только когда мы находимся в пределах слышимости; но это можно исправить с помощью технологии Интернета Вещей [Internet of Things, IoT]. Для данного проекта мы использовали дешевый беспроводной звонок (на Amazon он стоит £5). Мы убрали кнопку и обнаружили схему, которая использует простой переключатель без фиксации на батарейке 12 В. Raspberry Pi GPIO не может напрямую работать с напряжением более 3,3 В, поэтому сперва надо сменить источник питания на более слабый.

Для механизма нажатия кнопки надо припаять два проводка к контактам батарейки. При нажатии переключатель соединяет питание с заземлением и эффективно снижает силу тока, изменяя состояние устройства со Вкл на Выкл и создавая пусковой механизм. С помощью мультиметра определите должные выводы для вашего устройства и припаяйте к ним провода. Для большей надежности крепления проводов к контактам используйте клеевой пистолет. Присоедините положительный полюс батареи к выводу 3V3 GPIO, а заземление [GND] батареи — к заземлению [GND] вашего Raspberry Pi. На своем переключателе присоедините кнопку к выводу 17 (по нумерации Broadcom), а другой провод — к выводу 3V3 GPIO.



» Twilio — наш мост между звонком и SMS. Это SMS-сервис онлайн, который можно использовать через библиотеку Python.

Для отправки SMS необходимо создать себе пробную учетную запись на <https://www.twilio.com>. Загрузите свой Raspberry, перейдите в терминал и введите \$ sudo w pip3 install twilio для установки API Twilio для Python. Откройте приложение Python 3 через меню Programming, создайте новый файл и тут же сохраните его как Doorbell-SMS.py.

Мы начинаем наш проект с импорта API Twilio, библиотеки time и библиотеки GPIO:

```
from twilio.rest import TwilioRestClient
import time
import RPi.GPIO as GPIO
```

Затем следует настроить GPIO на использование схемы расположения контактов Broadcom; задайте контакт 17 как ввод и установите его резистор на снижение тока:

```
GPIO.setmode(GPIO.BCM)
GPIO.setup(17, GPIO.IN, GPIO.PUD_DOWN)
```

Далее создадим функцию для отправки текстовых сообщений через API Twilio. Замените учетную запись и данные аутентификации на свои, а телефонные номера to= и from\_= — на соответствующие вашим требованиям:

```
def sendsms():
    ACCOUNT_SID = "ACCOUNT ID"
    AUTH_TOKEN = "AUTH TOKEN"
    client = TwilioRestClient(ACCOUNT_SID, AUTH_TOKEN)
    message = client.messages.create(
        body="Звонок зазвонил",
        to="НОМЕР, КУДА ПОСЫЛАТЬ SMS",
        from_="ВАШ НОМЕР ТЕЛЕФОНА TWILIO",
    )
    print(message.sid)
    time.sleep(5)
```

Последняя часть нашего кода является циклом, который будет прогоняться постоянно.

Мы наблюдаем за током на контакте 17 для запуска цикла, и когда он обнаружится, вызываем функцию отправки SMS на ваш мобильник:

```
while True:
    GPIO.wait_for_edge(17, GPIO.FALLING)
    sendsms()
```

Сохраните получившийся код и нажмите на Run > Run Module, чтобы его протестировать.

» Мы купили звонок дешевле £5 и взяли его за основу этого беспроводного проекта.

Вам нужны...

- » Любой Raspberry Pi, но лучше А+
- » Беспроводной звонок
- » Умение паять
- » Учетная запись Twilio
- » Свежая ОС Raspbian
- » Весь код см. на <https://github.com/lesp/LXFPiHome-SMSDoorbell>

## Внешние сервисы

Работа с внешними источниками данных и сервисами — это потрясающая область исследования с помощью Raspberry Pi. Источников множество — например, погода, астрономические данные и данные мобильного общения.

Источники данных можно использовать как способ запуска события в физическом мире — например, чтобы включить вентилятор при определенной температуре; или же ради выходных данных,

например, для регистрации колебаний атмосферного давления.

В данном проекте мы использовали сервис Twilio для доступа к функции отправки SMS через API Python. Twilio — дешевый и надежный сервис для проектов, и по завершении бесплатного пробного периода стоит всего \$1 в месяц и около \$0,04 за SMS. С помощью Twilio мы можем пойти дальше и превратить наш несложный IoT (Internet

of Doorbells — Интернет Дверных Звонков) в действительно мощное устройство, придав ему функциональность MMS (Multimedia Messages), содержащих видео и фото, сделанные камерой Raspberry Pi.

Есть и другие провайдеры SMS; один из них — [www.smsapi.co.uk](http://www.smsapi.co.uk), там сами используют Pi для отправки и получения SMS и предлагают 2000 бесплатных SMS.

# Свет при входе

Да будет свет, когда вы открываете дверь.

Вам нужны...



» Любой Raspberry Pi A+, B+ или Pi 2

» Свежая ОС Raspbian

» Розетки питания Energenie и Pi Remote <https://energenie4u.co.uk>

» Геркон

» Контактные перемишки

» Магниты

» Весь код см. на <https://github.com/lesp/LXFPi-Home-EntryLight>

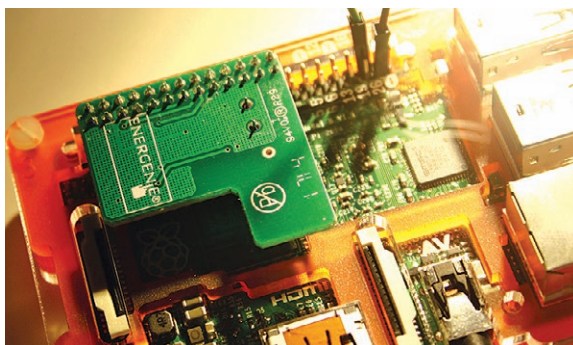
Довольно грустно входить зимой в темный дом, так что воспользуемся готовыми компонентами для создания яркого проекта, приветствующего ваше возвращение домой. Для начала надо к первым 26 контактам GPIO на вашем выключенном Pi присоединить розетку Energenie. (Для справки, контакт 1 — это контакт, ближайший к слоту SD-карты). Плата точно усядется над Pi, без всяких выступающих частей.

Теперь соедините кабелем-перемычкой «мама-мама» GPIO20 и GND через свободные контакты GPIO. (Желая удлинить кабель, просто пристыкуйте кабели «папа-мама» до достижения желаемой длины). К одному концу кабеля присоедините магнитный геркон-защелку, а затем к другому. Клейкой лентой прикрепите геркон к дверному косяку и присоедините магниты на уровне геркона, но уже на самой двери, чтобы при закрытой двери геркон замыкался.

Загрузите свой Pi и откройте терминал. Для установки библиотеки Energenie для Python 3 скомандуйте \$ sudo pip-3.2 install energenie. После установки откройте новую сессию Python 3 через меню Programming. Для подключения Energenie к нашему Pi откройте оболочку IDLE и введите from energenie import switch\_on, switch\_off. Теперь подключите Energenie и нажмите кнопку Green на шесть секунд. Это заставит Energenie искать новый трансмиттер. Снова в оболочке IDLE введите switch\_on(1). Ваш Pi соединится с устройством и присвоит ему '1', и этот процесс можно повторять для четырех устройств. При открытой IDLE щелкните по File > New Window и сохраните свою работу как **entrylight.py**.

Начнем с импорта библиотек для данного проекта:

```
from energenie import switch_on, switch_off
import time
```



» Устройство от Energenie превосходно размещается поверх первых 26 контактов Pi 2 или поверх всех контактов GPIO более старого Raspberry Pi.



» Ресивер в розетке Energenie содержит реле для включения и отключения электропитания.

```
import RPi.GPIO as GPIO
```

Библиотека energenie управляет устройствами для включения света, time управляет длительностью включения устройств, а RPi.GPIO — библиотека для работы с GPIO.

```
GPIO.setmode(GPIO.BCM)
GPIO.setup(20, GPIO.IN, GPIO.PUD_UP)
switch_off()
```

Далее мы настраиваем GPIO на использование схемы расположения контактов Broadcom и настраиваем GPIO20 как ввод, приводя его внутренний резистор в верхнее положение, чтобы подать ток на этот контакт. И, наконец, мы выключаем Energenie, чтобы убедиться в их готовности. Основной код использует структуру try...except для запуска бесконечного цикла:

```
try:
    while True:
        if GPIO.input(20) == 1:
            switch_on()
            time.sleep(30)
            switch_off()
```

Внутри цикла используется условное утверждение, чтобы проверить, был ли запущен ввод, т.е. была ли открыта дверь. Если это так, то устройства включаются на 30 секунд и снова выключаются.

```
else:
    switch_off()
except KeyboardInterrupt:
    print("EXIT")
    switch_off()
```

Мы заканчиваем условное утверждение условием else. Оно выключает устройства, и цикл работает непрерывно. Структура try...except замыкается методом для завершения проекта: нажатие CTRL+c завершит проект и при необходимости отключит устройства. Доделав код, сохраните свою работу и нажмите Run > Run Module, чтобы протестировать код.

## Energenie

Работать с устройствами с высоким напряжением положено профессионалам, однако с помощью Energenie можно существенно снизить риск.

Устройства Energenie по сути своей просто ресиверы 433 МГц, которые управляют реле; это компонент, который использует низкое напряжение для управления магнитным переключателем в цепи высокого напряжения. На Raspberry Pi имеется трансмиттер, который дает ресиверам команду включаться и выключаться.

Устройства Energenie дают безопасный способ управлять электропитанием. Стандартная библиотека Python для Energenie довольно громоздкая и требует от пользователя управления контактами GPIO, используемыми трансмиттером для соединения с каждым устройством и выдачи инструкции по включению/выключению.

Эта библиотека намного упростилась благодаря Бену Наттелу [Ben Nuttal], члену Образовательной команды Raspberry Pi Foundation, и Эми Мейзер

[Amy Mather], которую многие знают как Mini Girl Geek; это юная хакерша. Их улучшенная библиотека, которую мы применили в нашем руководстве, требует знания номера каждого устройства, чтобы выдавать инструкцию одному либо всем устройствам сразу.

Библиотеку можно найти на GitHub, если вы захотите рассмотреть код и разобраться, как она работает. Загляните на <https://github.com/RPi-Distro/python-energenie>.



# Следим за посылками

Получайте по электронке вести о своих курьерских доставках.

**В**ы всегда поддерживаете кикстартеров, но вас вечно нет дома, чтобы почтальону было кому вручить вознаграждение? Что ж, о доставке посылки этот проект может сообщить вам по электронной почте. На выключенном Raspberry Pi присоедините камеру к слоту камеры рядом с портом Ethernet. Далее, соедините свой датчик Passive Infra-Red (PIR) к следующим контактам GPIO вашего Pi. Пожалуйста, учтите: мы используем схему расположения контактов Broadcom.

Загрузите свой Pi и используйте инструмент настройки в меню Preferences. Включите камеру и проверьте, что включен логин с SSH. Перезагрузитесь и затем откройте Python 3 из меню Programming. Создайте новый файл, сохраните и назовите его **emailer.py**. Мы начинаем наш код с импорта ряда библиотек. (Полный список можно посмотреть через ссылку на наш исходный код, и она начинается с `from mail_settings import *`). Они занимаются отправкой электронной почты, фотографированием с камеры и таймингом нашего проекта. Дополнительная библиотека — `mail_settings`. Это внешняя библиотека, написанная сугубо для этого проекта и используемая для хранения имен пользователей и паролей электронной почты. Мы выбрали схему расположения контактов Broadcom, и надо настроить ее:

```
GPIO.setmode(GPIO.BCM)
global file
PIR = 17
GPIO.setup(PIR, GPIO.IN)
```

Теперь создадим две переменные: первая — глобальная, которую можно использовать между функциями, а вторая, с именем PIR, хранит номер контакта, используемого для нашего датчика. Мы настроили нашу PIR на соединение с GPIO17, служащий вводом. Далее создадим две функции: первая делает фото с камеры:

```
def takepic():
    global file
    current_time = str(datetime.datetime.now())
    current_time = current_time[0:19]
    with PiCamera() as camera:
        camera.resolution = (800, 600)
        camera.framerate = 24
        camera.capture((current_time)+'.jpg')
    takepic.file = ((current_time)+'.jpg')
```

Посредством `takepic()` мы фиксируем текущее время и дату в качестве имени файла, и обрезаем строку, хранящуюся в переменной, оставив только нужный нам текст. Далее мы делаем фото и сохраняем его с этим именем файла. Наша вторая функция работает с электронной почтой:

```
def email_send(to,file):
    current_time = str(datetime.datetime.now())
    current_time = current_time[0:19]
    msg = MIMEMultipart()
    msg['Subject'] = 'ВНИМАНИЕ - В '+current_time+' ПРИБЫЛА ПОЧТА'
    msg['From'] = email
    msg['To'] = to
    with open(takepic.file, 'rb') as pic:
        pic = MIMEImage(pic.read())
    msg.attach(pic)
    server = smtplib.SMTP('smtp.gmail.com',587)
    server.starttls()
    server.login(email,password)
    server.ehlo()
    server.send_message(msg)
    server.quit()
```

Тот же метод можно употребить для фиксации времени и даты события. Мы создаем электронное сообщение со смешанным контентом и темой, полученной из строковой переменной события с указанием времени и даты события. Отправитель сообщения генерируется из библиотеки `mail_settings` заказчика. Получатель передается в качестве аргумента в функцию, а наше изображение будет файлом, вложенным в сообщение. Переменная с именем `server` хранит местоположение нашего почтового сервера, в данном случае — учетной записи Gmail. Мы открываем безопасное соединение с сервером, входим и сообщаем серверу, что мы здесь. Затем отправляем сообщение — и закрываем соединение с сервером.

Подготовив все функции, мы используем цикл `while true` для постоянной проверки включения датчика PIR. Если он включается, делается фото, которое прикрепляется к сообщению электронной почты и отправляется получателю. Если датчик не включен, то цикл повторяется. Теперь сохраните свою работу и нажмите на Run > Run Module для запуска.

► Можно разместить этот проект в любом корпусе, лишь бы в прямой видимости от почтового ящика.

Вам нужны...

- Любой Raspberry Pi A+, B+ или Pi 2
- Датчик PIR
- Камера Raspberry Pi
- Приставка Wi-Fi
- Самая свежая версия Raspbian
- Учетная запись Gmail
- Весь код см. на <https://github.com/lesp/LXFPiHome-PostWatch>

## Датчики

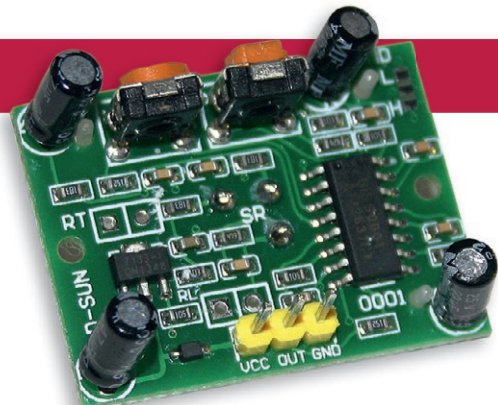
Датчики — это потрясающая возможность автоматического ввода, которую можно применить для запуска событий по движению, звуку, свету и т. д. Raspberry Pi можно соединить со множеством датчиков разного типа.

В этом проекте мы используем простой инфракрасный датчик для определения движения. Он работает посредством подачи тока на Pi при запуске.

Другой пригодный тип датчика — ультразвуковой, он отправляет ультразвуковой

сигнал для определения расстояния до объекта от датчика. Это отличный датчик, но чтобы он заработал, понадобится некоторая математика. И PIR, и ультразвуковой датчики можно найти на eBay дешевле £3.

Напрямую Pi умеет работать только с цифровыми датчиками, поскольку аналоговых GPIO не поддерживает, но менее чем за £10 вы найдете аналого-цифровой преобразователь (АЦП), который решит эту проблему.



► Датчики PIR могут добавить новую форму ввода быстро и легко, благодаря своей простой работе.

# Монитор отопления дома

Визуализируйте свое центральное отопление.

## Вам нужны...



- » Любой Raspberry Pi A+, B+ или Pi 2
- » Свежая ОС Raspbian
- » Датчик DS18B20 (часть CamJam EduKit 2)
- » Макетная плата
- » Кабели-перемычки «папа-мама»
- » Резистор 4,7 кОм
- » Приставка Wi-Fi
- » Учетная запись на [www.initialstate.com](http://www.initialstate.com)
- » Весь код см. на <https://github.com/leap/LXFPi-Home-InitialState>

Для данного проекта мы с головой окунулись в Интернет Вещей (IoT). Будем определять температуру в доме с помощью недорогого датчика, передавать эти данные в облако и на их основании строить диаграмму. Используемый нами датчик — Dallas DS18B20. Его можно добыть относительно дешево, однако простым решением будет приобрести комплект CamJam EduKit 2, поскольку в него входит водонепроницаемый Dallas DS18B20. Соберите оборудование и присоедините к Pi по схеме (см. справа). Далее мы настроим датчик, и для этого есть удобная схема CamJam (<http://bit.ly/CamJamTempWorksheet>). Для работы понадобится учетная запись на [www.initialstate.com](http://www.initialstate.com) и ключ API, который вы найдете в настройках своей учетной записи. Для установки стримера Initial State введите

```
\curl -sSL https://get.initialstate.com/python -o - | sudo bash
```

Начнем наш код с импорта библиотек для работы с ОС и временем, а также для передачи наших данных в облако:

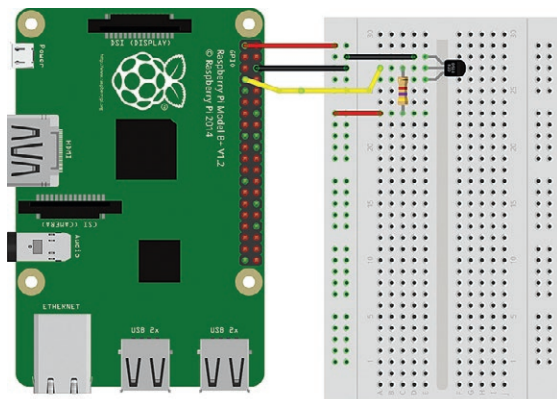
```
import os, glob, time
from ISStreamer.Streamer import Streamer
```

Далее мы загрузим модули ядра для датчика с помощью modprobe, упакуем команды Bash в функцию os.system() для Python и сообщим нашему коду, где найти файл для хранения данных о температуре:

```
os.system('modprobe w1-gpio')
os.system('modprobe w1-therm')
base_dir = '/sys/bus/w1/devices/'
device_folder = glob.glob(base_dir + '28*')[0]
device_file = device_folder + '/w1_slave'
```

Затем создадим функцию для считывания содержимого файла, который хранит исходные данные о температуре и сохраняет данные как переменную:

» Initial State справляется с многочисленными потоками ввода данных.



» Мы присоединили DS18B20 к макетной плате и подали питание на его информационный вывод через резистор 4,7 кОм.

```
def read_temp_raw():
    f = open(device_file, 'r')
    lines = f.readlines()
    f.close()
    return lines
```

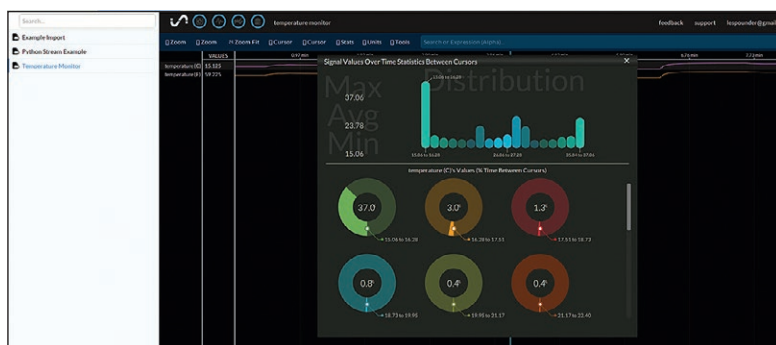
Теперь прочитаем данные и переработаем их в нечто более удобоваримое: сохраним дельную информацию и откинем шелуху, после чего конвертируем данные в температуру.

```
def read_temp():
    lines = read_temp_raw()
    while lines[0].strip()[-3:] != 'YES':
        time.sleep(0.2)
        lines = read_temp_raw()
    equals_pos = lines[1].find('t=')
    if equals_pos != -1:
        temp_string = lines[1][equals_pos+2:]
        temp_c = float(temp_string) / 1000.0
    return temp_c
```

Наш последний раздел — это цикл, который непрерывно проверяет температуру, выполняет преобразования и ежеминутно передает данные в Initial State.

```
while True:
    temp_c = read_temp()
    temp_f = temp_c * 9.0 / 5.0 + 32.0
    streamer.log('temperature (C)', temp_c)
    streamer.log('temperature (F)', temp_f)
    time.sleep(60)
```

Сохраните код и нажмите на Run > Run Module, чтобы запустить его.



## Initial State

В этом проекте мы отправляем температурные данные в облако с помощью сервиса под названием Initial State. Этот сервис позволяет отображать и обрабатывать данные из разных источников практически мгновенно. В этом руководстве мы выбрали бесплатную опцию, с хранением данных 24 часа, а потом они удаляются. Есть и другие опции, в т. ч. с хранением данных неопределенный период времени для неограниченного числа датчиков.

Для нашего проекта мы использовали информацию с одного датчика, DS18B20, но благодаря Raspberry Pi и его GPIO можно собирать информацию по всему дому намного большим количеством датчиков: например, в другом проекте мы использовали геркон. Его тоже можно подружить с Initial State и показывать данные об открывании двери. То есть с помощью этого сервиса можно интерпретировать данные о нашем доме. Такие вещи, как

герконы на окнах; температурные датчики в комнатах; фиксатор на электросчетчике и световые датчики снаружи соберут нам данные о том, насколько энергосберегающим является наш дом, и эти данные можно отображать графически за несколько месяцев, чтобы отследить энергопотребление в разные времена года. Эти данные помогут автоматически управлять системой центрального отопления вашего дома — с помощью скромного Pi!



# Удаленное видеонаблюдение

Следите за своим хозяйством или за домашними животными.

**В** данном проекте мы создадим удаленный монитор для отслеживания активности в вашем доме. Прежде чем начать, убедитесь, что ваша web-камера подключена к вашему Pi. Для обновления нашей системы и установки программы *motion* для web-камеры надо открыть терминал и ввести

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

Установив *motion*, настроим ее с помощью

```
$ sudo nano /etc/default/motion
```

Вы увидите строку `start_motion_daemon=no`; вместо `no` поставьте `yes`. Теперь нажмите `Ctrl+o` для сохранения и `Ctrl+x` для выхода. Затем надо внести пару изменений в файл настройки — **motion.conf**. Откройте его с помощью `$ sudo nano /etc/motion/motion.conf`. Проверьте, что следующие строки правильные, сохраните файл (`Ctrl+o`) и выйдите (`Ctrl+x`) из *nano*:

```
daemon on
width 640
height 480
framerate 100
stream_localhost off
```

Прежде чем продолжить, перезагрузите Raspberry Pi. Пора будет проверить наш стриминг. В терминале введите

```
$ sudo service motion start
```

Потом в браузере на другом компьютере введите IP-адрес своего Raspberry Pi, который вы можете определить через терминал, введя `hostname -I` и затем `:8081`; мой IP-адрес, например, оказался `192.168.0.3:8081`.

Теперь вы должны увидеть в своем браузере видеопоток. А раз поток работает, встроим его в web-страницу в режиме *live*. Для этого надо установить *Apache*. В терминале введите `$ sudo apt-get install apache2 -y`. Заодно создается новая директория в `/var/`, с именем `/www/`, ее мы будем использовать для обслуживания наших страниц.

Откройте на своем Pi текстовый редактор. Напишем несколько строк на HTML, чтобы создать простенькую web-страницу.

```
<!DOCTYPE html>
<html>
<title>Мониторим домашнего любимца</title>
<body>
<xmp theme="cyborg" style="display:none;">
## Что он там вытворяет?

```



» Наше видео легко встроить в web-страницу.

```
</xmp>
</body>
<script src="http://strapdownjs.com/v/0.2/strapdown.js">
</script>
</html>
```

Мы начинаем с объявления документа действующим документом HTML и даем странице название, чтобы определять ее в браузере. Затем переходим к `<body>`, где мы используем среду под названием *strapdown*, которая объединяет *markdown* — популярный формат — со средой запуска *Twitter [bootstrap]*. Вполне приличная страница создается довольно быстро. Мы используем стиль *cyborg*, поскольку он темный и отлично смотрится на устройствах. Для создания заголовка мы используем две решетки (`#`) и затем вводим содержимое заголовка. Далее мы добавляем изображение, источником которого является IP-адрес потока с web-камеры. Чтобы обеспечить соответствие IP-адреса адресу вашего Pi, добавляем в конце `:8081` и даем браузеру команду загрузить файл *JavaScript* с функциональностью *strapdown*. Сохраните файл как **index.html** в своей директории **home**. Откройте терминал и введите следующее, чтобы скопировать файл на наш web-сервер:

```
$ sudo cp /home/pi/index.html /var/www/html/
```

И, наконец, надо запустить наш web-сервер и перезапустить сервис *motion*.

```
$ sudo service apache2 start
$ sudo service motion restart
```

Теперь загляните на IP-адрес своего Raspberry Pi (8081 в конце IP добавлять уже не надо) — и наблюдайте видеопоток с вашего Pi. »

Вам нужны...

- » Любой Raspberry, но лучше всего Pi 2
- » Свежая ОС Raspbian
- » Выход в Интернет
- » Совместимая web-камера

## Видеонаблюдение [CCTV]

Raspberry Pi сделал возможными множество проектов, и один из популярных — видеонаблюдение. Официальная Pi Camera вместе с Pi составят недорогой, высококачественный и потребляющий мало энергии проект, создать который очень быстро. Здесь мы для передачи видеопотока с нашей web-камеры на web-страницу использовали *motion*, но *motion* также пригодно для обнаружения движения и передачи

видео: например, взломщик, ребенок или ваш фокстерьер могут запускать передачу видеопотока на локальное или облачное устройство, где видео и запишется. Добавьте к этому коду датчик *Passive Infra Red (PIR)*, как в нашем проекте наблюдения за доставкой почты, и вы получите мощное приложение, которое будет вас оповещать обо всех инцидентах и регистрировать свидетельства. Еще одно отличное приложение

для использования с web-камерой — *Zoneminder* ([www.zoneminder.com](http://www.zoneminder.com)), оно тоже работает с Raspberry Pi. С помощью *Zoneminder* можно следить за несколькими потоками и настраивать зоны наблюдения, которые будут запускать предупреждение или действие: например, зона около двери сработает, если кто-то войдет в дверь, но не будет проверять на предмет активности окружающую область.



» Проект передает видео через сетевое соединение.

# Автоматический полив

Поливайте цветочки, если прогноз погоды не сулит осадков.

Вам  
нужны...



- » Любый Raspberry Pi A+, B+ или Pi 2
- » Свежая ОС Raspbian
- » Piface Relay Plus
- » 12 В насос Peristatic
- » Питание 12 В 1 А
- » Гнездо для терминала
- » Водный шланг
- » Умение пять
- » Приставка Wi-Fi
- » Учетная запись на [openweathermap.org](http://openweathermap.org)
- » Весь код см. на <https://github.com/lesp/LXFPiHomeGardenManager>

Возможно, для кого-то верхом блаженства будет откопать старую лейку и таскаться с ней по саду, однако это не совсем в стиле XXI века. Кроме того, подумайте о времени на еще один хакерский проект, которое удастся компенсировать. В нашем финальном в этой статье проекте Raspberry Pi мы автоматизируем весь процесс полива с помощью Pi, соединенного с сервисом прогноза погоды и платой-приставкой, соединенной с насосом.

Для начала припаем соединения с клеммами нашего насоса. Соединения можно укрепить с помощью клеевого пистолета или термоусадки. Вам придется использовать больше проводов на клеммах винта резервуара, и обратите внимание на то, где плюс (+), а где минус (-). На Piface Relay Plus найдите реле 3 и вставьте GND (-) питания в терминал COM и в одно из соединений насоса. Найдите терминал NO (Normally Open) и вставьте оба оставшихся провода. Затем надо присоединить плату Piface Relay Plus к вашему Pi и загрузить рабочий стол. Для установки ПО для платы Piface и использования бесплатной карты погоды [openweathermap] с Python 3, откройте *XTerminal* и введите:

```
$ sudo apt-get update && sudo apt-get install
python3-pifacerelayplus
$ sudo pip-3.2 install pyowm
```

Откройте Python 3 IDLE через меню программ и создайте новый файл. Сохраните свой проект как **garden\_manager.py**. Мы начинаем код с импорта библиотек Piface, pyowm и time с помощью `import pifacerelayplus, time, pyowm`. Далее создадим переменную с именем key и сохраним в ней свой ключ API из <http://openweathermap.org>.

Теперь надо создать две функции; первая управляет насосом, присоединенным к Piface. Ее мы назовем `pump` — она принимает один аргумент: долго ли надо поливать сад.

```
def pump(time):
    pfr = pifacerelayplus.PiFaceRelayPlus(pifacerelayplus.RELAY)
    pfr.relays[6].toggle()
    time.sleep(time)
    pfr.relays[6].toggle()
```

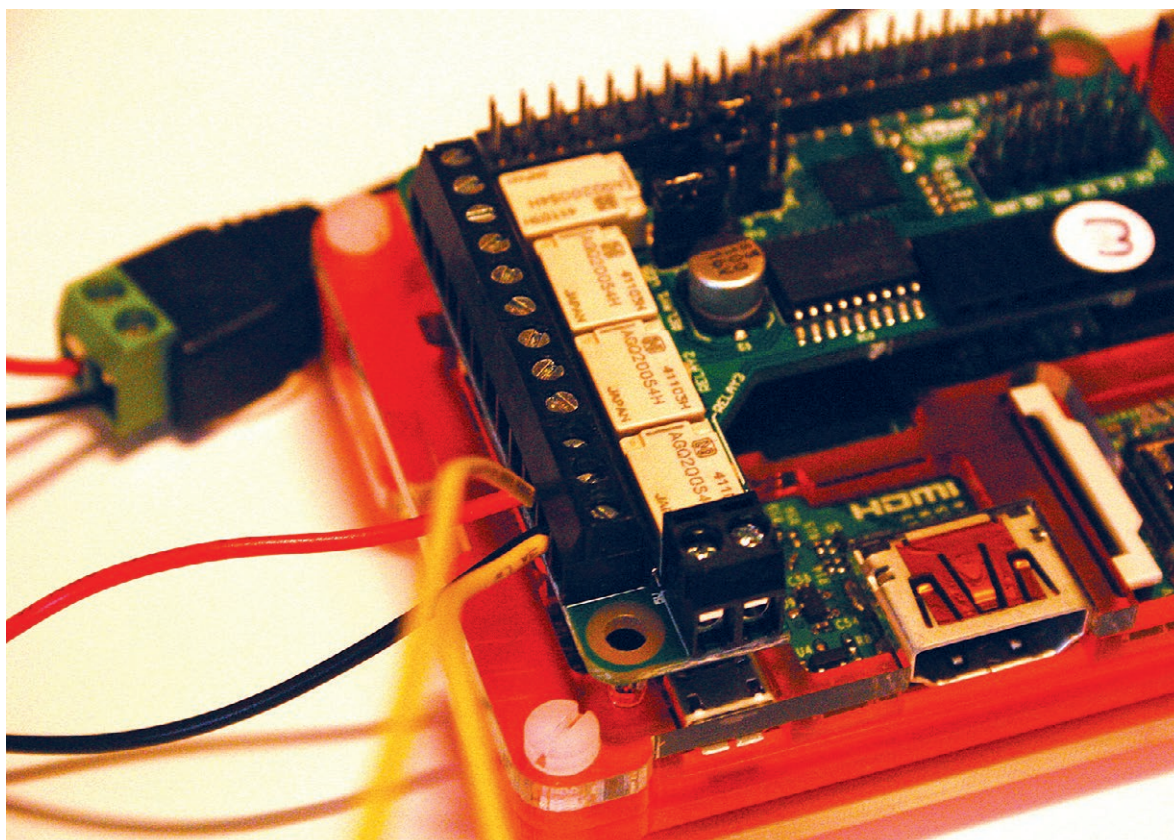
Переменную `pfr` мы используем, чтобы укоротить вызов функции для использования реле. Затем мы включаем или выключаем реле в зависимости от текущего состояния и делаем остановку с помощью `time.sleep()`, позволяя воде течь вплоть до выключения реле.

Наша вторая функция находит прогноз погоды на следующие сутки. Она принимает два аргумента: наше местоположение и количество дней в прогнозе. Затем мы создаем переменную для хранения нашего ключа API `openweathermap`, а следующие две переменные содержат результат функций прогноза погоды [`forecast`].

```
def forecast(x,y):
    owm = pyowm.OWM(key)
    fc = owm.daily_forecast(x,y)
    f = fc.get_forecast()
```

Для итерации данных прогноза погоды мы пользуемся циклом `for`. Эта функция будет уместна для прогноза погоды на несколько дней:

```
for weather in f:
    rain_forecast = str(weather.get_status())
```



» Плата Piface Relay Plus предлагает ряд реле, которые можно контролировать напрямую через Python.



## Работа с высоким напряжением

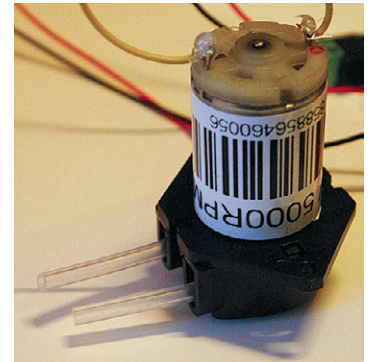
В этом проекте мы использовали 12-В источник питания насоса, но у вас может возникнуть вопрос: зачем тут понадобилось реле?

Дело в том, что Pi не может работать с напряжением выше 5 В, и при более высоком напряжении есть риск повредить GPIO или сам Pi. Реле — это магнитная кнопка, запускаемая электрической цепью, соединенной с Raspberry Pi. Цепь работает с напряжением 5 В и при включении активирует магнит, который

закрывает переключатель внутри реле. Прямое соединение Pi и цепи с высоким напряжением отсутствует, т. е. высокое напряжение безопасно контролируется.

Мы использовали плату Piface Relay Plus с четырьмя реле. Альтернатива — реле на макетной плате, однако для безопасности мы бы советовали использовать на плате напряжение, не превышающее 12 В, поскольку более высокое напряжение требует более надежных решений.

Реле — не единственное решение: для контроля за высоким напряжением пригодны также транзисторы. Транзисторы работают так же, как и реле: они изолируют цепь с высоким напряжением, но ими самими управляет низковольтная цепь. И реле, и транзисторы являются недорогим способом управления проектами с высоким напряжением. Помните, что если вы не уверены в схеме, лучше сперва попросить помощи у знатоков, а уж потом включать питание!



Наконец, функция использует утверждение `if...else` для проверки прогноза. Если дождя не ожидается, эта информация печатается в оболочке, и вызывается функция `pump()`. Если предсказан дождь, эта информация тоже печатается в оболочке, а следующая проверка откладывается на сутки.

```
if rain_forecast != "rain":
    print('Дождя не ожидается')
    pump(300)
    time.sleep(86400)
else:
```

```
print('Предсказан дождь')
time.sleep(86400)
```

И, наконец, надо создать цикл, который будет вызывать функцию `forecast()` для Блэкпула на следующие 24 часа. (Естественно, смените Блэкпул на свое место жительства.)

```
while True:
    forecast('Blackpool,uk',1)
    Как обычно, в этот момент надо сохранить код и нажать Run
    > Run Module для проверки. Для тестирования будет умно задать
    длительность time.sleep() покороче.
```

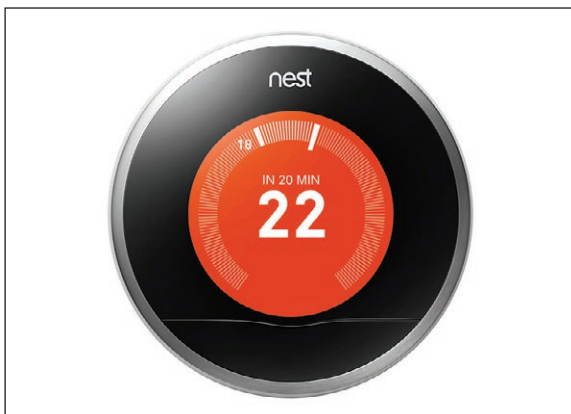
# Едем дальше с Pi

Перед нами целый мир домашней автоматизации.

**Д**омашняя автоматизация охватывает целый ряд продуктов, сервисов и новых концепций. Если вы уже подцепили вирус IoT и жадете расширить свои знания и углубиться в создание домашней автоматизации еще дальше, вот вам несколько идей для дальнейшего исследования.

## Nest

Приобретенный Google в начале 2014 г. за \$3,2 миллиарда, Nest является крупным игроком в области домашней автоматизации. Его



» Nest добавляет центральному отоплению нотку стиля.

линейка продуктов начинается с систем управления центральным отоплением, соединяемых с мобильными устройствами, и охватывает детекторы дыма и угарного газа и IP-камеры. Основная проблема с этими устройствами — их дороговизна: например, обычный детектор дыма/угарного газа стоит £89. Эту немалую сумму вы выкладываете за удобство, и всё уже упаковано в симпатичное и приятное устройство.

## X10

Это протокол для электронных устройств, который использует линии питания для управления и сигнализации. X10 существует с 1975 г., и хотя это не самый новый протокол, у него всё же солидная пользовательская база, в основном благодаря невысокой стоимости компонентов. Кроме того, X10 поддерживается рядом плат, в том числе Arduino и Pi, что позволяет использовать его для управления бытовым оборудованием разными способами.

## Беспроводные штуки Open Pi

Проект Open Pi использует менее знаменитого члена семейства Pi, Compute. Эта модель — более мелкий модуль SODIMM, готовый к встраиванию в проект, и Open Pi помещает этот плоский модуль в маленький пластиковый корпус. Open Pi создан, чтобы хакеры использовали его в разных проектах IoT, используя сочетание Bluetooth Low Energy, инфракрасного ресивера и антенны SRF, которая обеспечивает для устройств, оснащенных SRF, радиосвязь на большом расстоянии.

»

Среди доступных устройств: совместимая с Arduino плата Xi-по RF; дополнение SRF GPIO для Pi и USB-брелок для компьютеров. SRF и его более мощная версия ARF могут осуществлять передачу на гораздо более дальние расстояния, чем стандартные беспроводные устройства IoT.

## Bluetooth LE

Bluetooth уже много лет присутствует в нашей жизни, но недавно мы обнаружили новую версию с низким энергопотреблением, которая предлагает менее мощное соединение для коротких расстояний под названием Bluetooth LE. Они встроены в маячки [beacon] типа Estimote (<http://estimote.com>), которые можно запрограммировать, чтобы они реагировали на устройства Bluetooth изобретательно (или изводяще, в зависимости от точки зрения): например, транслируя соединение Bluetooth, способное передавать данные на ваше устройство. В домашних условиях они могут распознать, что кто-то вернулся с работы, и начать взаимодействие с бытовой техникой через X10, чтобы настроить дом на отдых. Эти маячки можно создать с помощью Raspberry Pi и приставки Bluetooth LE, получив недорогое и не-проприетарное решение.



▶ Bluetooth LE может отправлять вести от объектов в вашем доме, когда вы будете слоняться из комнаты в комнату.

## Kore и Yatse

Один из самых популярных проектов Pi — медиа-центр, особенно после появления Pi 2 в начале 2015 г. Вместо беспроводной клавиатуры и мыши, почему бы не использовать как пульт-лентяйку телефон на Android? *Kore* — официальный дистанционный пульт управления для Kodi, а *Yatse* хоть и не официальное, но мощное приложение.

Оба приложения позволяют осуществлять навигацию по вашей подборке мультимедиа через отлично разработанный и интуитивный интерфейс. *Yatse* также предлагает ряд плагинов для включения управления жестами и отправки SMS-сообщений на ваш телевизор.

## Xbee

Это один из простейших способов автоматизации вашего дома. Xbee использует только четыре соединения — для питания, заземления и передачи данных, и любое устройство может обращаться к Xbee через последовательный канал. Xbee использовался многими умельцами в автоматизации дома на ранних стадиях этого процесса, в частности, для интеграции Arduino, чтобы создать



▶ Любое устройство может обращаться к Xbee через последовательный канал.

## Мобильные приложения для управления вашим домом

Создание собственного интерфейса для проекта — дело хлопотное. Например, надо выбирать язык, среду и протокол. Но общее у всех этих соображений — желательность управления проектом по домашней автоматизации с мобильного устройства. Эти устройства вошли в нашу жизнь, и сейчас вполне обычно для пользователя управлять ТВ, музыкой и освещением со своего планшета или телефона — так же, как и сновать по Сети с помощью этих устройств. Но как управлять нашим проектом домашней автоматизации с мобильного устройства?

### ▶ Создать свое приложение Android

Кодирование приложений Android — трудоемкий процесс, требующий скачивания Android Studio SDK (Software Development Kit) и изучения написания приложений с его помощью. Подробности см. на их сайте — <https://developer.android.com/training/index.html>.

Более простой способ писать приложения Android — применить App Inventor от Массачусетского технологического института (MIT). Он использует среду разработки на базе Web, пригодную для создания структуры и контента проекта и кодирования проекта через блочный интерфейс, похожий

на Scratch. Интерфейс, хотя и выглядит отчасти по-детски, скрывает мощную среду, имеющую доступ к мощи Google.

Наш первый проект использует преобразование речи в текст — это просто делают серверы Google. Проект можно адаптировать для отправки SMS на специальный номер, например, Twilio, который затем можно перенести в Pi, управляющий вашим домом. Это означает, что прямо из офиса вы можете обеспечить, чтобы ваше центральное отопление подогрелось к вашему приходу домой. Узнать больше о MIT App Inventor можно на их официальном сайте: <http://appinventor.mit.edu/explore>.

### ▶ Создать GUI для своего Pi

Недавно Raspberry Pi Foundation выпустили свой новейший продукт, семидюймовый тачскрин, менее чем за £50. Raspberry Pi присоединяется к тыльной стороне экрана, и они могут иметь один источник питания. Интерфейс пользователя для тачскрина можно соорудить на Python, для этого есть масса методов, и два наиболее распространенных — во-первых, среда Tkinter для создания меню и диалоговых окон в стиле традиционных ОС, а во-вторых, создание индивидуального интерфейса с помощью

*pygame*, библиотеки для создания мультимедиа/видеоигр. Например, в Spencer Organ использовали библиотеку *pygame* для создания радиоплеера с индивидуальным интерфейсом пользователя (<http://bit.ly/PiInternetRadioPlayer>).

### ▶ Употребить Flask

Flask — микросреда для web-разработок на Python, которая относительно легко превратит ваш проект в web-приложение, работающее со всеми устройствами через браузер. Flask перекинет мост от Web к вашему проекту, запустив на Pi сервер, который перехватывает информацию с web-страницы, например, гиперссылки или кнопки, и вызывает функцию Python для выполнения действия. Проиллюстрирует это код для управления Energenie с помощью Flask (см. <http://bit.ly/EnergenieFlask>), созданный Беном Наттелом из образовательной команды Raspberry Pi:

```
from flask import Flask, render_template
from energenie import switch_on, switch_off
app = Flask(__name__)
@app.route('/')
def index():
    return render_template('index.html')
```



► Популярная ESP8266, используемая в качестве дешевой платы с соединением Wi-Fi.

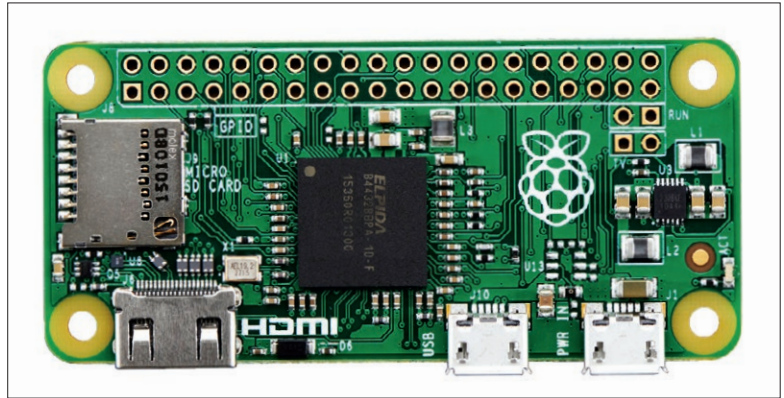
беспроводные устройства, не соединенные с Интернетом и все-таки автоматизированные.

## Particle

Платы Particle впервые появились под названием Spark Core; это совместимые с Arduino платы со встроенным Wi-Fi для управления и программирования издалека. После успешной кампании на Kickstarter команда создала более дешевую новую версию под названием Photon, которая предлагает все функции Spark Core за половинную цену. А самая последняя плата под названием Electron тоже предлагает соединения через сотовую сеть 2G или 3G и позволяет отправлять и получать данные из изолированных мест. Например, Electron можно поместить внутри звонка, где он может отправлять SMS без всяких внешних SMS-провайдеров; так что если вам надо настроить свой проект, находясь на пляже, то вы можете загрузить код напрямую на плату в вашем доме.

## ESP8266

Не будет преувеличением сказать, что плата ESP8266 навсегда изменила домашнюю автоматизацию и IoT. Это дешевая плата с Wi-Fi,



► Raspberry Pi Zero — новое устройство за £4 для создания умных соединенных устройств.

программируемая с помощью Arduino IDE, что позволит обеспечить быструю интеграцию с любым существующим проектом. ESP8266 стала находкой для специалистов по домашней автоматизации, имеющих ограниченный бюджет, поскольку имеются платы разработки, которые обеспечивают доступ к GPIO и программируются на языке скриптов Lua или Micro Python.

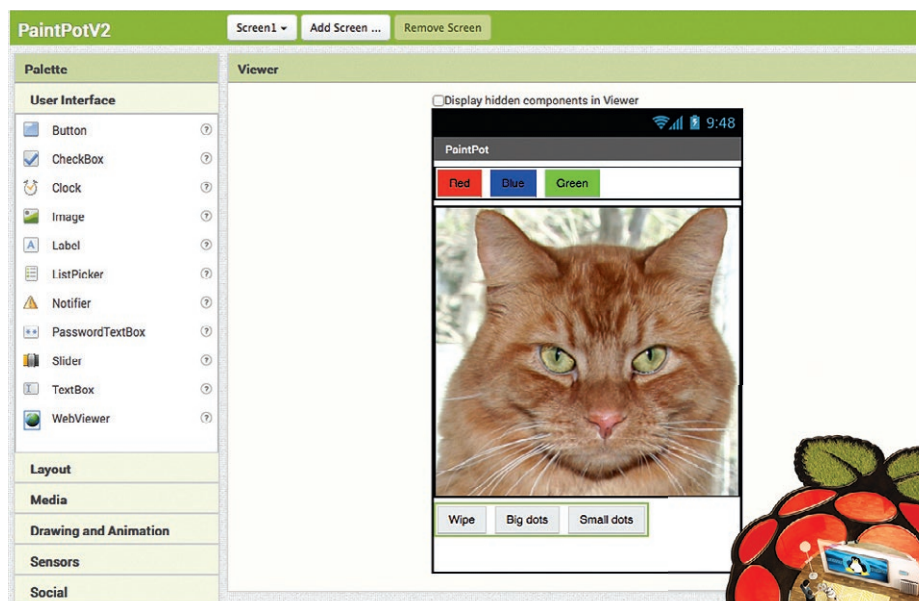
## Raspberry Pi Zero

Foundation снова сумела сделать это — представить версию Raspberry Pi за £4, благодаря менее мощной, урезанной модели Zero (см. полный ее обзор на стр. 18). И это делает Pi главным выбором при создании умных соединенных устройств по всему дому. Поскольку утечка в режиме ожидания снижена до 70 мА (без присоединения HDMI или других периферийных устройств), становится практически даже запитать Pi Zero от батареек AA. Однако Pi всё равно значительно мощнее устройства Arduino, и способен на большее! Pi Zero также поддерживает стандартный GPIO и ПО Linux, благодаря чему запросто можно разработать прототип на стандартных платах Pi, оптимизировать их, а затем развернуть на Zero.

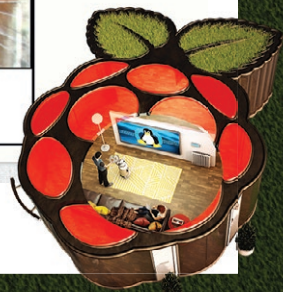
На данном этапе, мы надеемся, у вас возникло достаточно идей, стартовых проектов и технических знаний, чтобы превратить свой дом в автоматизированную нирвану. **LXF**

```
@app.route('/on/')
def on():
    switch_on()
    return render_template('index.html')
@app.route('/off/')
def off():
    switch_off()
    return render_template('index.html')
if __name__ == '__main__':
    app.run(debug=True, host='0.0.0.0')
```

Здесь мы видим, что импортируются библиотеки для Flask и Endergenie и создается экземпляр класса Flask. Далее мы используем декоратор маршрута, чтобы сообщить Flask, какой URL будет запускать наши функции. Мы продолжаем и создаем три функции, которые будут работать с загрузкой шаблона `index.html` и включением и выключением устройств Endergenie в доме. И наконец, мы запускаем приложение Flask в режиме отладки, включаем многословный результат в оболочке Python и настраиваем приложение на прием соединений со всех IP-адресов. Код Python работает с шаблоном HTML, который содержит структуру и контент веб-интерфейса. Можно также использовать CSS для создания стиля веб-страницы.



► Не создать ли приложение для смартфона, чтобы управлять своим умным домашним котом?



# Цифровой океанограф

Джонни Бидвелл встретился с Тэмми Бютов из Digital Ocean, чтобы поговорить о скейтбординге, панк-роке и жизни во втором по величине в мире провайдере хостинга.







**Digital Ocean (DO)** — провайдер облачного хостинга, базирующийся в Нью-Йорке. С момента своего основания в 2011 г. компания демонстрирует феноменальный рост, уступая сейчас лишь AmazonWeb Services (по числу зарегистрированных IP-адресов). Тэмми Бютов (Tammy Bütow), в прошлом сотрудник по борьбе с киберпреступлениями Австралийского национального банка и основательница Ladies Who Linux [Дамы в Linux], теперь возглавляет Службу поддержки Digital Ocean. Мы поймали ее, чтобы побеседовать о том, как живет на краю облака.

**Linux Format:** Чем именно вы занимаетесь в Digital Ocean?

**Тэмми Бютов:** Я возглавляю Службу поддержки [Platform Support Manager] и работаю с командой системных администраторов Linux, а также со специалистами по анализу претензий [trust and safety analysts]. Мы оказываем помощь разработчикам, желающим использовать облако Digital Ocean, и если у них возникают вопросы или проблемы, они могут просто подать нам заявку [ticket]. Заявка поступает к нам, и наша команда работает 24 часа и 7 дней в неделю, гарантируя оперативный ответ. Работаем мы все по удаленке, из разных частей США, так что перекрываем все часовые пояса. Кроме того, все прекрасно разбираются в Linux и легко справляются с решением проблем. Зачастую вы едва успеваете дописать заявку, а мы уже готовы помочь вам дельным советом.

**LXF:** И давно вы этим занимаетесь? Как вы попали в эту сферу? Судя по вашему акценту, вы происходите не из этой части света?

**ТБ:** В Digital Ocean я работаю год. А до этого всю жизнь прожила в Австралии; ну, а потом решила, что хочу заниматься облачной инфраструктурой. Кроме того, меня интересовала работа в стартапе, поэтому я послала резюме в DO, прошла через Интернет все интервью и прочее, и потом перебралась в Нью-Йорк.

**LXF:** Значит, Digital Ocean возник практически из ничего, а теперь внезапно это второй по величине провайдер хостинга в мире. Как такое возможно?

**ТБ:** Да, это довольно круто. Ну, сама я к этому причастна всего один год, но мне кажется, основная причина успеха в его простоте, именно это так нравится разработчикам. В порядке иллюстрации: когда вы попадаете на страницу Droplet Create, чтобы создать сервер, нужно только щелкнуть на желаемом дистрибутиве и выбрать размер, и стоимость рассчитывается очень просто. А если вы впоследствии решите, что он вам уже не нужен, то просто нажмете «удалить», и его не станет. Панель управления очень интуитивна, интерфейс тоже очень понятен. Использовать его — одно удовольствие; потому-то я и хотела здесь работать, что это просто и здорово.

**LXF:** Мне приходилось сталкиваться с парой VPS-провайдеров низкого уровня, порой без всякого удовольствия, и я понимаю, о чем вы говорите.

**ТБ:** Простота — одна из главнейших наших ценностей. Все должно быть таким, чем приятно и легко пользоваться. Я люблю облака и Linux, люблю запускать новые серверы, пробовать новые дистрибутивы. Сама возможность раскрутить сервер и попробовать что-нибудь новое обойдется вам в несколько центов, ведь это недолго. Будь то новый сервер или полноценный новый дистрибутив, вроде CoreOS, например, — это подходящее решение, потому что с ним легко разобраться.

**LXF:** Как вы пришли в технологию?

**ТБ:** Мне всегда это нравилось. У моих родителей был бизнес, связанный с садоводством, ничего общего с компьютерами. Мама занималась на ферме, папы вечно не было дома. Их обоих не особо тянуло к компьютерам, но когда я была маленькая, мама как-то поняла, что за ними будущее и неплохо бы этому научиться. И они выложили кучу денег, чтобы купить мне компьютер, а когда мне исполнилось 12, его подключили к Интернету. Так я оказалась девочкой из Сети, в начале 1990-х, и там было довольно скучно — 12-летним девочкам там было заняться нечем — вот я и решила, что буду создавать сайты для себя и друзей. Я, по сути, участвовала в создании Интернета, и это было здорово. С тех самых пор я всегда хотела работать в технологии,

## О ПОДСАДКЕ НА ТЕХНОЛОГИЮ Я начала участвовать в создании Интернета, это было здорово.

и в университет поступила именно на эту специальность. Далее я пошла обычным путем: получила диплом программиста и стала разработчиком ПО.

Мне всегда хотелось, чтобы в сфере технологии было больше женщин, поскольку, начав здесь работать, я заметила, что нас не так много. Мне было непонятно, почему так: ведь многие мои подруги интересовались технологией. И все они создавали всяческие крутые штуки, так что я начала организовывать разные группы: GirlGeekAcademy.com, @LadiesWhoLinux, где мы собираемся, учимся вместе, и это правда очень здорово — осознавать, как много в мире женщин с техническим складом ума, даже если они не твои соседи по столам.

**LXF:** Полагаю, теперь, имея такую платформу для знакомства и сотрудничества, вам стало проще объединять свои усилия.

**ТБ:** Именно — это своего рода сообщество, которое действительно помогает людям стать увереннее в себе. Что удивительно, в Нью-Йорке особенно много женщин, давным-давно работающих с Linux. Та же Кармен Андо [Carmen Andoh], которая живет здесь и сейчас работает в сфере DevOps [методология разработки ПО, основанная на плотном взаимодействии разработчиков и сисадминов, — прим.



» Бютов основала движение LadiesWhoLinux, чтобы женщины-технари могли держаться вместе.

пер.), имеет более 15 лет опыта и может стать прекрасным наставником.

**LXF:** Linux Kernel Mailing List часто упрекают в недружелюбии, бестактности и грубости.

**Сам Линус даже не считает нужным извиняться за свои вспышки гнева, утверждая, что это в духе проекта. Он противится любым попыткам внедрить «корпоративный псевдо-политес», что довольно справедливо; а лично вам среда разработчиков никогда не казалась агрессивной?**

**ТБ:** Лично я в юности каталась на скейт-борде и слушала панк-рок — с 14 лет я стала спонсируемым райдером, так что росла в скейт-парках, набивая себе шишки каждый день. Но в этом деле всегда хочется добиться большего, и в итоге идешь домой весь в крови. А когда я была не в парках, то пропадала на рок-концертах. И потом, я австралийка, там совсем другая жизнь, чем в США.

**LXF:** Суровая школа жизни, да?

**ТБ:** Именно. Так что это меня не задевает, а если кого-то и задевает, то зря, потому что надо избавляться от предрассудков, если мы хотим быть открытыми для всех. Имею в виду, что сегодня техническое сообщество гораздо более многолико, в культурном, гендерном, социальном плане, это уже факт, и нам пора задуматься, как справляться с таким разнообразием. Как нам добиться, чтобы никто не чувствовал себя в изоляции, когда все такие разные.

Очевидно, что кое-где личные оскорбления и троллинг представляют гораздо большую проблему, но я в Интернете такого не вижу: это обширное пространство, и в подобные места я просто не хожу. Я не сталкиваюсь с этим изо дня в день — просто потому, что меня там нет. Вот как вы ходите в одни рестораны и не ходите в другие, так и я поступаю с некоторыми сайтами. В реальной жизни я бы не осталась там, где мне нагрубили, и зачем



мне быть в таких местах в Интернете? Кто-то думает: «судя по записям на „стене“, девчонок здесь мало» — а возможно, дело в том, что «они не хотят здесь быть». Никто и не будет сидеть на всех подряд сайтах, просто потому, что времени мало.

Когда я переезжала в Штаты, я думала, что здесь всё будет похоже на Австралию, но на самом деле всё совсем иначе — у меня целый год ушел на осознание этих отличий. В том числе то, что воспитание в США совсем не как у нас в Озе. Уверена, что и в Британии оно очень отличается.

**LXF:** Это точно, мы в школе никаких клятв верности не приносили. Разве что, не спорить с тем, что учитель всегда прав.

**ТБ:** Ха. В Австралии, когда мы занимались спортом, мальчики и девочки всегда играли вместе, хоть в крикет, хоть в футбол — нас по гендерному признаку не делили. А в Штатах все наоборот: в классе вы сидите все вместе, а на физкультуру вас вдруг делят. Для меня это так непривычно.

**LXF:** Верно подмечено. А нас в школе заставляли учиться вязать, заниматься гимнастикой и тому подобное. Но давайте забудем о тех черных днях и поговорим о Linux. У вас есть любимый дистрибутив?

**ТБ:** Сейчас я много работаю с Ubuntu, поскольку мы применяем его на работе и наши клиенты, в основном, тоже. Но я пробую всё — сейчас многие в моей команде проходят обучение на сертификат Red Hat, поэтому мы много пользуемся Red Hat Enterprise Linux (RHEL), чтобы побольше о нем узнать. Так что в этом плане я весьма поднаторела, особенно за последние 8 месяцев. Но да, я работаю и с Fedora, и с другими крупными дистрибутивами. Я с большим удовольствием изучала CoreOS, ведь он значительно отличается от других Linux'ов — очень минималистичен. Мне очень нравится заглядывать на [Distrowatch.com](http://Distrowatch.com) и рассматривать все эти дистрибутивы, определять, к какой категории они относятся, замечать, что появляется нового. У меня

в Digital Ocean есть друг Райан, который разработал свой дистрибутив Symphony OS, и занимается им уже 10 лет. По-моему, это круто, что можно создать свой дистрибутив.

**LXF:** Да и работа над ним 10 лет тоже очень впечатляет.

**ТБ:** Да, это правда круто. Вокруг него уже образовалось целое сообщество, и каждый новый релиз очень интенсивно скачивается. Круто и здорово, что есть люди, которые этим занимаются.

**LXF:** Я всегда пытаюсь донести до наших читателей, как полезно разбираться в нескольких дистрибутивах. Здесь уместна параллель с языками: зная несколько, вы лучше понимаете, в чем суть языка как такового. Так и тут, пока вы не посмотрите с позиций разных дистрибутивов, вы не поймете, где заканчивается дистрибутив и начинается Linux.

**ТБ:** Это как раз плюс работы в Digital Ocean: мне следует владеть несколькими дистрибутивами, знать, как они работают и чем отличаются, поскольку наша платформа должна поддерживать все. На данный момент у нас, по-моему, восемь дистрибутивов, и чтобы освоить их досконально, приходится много чему учиться. Совсем недавно мы добавили FreeBSD, и там всё совсем по-другому — эта система ближе к Unix, чем к Linux.

**LXF:** Сегодня создать свой сервер может любой, без труда и больших вложений, а с Digital Ocean это еще проще. А значит, серверов становится все больше, и как следствие, все больше из них,

## ОБ ИНТЕРНЕТ-ТРОЛЛЯХ В реальной жизни я бы не осталась там, где мне нагрубили.

в итоге, взламывают. Как вы справляетесь с тем, что пользователи пренебрегают безопасностью, становясь частью этой статистики?

**ТБ:** На самом деле, этим занимается как раз моя команда, специалисты по анализу претензий. Наша работа во многом связана с обучением людей азам безопасности. В Австралии я тоже занималась расследованием и борьбой с киберпреступлениями: отслеживала вредоносные программы, удаляла фишинговые сайты, так что у меня есть этот опыт. В основном мы учим наших клиентов самым простым вещам — в первую очередь, использовать SSH-ключи вместо паролей. Многие этого не делают, особенно если только начинают осваивать облачную инфраструктуру и раньше не имели дела с ключами; а мы не жалеем времени, чтобы им это объяснить — и в теории, и на практике.

**LXF:** Да, и потом, сгенерировав и авторизовав ключи, важно не забыть сообщить sshd (SSH Daemon), чтобы он впредь не принимал пароли, а то всё пойдет прахом.

**ТБ:** Именно, а также изменить порт, использовать fail2ban и настройки брандмауэра. Если же поступает жалоба на неправильную работу чье-то сервера, мы привязываем ее к их учетной записи, и это служит им предупреждением. Обычно это связано с тем, что их пытаются взломать, и они начинают спрашивать, как же такое случилось. Мы помогаем разобраться с этим, при необходимости уничтожая сервер и по возможности восстанавливая данные с него. Как правило, уничтожения не требуется. Даже в тех случаях, когда кто-то пытается настроить WordPress и устанавливает всякие небезопасные плагины, и много чего способно пойти не так. Но именно потому и важна безопасность, и я, имея опыт в данной сфере, очень об этом забочусь. Думаю, здесь важно как минимум объяснить людям, что если вы создаете сервер с паролем 'password', то...

**LXF:** У вас есть секунд пять, прежде чем вас удалят.

**ТБ:** Примерно так. Это же и вправду плохая идея. Но клиенты частенько спрашивают меня: «А почему я?» И приходится объяснять, что здесь нет ничего личного: никто на вас не охотится, подобные атаки происходят автоматически. Объясняя такие вещи, понимаешь, что для кого-то это новая вселенная. Они знать не знают, что можно кого-то скомпрометировать и взломать, так что обучить их основам безопасности — важнейшая наша задача.

**LXF:** Касаемо образования, я знаю, что вы в этом году дали блестящий мастер-класс по Linux на OSCON. Не могли бы вы немного об этом рассказать?

**ТБ:** Да, это действительно было очень круто, вместе с Джесси Фразель [Jessie Frazelle] (из Docker) и Джорджи Нокс [Georgi Knox] (bit.ly) мы сделали доклад «Новобранцам в Linux: от простого пользователя до мастера ядра». И это было очень здорово, потому что мы начали с самых азов, с того, как работать в Linux, что такое Linux, как управлять Linux. Потом перешли к файлам, процессам, работе с vim. Затем мы поговорили о файловых системах и разделах — что это такое и как их строить. Затем рассмотрели устройство дерева ядра и как создать ядро самим. Было очень интересно; мне нравится узнавать больше о Linux, о том, как он устроен, получать более глубокое понимание.

**LXF:** Я люблю упоминать о том, что когда-то был математиком. И еще в той, прошлой жизни я усвоил, помимо прочего, и то, насколько глубже надо понимать суть предмета, чтобы стать хорошим учителем.

**ТБ:** Абсолютно согласна. Параллельно с техническим я получала и педагогическое образование, и сама год проработала учителем, так что у меня есть опыт в сфере образования, и я люблю учить. А Джесси в университете сначала занималась математикой, потом отдельно информатикой, и теперь работает в Docker. И она довольна своим образованием, все эти навыки критического мышления и решения задач очень помогают во всем. **LXF**

# 7 советов быстрого поиска работы от hh.ru

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

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



# Plasma 5

Джонни Бидвелл приглашает нас в тур по KDE Plasma 5, одному из самых замечательных рабочих столов.

**В**ыход KDE 4 в январе 2008 г. был традиционно встречен шквалом критики. Сперва озбоченность вызвали нестабильность и отсутствие отделки; потом — чрезмерность настройки многочисленных «плазмOIDов [plasmoid]»; и, наконец, нагрузка на системные ресурсы. Конечный продукт кардинально отличался от KDE 3.5, что глубоко потрясло многолетних фанатов-пользователей. Но такая цена прогресса; и KDE 4, без сомнений, удалось наконец вывести среду рабочего стола из эпохи Windows XP.

Переход на инструментарий *Qt4* способствовал появлению более привлекательных приложений, а переход на среду Plasma позволил получить гармоничный рабочий стол с информацией от элементов с современным дизайном. Но, при всех этих

переменах, KDE 4 по большей части придерживался традиционной метафоры рабочего стола с меню и пайджерами приложений и значками системного лотка. Сравните все это со, скажем, Gnome 3, чей минималистский стиль и резкая смена парадигмы вызвали (и продолжает вызывать) критиче-

## Пользователи любого рабочего стола сочтут его привлекательным и интуитивным.

ские бури. Пользователи KDE могут самодовольно развалиться в кресле, пока пользователи рабочего стола-конкурента пытаются разобраться с новым порядком. Вполне резонно будет заявить, что именно недовольство Gnome 3 и привело к появлению рабочих столов Unity, Cinnamon и Mate.

Но теперь KDE больше нет, то есть дело вовсе не в том, что нет нового потрясающего рабочего стола, который пришел на смену KDE 4 [*Ред.*: — Иначе о чем, черт возьми, ты пишешь?] — он есть, но просто уже не называется KDE 5. Его название вы узнаете, прочитав вставку (см. врезку *KDE 5 не существует*, стр. 43). И что это за новинка? Он в высшей степени впечатляет. Он полностью перепи-сан, но у пользователей, последние пять лет работавших с KDE (SC) 4, не будет проблем с переходом. На самом деле, пользователи

любого рабочего стола (даже в проприетарных ОС) сочтут его привлекательным и интуитивным. Более того, поскольку все здесь создавалось по принципу простоты, он как минимум не менее доступен для новичков, чем любой из его конкурентов (включая вас, завитушечники).

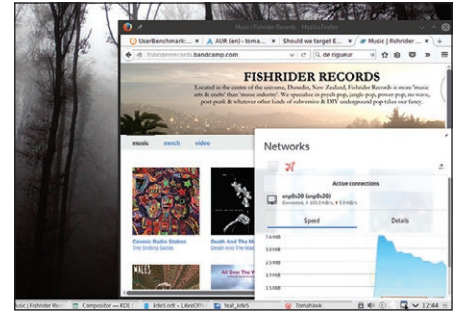


Оставаясь «традиционным» рабочим столом, Plasma 5 выглядит очень современно. Новая тема Breeze предлагает ровный, ясный материальный дизайн, предавший забвению большую часть старого хлама от Plasma 4. KDE 4 часто обвиняют в том, что это громоздкий и раздутый пожиратель памяти; но для его преемника эта критика рассыпается в прах. Да, он очень активно использует композитинг для создания эффектов прозрачности и затухания, и да, все эти прелести и функции означают, что объем потребляемой Plasma памяти выше обычного (около 700 МБ в системе с 6 ГБ ОЗУ), но он всегда быстр и быстро откликается. В частности, когда рабочий стол находится в режиме ожидания, или даже когда вы начинаете неистово перетаскивать окна, фоновая активность CPU крайне мала. Так было на довольно старом Core 2 Duo CPU приблизительно 2006 г., то есть суперсовременное оборудование не требуется. Интерфейс пользователя Plasma создан с помощью среды *QtQuick 2*. Все элементы интерфейса нарисованы на графе сцены OpenGL(ES), благодаря чему большая часть рендеринга (если не весь) производится в GPU. Некоторые эффекты включены по умолчанию: при максимизации окна

растягиваются, при перемещении и изменении их размеров становятся полупрозрачными, а переключение рабочих столов происходит весьма плавно. Кто-то захочет отключить эти эффекты, но многим они действительно полезны — например, при перемещении окна очень удобно видеть, что находится под ним. Есть также и менее полезный, но иногда забавный эффект колышущегося окна, для тех, кого волнует подобная мишура.

## Множество рабочих столов

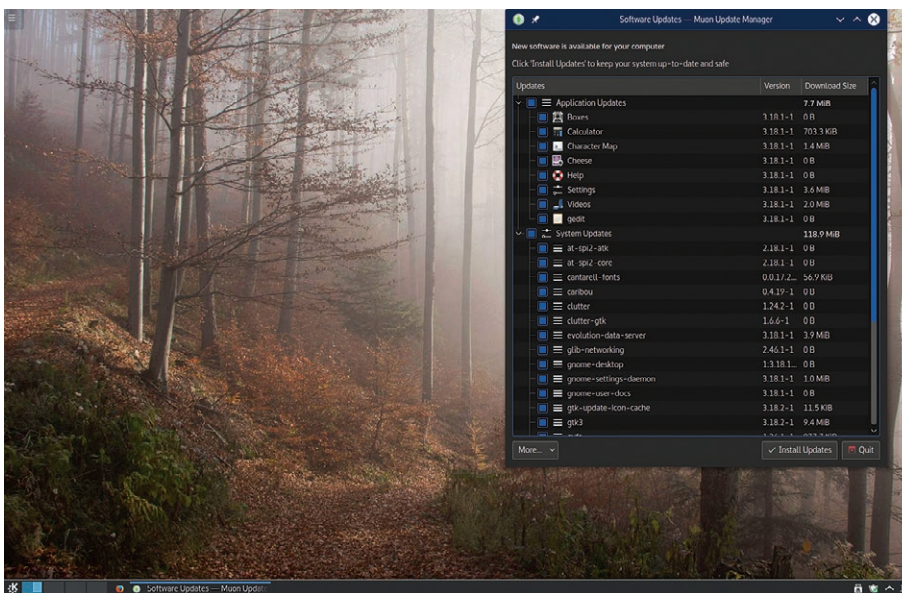
Все любят виртуальные рабочие столы, однако Plasma 5 пошла еще дальше с введением Activities [Комнат]. Назначение новой Activity (например, «работа [work]» или «зависимость от соцсетей [social media addiction]») позволяет вам настроить то, какие приложения открывать и где. Можно настроить параметры конфиденциальности отдельно для каждой Activity, а также создать «страдающую амнезией» Activity, которая не помнит, какие документы вы открывали — или помнит, но только для определенных приложений. Можно настроить комбинации клавиш для быстрого переключения между разными Activity (это бывает невероятно кстати, если вы на работе — и вдруг внезапно входит



► **Апплет Plasma NetworkManager обновлен и намного лучше работает с OpenVPN, плюс поддерживает WPA2-Enterprise. И создает отличные диаграммы.**

начальник). Activity, правда, предлагают довольно неуклюжее решение для тех, кто жаждет иметь разные фоны на всех виртуальных рабочих столах. Очевидно, для этого ограничения существуют технические причины, и несомненно, кто-нибудь вскорости предложит лучшее решение, но все же это огорчает, особенно если учесть, насколько волшебным выглядит все остальное.

Еще один момент, способный досадить новичкам в Plasma 5 — меню по умолчанию Start, именуемое запускателем приложений [launcher]. Среди пунктов разногласий то, что оно неоправданно велико (оно широкое из-за пяти вкладок по горизонтали); что в удобной подсказке *Type to search* [Наберите для поиска] (окно поиска не появляется, пока вы не начнете печатать) имеет место раздражающий эффект прокручивания вашего имени пользователя и дистрибутива (в предположении, что вы способны их забыть); и тот факт, что для активации нижележащих вкладок достаточно подвести к ним курсор, а для открытия в них категории приложения надо жать изо всех сил. Однако Plasma отлично настраивается; немного поискав, вы обнаружите еще два launcher'a: классическое меню с темами и полноэкранную панель управления в стиле Unity/Gnome. Последовав эдикту *Type to search*, вы уже через пару нажатий клавиш увидите список подходящих приложений, недавних документов или web-страниц. Благодаря Baloo (сменившему амбициозный семантический поиск *Nepotik*) вся необходимая индексация для этого вуду происходит за сценой, при минимуме вашего



► **Muon знает все о приложениях Gnome, но ничуть не прячет другие пакеты.**

## KDE 5 не существует

Использование сокращения KDE для рабочего стола стало постепенно прекращаться после версии 4.5, которая вышла еще с двумя буквами, став KDE SC (Software Compilation). В наши дни прозвание KDE переходит на все сообщество, сосредоточенное вокруг этого рабочего стола.

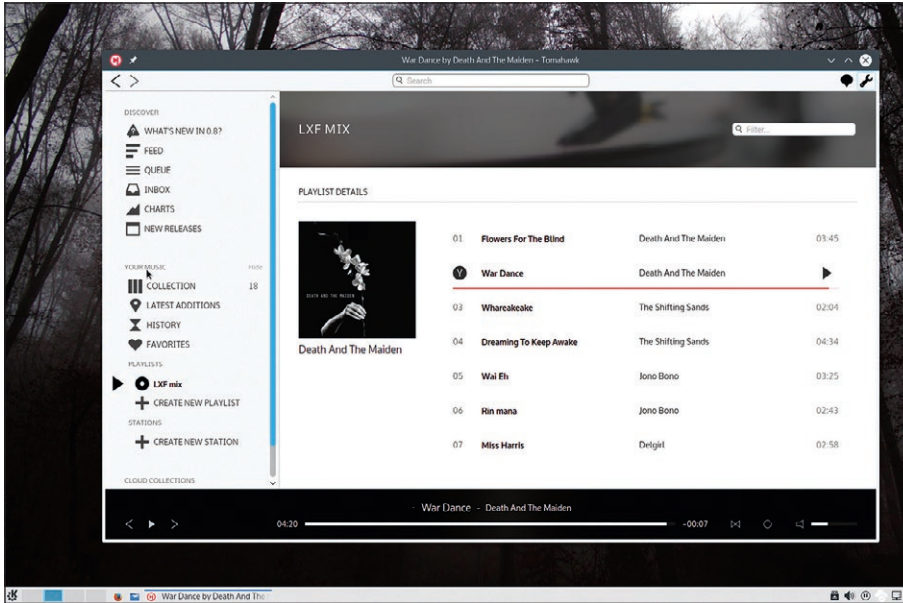
Хотя лежащие в основе этого рабочего стола библиотеки Qt всегда были отдельными от его среды, KDE 4 дал толчок к появлению вспомогательных библиотек (собранных воедино и коллективно именуемых *kdelibs*). Некоторые из них являлись частью самого рабочего стола, а некоторые требовались

только для конкретных приложений. В последней реализации рабочего стола эти библиотеки обновлены и реорганизованы: теперь одни их функции предлагают компоненты Qt, другие присоединены к подборке, именуемой KDE Frameworks (Kf5), а остальные сгруппированы с приложениями, которым они необходимы. Сами приложения составляют пакет под названием *KDE Applications*, а новое рабочее окружение получило известность под именем KDE Plasma 5.

Отделение приложений от рабочего стола — смелый шаг, но он явно послужил к общему благу:

пользователи Plasma могут выбирать, какие приложения устанавливать, а какие нет, тогда как пользователи других рабочих столов могут установить приложение KDE, не втягивая вместе с ним в качестве зависимостей большую часть всего рабочего стола.

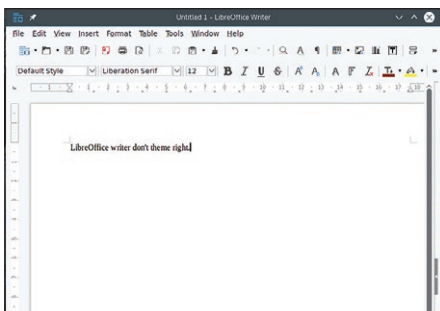
Подобным же образом разделение Frameworks и Plasma позволило *LXQt* стать тем, чем он стал: легковесным рабочим столом на основе Qt5, который использует некоторую часть библиотек Kf5, оставаясь полностью независимым от рабочего стола Plasma.



► **Tomahawk** — многофункциональный музыкальный плеер с опцией для *Qt5* — позволит слушать чудесную музыку антиподов. По редкому совпадению, *Томо-хака* на языке маори означает «танец войны».

участия. Сейчас многие привыкли к подобному типу живого поиска для навигации по рабочему столу. Необходимость тащить тяжкий курсор мыши вниз до упора, в нижний левый угол, и нажать и сделать жест для запуска программы некоторым кажется непосильной задачей. К счастью, есть также *Krunner*, в который в любой момент можно зайти, просто нажав **Alt+пробел**. Откроется минималистичное окно запуска вверху в центре, и вы сможете использовать его так же, как живой поиск в программе запуска приложений.

Если отставить мелочи, совершенство Plasma 5 окажется трудно переоценить; ни наши скриншоты, ни автор, которому постоянно недоплачивают, не в силах воздать ему по заслугам. Отдадим должное усилиям Группы визуального дизайна KDE, сумевшей добиться всего этого в результате открытого демократического процесса. В частности, тема значков *Breeze* — это высший пилотаж: значков там не менее 4780, и все вместе они гарантируют, что панели инструментов и программы запуска вашего приложения всегда будут гармонично выглядеть. *Breeze* использует монохромные значки для действий и контекстных меню, а приложения и папки отображаются в цвете.



► **LibreOffice** не очень вписывается в тему *Breeze*, несмотря на стильные кнопки панели инструментов.

Настройка рабочего стола по умолчанию очень тщательно продумана, чтобы стать как можно удобнее и ненавязчивее. Критику по поводу чрезмерной настраиваемости KDE 4 (настраивается всё) приняли во внимание, но без фанатизма. Многослойные гамбургеры-меню на панели задач и рабочем столе легко скрыть, добавив требуемые виджеты — а выбор виджетов впечатляет: в их числе вдохновленные Vista аналоговые часы и листочки для заме-

## Если отставить мелочи, совершенство Plasma 5 окажется трудно переоценить.

ток. Большинство настроек попали в апплет *System Settings*. Это отличная новость: большинство пользователей KDE 4 сильно злобствовались, прекрасно помня, что некие настройки где-то есть, но чтобы найти их, надо перерыть весь рабочий стол, заглянув в самые дальние уголки и контекстные меню. Правда, пока некоторые опции явно не на месте: например, панель *Desktop Settings* доступна только по щелчку правой кнопкой на рабочем столе, и это также единственное место, где можно снова включить Панель инструментов. И даже внутри апплета *System Settings* некоторые опции похоронены под тремя уровнями категоризации. К счастью, в большинстве случаев имеется разумная иерархия, и вам, по крайней мере, не придется блуждать в лабиринте, как в былые дни.

### Сила трех

Устанавливая четкие границы между рабочим столом, библиотеками и приложениями, команда KDE открыла новый способ восприятия того, где заканчивается рабочий стол и начинаются другие компоненты. В подборке KDE Frameworks 5 мы обнаружили *Valoo* (новый пакет для поиска, индексации и сбора метаданных), *Solid* (среду обнаружения



► Не переживайте, можно залепить рабочий стол вращаемыми виджетами и улучшить продуктивность.

и интеграции оборудования) и *KDED* (демон для обеспечения сервисов системного уровня). Plasma 5 состоит из менеджера окон *Kwin*, темы *Breeze*, приложения настроек системы и т.д. KDE Applications включают менеджер файлов *Dolphin*, PIM-пакет *Kontact* и *Kstars*, программу составления карты звездного неба. Разделение этих трех компонентов также позволяет каждому проекту развиваться более или менее независимо, поэтому KDE Frameworks придерживаются более быстрого ежемесячного цикла, тогда как Applications и Plasma выбрали более консервативный 3-месячный цикл.

То, что эти группы развиваются разными темпами, имеет странноватый побочный эффект: Plasma на момент вашего чтения должна уже достичь версии 5.5, а Frameworks — версии 5.17; но ряд основных приложений по-прежнему сидит на стадии портирования в *Qt5/Kframeworks 5*. Тем не менее, вы все равно можете попробовать Plasma (без блестящих версий *Konqueror* и *Okular* на *Qt5*), не затрагивая своей текущей системы благодаря соответствующему live CD. Например, Plasma 5 предлагают Fedora 23, Ubuntu 15.10 (оба есть на *LXF DVD*) и openSUSE Tumbleweed. В порядке альтернативы, если у вас не установлен KDE 4, большинство дистрибутивов позволяют добавить несколько репозиториями, чтобы получить желаемое. Конечно, более передовые дистрибутивы, вроде Arch и Fedora, включают Plasma 5 как стандарт, а пре-релизные версии приложений на Kf5 можно найти в репозиториях AUR или сорг, но их нельзя считать стабильными. О состоянии портирования всего семейства Applications можно узнать на <http://developer.kde.org/~cfeck/portingstatus.html>, и релиз Applications 15.12 вышел в середине декабря 2015 г., хотя некоторые его составляющие по-прежнему зависят от старого пакета *kdelibs*. Пуристам Frameworks 5 придется тщательно подбирать свои приложения. Почтенный и страдающий проблемой самоидентификации (менеджер файлов это или web-браузер?) *Konqueror* по-прежнему использует старые библиотеки, а вот более новый менеджер файлов *Dolphin* — уже нет.

Интересно будет отметить, что удален менеджер отображения *KDM*. Возможно, на рабочем столе нет шлюза, через который он мог бы проникнуть, или же команде хватало иных забот. В любом

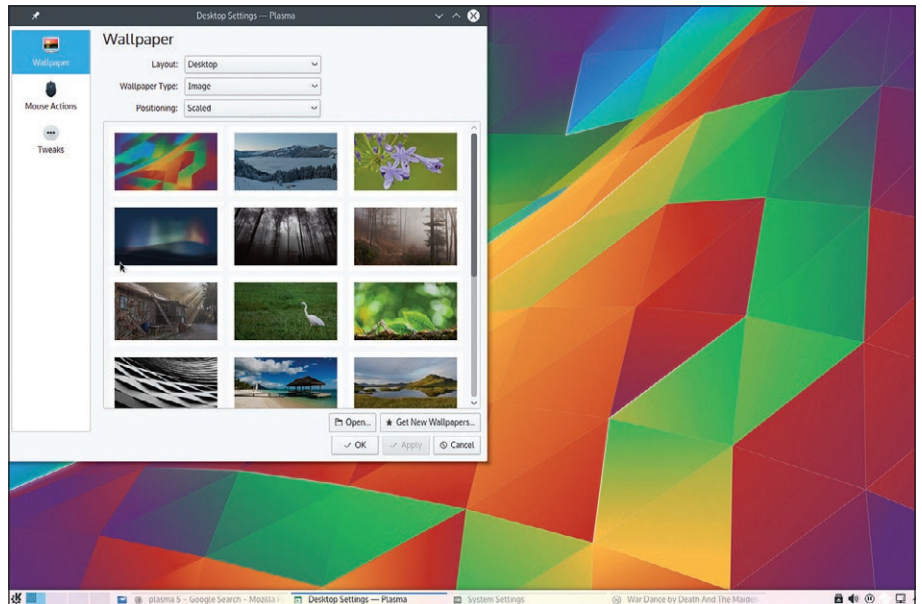
случае, есть множество альтернативных менеджеров отображения. KDE рекомендует Simple Desktop Display Manager (SDDM), который применяет инструмент *Qt5* и даже может взять тему Plasma 5 Breeze. Конечно, можно с тем же успехом употребить *GDM* от Gnome или же *LightDM* (как в Ubuntu), а то и вообще не применять менеджер отображения (обратитесь к *.xinitrc* и используйте *startx*).

После долгих лет насмешек Ubuntu наконец-то ликвидировал свое номинальное приложение *Software Center*, задействовав вместо него более функциональное *Gnome Software*. KDE раньше имел похожий инструмент под названием *Apper*, но от него тоже отказались в пользу *Muon* из Plasma 5. Все эти инструменты работают (или работали) через среду *PackagKit*, которая абстрагируется от лежащего в основе менеджера пакетов, способствуя созданию независимого от дистрибутива GUI для упрощения управления пакетами.

*Muon* — это два приложения: *Muon Discover*, похожий на витрину магазина, и *Muon Updater*, простой инструмент, который сидит в системном лотке, сообщая, какие обновления доступны для установленных пакетов. *Muon* работает с данными *Appstream*, чтобы пользователи могли находить приложения, а не пакеты (концепция которых сложнее в освоении). *Muon* не пытается наступить на пятки вашему достойному менеджеру пакетов: тот работает по-прежнему, и важные транзакции по-прежнему требуют обращения к нему напрямую. Работа *Appstream* всего лишь позволяет производить обновления прямо с рабочего стола, что весьма разумно для современных сред.

## Те же и Wayland

Plasma 5.4 ввела технологический предпросмотр *Wayland*, библиотеки отображения окон следующего поколения, которая когда-нибудь сменит почтенный сервер отображения *X.org*. А пока пользователи могут просто вызвать *Weston* (компилятор ссылок для *Wayland*) внутри окна *X11* и запустить поддерживаемые приложения *KF5* с помощью аргумента `-platform wayland`. Он работает только с драйверами, поддерживающими *KMS* (т.е. не проприетарными), и хоронит *X.org* пока еще рано. Основные усилия по *Wayland* внутри лагеря KDE направляются потребностями *Plasma Mobile*, который вы теперь можете запустить на смартфоне *Nexus 5*, если смелости хватит.



► Похоже, эти яркие треугольники стали фоном по умолчанию. Но есть и множество других, если вы не намерены терзать себе глаза.

Как и для всех современных рабочих столов, требуется наличие аппаратного ускорения 3D-графики. Композитор умеет осуществлять рендеринг через *OpenGL 2.0* или *3.1* или даже *Xrender*, в большей степени опирающийся на CPU. Пользователи более новых карт *Nvidia* сообщают о наличии артефактов разрыва во время воспроизведения видео или игры, но это можно исправить, отключив композитор для полноэкранных окон. У более старого оборудования будут проблемы с *OpenGL*, но вся современная интегрированная графика справится отлично, как и большинство видеокарт середины 2000-х. Так что, возможно, имеет смысл потратить £25 на eBay, если ваш слот *PCI-e* пустует. Глядя в будущее — доступ к контексту *OpenGL* сейчас можно получить через *EGL*, а не *GLX*, при наличии соответствующего драйвера. Это будет существенно для *Wayland*, однако *X.org* останется во всех дистрибутивах как минимум еще год.

Есть немало отличных приложений *Qt*, и многие из них портируются в *Qt5*. Однако рано или поздно вы наткнетесь на еще не портированное. К счастью, довольно просто оформить приложения *Qt4*, чтобы они не смотрелись неуместно. Это почти целиком относится и к приложениям *GTK*. Панель *Settings*

позволяет выбрать тему *GTK*, хотя тему, идеально вписывающуюся в *Breeze*, придется поискать. Традиционно здесь использовалась тема *Oxygen-GTK*, но она уже не поддерживается *GTK3*, и это больше не вариант. Впрочем, есть *Gnome-Breeze* и *Orion* — похожие, но не такие же точно. Тема *Arc* (<https://github.com/horst3180/Arc-theme>) определено похожа на *Breeze* ровностью, и достаточно привлекательна, чтобы простить ей некие неувязки. Вопросы у нас возникли к темам в ряде тяжелых приложений *GTK* (*Firefox*, *LibreOffice* и *Inkscape*), в основном касающиеся шрифтов в панелях меню. Приложения Gnome, такие как *Gedit* и *Files*, выглядели намного симпатичнее.

И это завершает наш рассказ о действительно чудесном рабочем столе (а также его библиотеках и приложениях). Если *Unity* заставила вас скорбеть по горизонтальным панелям задач или *LXQt/Mate* вызвали тоску по всяким украшениям, то это рабочий стол именно для вас. Части *Plasma 5* пока еще в разработке, так что вы можете натолкнуться на недоделанный кусок или же сбой в работе *Kwin*, но это не должно затмить для вас общую картину рабочего стола нового поколения, который не забывает поколения предков. **LXF**

## Конвергенция

Среди критических замечаний в адрес Gnome 3 и Unity чаще всего упоминалось то, что эти рабочие столы навязывают своим пользователям интерфейс, с виду созданный для тачскрина. Конечно, оба этих рабочих стола имеют по крайней мере, разумную поддержку тачскрина (оба поддерживают мультитач), однако пользователи, которые постоянно этим пользуются, явно в меньшинстве.

Plasma 5 тоже предлагает разумную поддержку тачскрина, однако сразу же видно, по крайней мере, в режиме по умолчанию, что он создавался для

работы с традиционными мышью и клавиатурой. И Windows, и Ubuntu есть что сказать по поводу конвергенции — суть в том, что вы берете свой телефон с соответствующей ОС, подключаете монитор и другую периферию, и происходит чудный фокус: ваша ОС трансформируется под дополнительное оборудование.

Plasma 5 в конечном итоге будет поддерживать конвергенцию, но не за счет традиционного способа работы. Большое внимание уделяется разработке *Plasma Active*, мобильного интерфейса на основе

KDE 4, и уже реализуется портирование его на Plasma 5, этим занят проект под названием *Plasma Mobile*, фактически очень тесно сотрудничающий с *Kubuntu*. Как бы то ни было, ни *Windows 10 Mobile*, ни *Ubuntu Touch* по-настоящему не доделаны, и пока эти мобильные платформы не будут готовы, любой разговор о конвергенции с практической точки зрения неактуален.





# По советам м-ра Брауна

**Джоллион Браун**

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

## Ответ телефонным жуликом

Многие читатели *Linux Format* общались нам, что они — или их родственники — получали звонки якобы от «технической поддержки». Я сталкивался с этим сам. Вежливый мужской голос в телефонной трубке, с иностранным акцентом, выражает сожаление по поводу недавних проблем с производительностью вашего компьютера.

Удивленный и немного заинтригованный тем, что это за мошенничество и как оно работает, я постарался подольше удержать собеседника на телефоне. Аккуратно подбирая слова, господин объяснил, что он работает в «Windows», и у меня обнаружены проблемы. Затем он предложил все их «решить» и спросил, «нахожусь ли я сейчас за компьютером». Я решил его провести и притворился, что толком не умею даже включать компьютер, в надежде вытянуть из него побольше информации прежде, чем он потеряет терпение или поймет, что я его надуваю. В конце концов он сказал, что ему пора, и на прощание пожелал счастливого Рождества (в разгар лета).

Хотя читатели этого журнала, несомненно, не ведутся на такое мошенничество и аналогичные вредоносные программы (пользуясь Linux, как и все мы), на прошлой неделе я столкнулся с примером, показавшим, что так поступают не все. Передо мной был компьютер пожилого родственника одного из моих друзей, с цепочкой доказательств, каким образом такие мошенники могут причинить реальную боль тем, кто падет их жертвой. В истории его браузера — список сайтов для денежных переводов, с помощью которых злоумышленникам удалось нагло обокрасть старика, пока тот сидел дома. Встречаясь с членами своей семьи, стоит напомнить им о подобных случаях — и даже, если возможно, рискнуть и перевести их компьютеры на Linux? Внесем свой вклад в то, чтобы уязвимые люди больше не страдали от таких мошенников.  
jolyon.brown@gmail.com

## Эзотерическое системное администрирование из таинственных закоулков серверной.



# Red Hat приобретает Ansible

Гигант открытого ПО упрочил решение для управления конфигурацией, купив за несколько миллионов долларов компанию, стоящую за популярным проектом.

Red Hat в этом месяце заключила важную сделку по приобретению Ansible — компании, стоящей за одноименным продуктом для DevOps и управления конфигурацией. Эта широко обсуждаемая сделка дает Red Hat важный новый компонент в портфолио компании по управлению гибридной конфигурацией.

Компания утверждает, что *Ansible* будет использоваться для помощи клиентам: для развертывания и управления приложениями в публичных и частных облаках, ускорения доставки сервиса, упрощения установок OpenStack и ускорения освоения контейнера. Стоимость сделки на данный момент не раскрыта, но по некоторым оценкам достигает миллиона долларов. В соответствии с FAQ Red Hat по поглощению, у компании, возможно, есть планы по открытию кода всего ПО *Ansible*, включая ранее доступный только по подписке продукт *Ansible Tower*.

Продукт *Ansible Tower* нацелен на корпоративных клиентов и предоставляет интерфейс пользователя для централизованного управления и отслеживания сценариев воспроизведения *Ansible*, а также поддержку доступа, которая основывается

на ролях и отчетах о совместимости (на эту штуку сразу клюют корпоративные клиенты). *Ansible* будет придан импульс для конкуренции с более крупными *Puppet* и *Chef*.

Red Hat подробно описала причины приобретения на своем сайте (<http://www.redhat.com/en/about/blog/why-red-hat-acquired-ansible>). План ее состоит в том, чтобы *CloudForms* (существующее средство для оркестрации и политики компаний) работало как контроллер верхнего уровня, а *Ansible Tower* обрабатывало автоматические развертывания. Продукт *Satellite* от Red Hat продолжит отвечать за развертывание и применение заплаток к нижележащей ОС (при этом останется возможность запуска *Ansible* автономно). Red Hat указала на способность *Ansible* управлять гетерогенными окружениями, включая Windows, сетевые устройства и облачную инфраструктуру на основе Amazon.

Теперь о менее серьезных темах. Популярное сообщество по стримингу игр Twitch анонсировало «гораздо более сложное» дополнение к проекту «Twitch играет в покер» — совместную установку Arch Linux. Каждые пять секунд самая популярная клавиша, нажимаемая в чате, отправляется в консоль транслируемой виртуальной машины. Если вам захочется присоединиться ко всеобщему веселью, то переходите на [www.twitch.tv/twitchinstallsarchlinux](http://www.twitch.tv/twitchinstallsarchlinux), но примите во внимание, что исходная установка была взломана ботнетом-хакемом. Ох уж этот Интернет!



► Red Hat скушала Ansible примерно за 100 миллионов долларов.



# Часть 2: Kubernetes

Джолион Браун сражается с командной строкой *Kubernetes* во второй части своего рассказа о ПО управления контейнерами от Google.

В прошлом месяце я начал рассказывать о *Kubernetes*, системе оркестрации для контейнеров Docker с открытым исходным кодом. На этом уроке я покажу, как запустить и настроить ее в локальной демо-системе. Для простоты демонстрации я сделаю это на узлах *VirtualBox*, запущенных на локальном рабочем столе, но их легко заменить физическими узлами, облачными экземплярами Linux или другой технологией виртуализации, какая вам нравится. Чтобы настройка была простой, я воспользуюсь *Vagrant* — думаю, читатели этой рубрики легко его установят (для этого достаточно установить пакет *vagrant* командами `yum install` или `apt-get` — если не получится, зайдите на <http://vagrantup.com> и следуйте инструкциям).

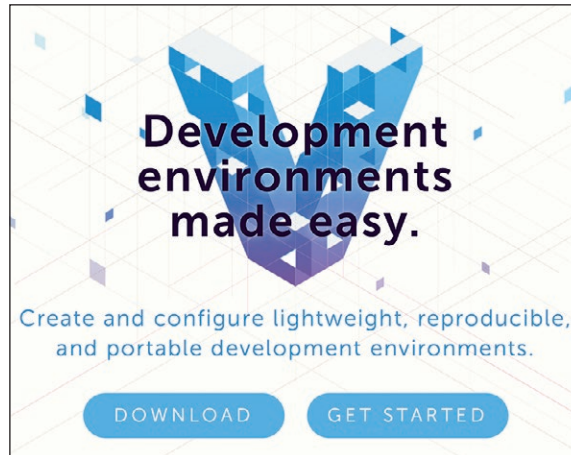
Если у вас установлены *VirtualBox* и *Vagrant*, для установки и запуска *Kubernetes* достаточно выполнить приведенные ниже команды. Но учтите: при выполнении этих команд будет загружен файл *Vagrant* и запустятся виртуальные машины в локальной системе. Я, кстати, ненавижу эту тенденцию вслепую устанавливать что-то из Интернета, а в проектах она встречается все чаще и чаще. Как это возможно? Я бы высмеял администратора Windows, загружающего случайный EXE-файл и дважды щелкающего по нему, но вот эквивалент этих действий для Linux:

```
$ export KUBERNETES_PROVIDER=vagrant
$ curl -sS https://get.k8s.io | bash
```

Впрочем, в данном случае это меня не остановило. Но если вы параноик (а системный администратор обязан им быть), всегда проверяйте такие вещи на тестовой системе, которую легко восстановить в случае сбоя и у которой нет доступа к каким бы то ни было важным данным.

## Быстрый Kubernetes

Тем не менее это самый быстрый способ установки и запуска *Kubernetes*, который я видел. В проекте проделана большая работа, чтобы хорошо упаковать его. *Vagrant*, как я и говорил ранее — отличное средство для быстрого создания прототипов тестовых систем и систем для разработки. Вся настройка через SSH выполняется «прямо из коробки», и это отлично подходит для нашей ситуации. Приведенные выше команды загружают файл *Vagrantfile* и несколько скриптов и код для *Kubernetes*. При запуске *Vagrantfile* первоначально содержит информацию о главном и единственном узле *Kubernetes*. В качестве основы скрипт использует «компьютер» *fedora*. Первоначальная установка займет некоторое время,



► Проект *Kubernetes* включает тестовую систему на основе *Vagrant*, что еще раз доказывает удобство *Vagrant* для быстрого тестирования развертываемых систем.

зависящее от скорости вашего интернет-подключения (но при последующих запусках все будет гораздо быстрее, так как *Vagrant* кэширует «компьютер» локально). Большая часть трудной работы для этой демонстрационной установки выполняется *SaltStack*, единственным решением для управления конфигурацией и *DevOps*, о котором я еще не писал за последний год. Для сети также используется *Open vSwitch*, и он пригодится вам, если вы не сталкивались с ним раньше.

После установки нескольких пакетов *SaltStack* попытается запустить кластер. Здесь моя установка зависла и вошла в цикл (медленно, но постоянно печатая точки в командной строке). Если у вас произошло то же самое, просто нажмите `Ctrl+c` и запустите следующие команды из созданной папки *kubernetes*, которая должна у вас появиться:

```
$.cluster/kube-down.sh
$.cluster/kube-up.sh
```

У меня это сработало; *Vagrant* уничтожил две локальные VM и создал их снова. На второй раз *SaltStack* справился со своей работой, и я получил ободряющее сообщение «Проверка кластера прошла успешно [Cluster validation succeeded]», за которым последовал список запущенных сервисов:

```
Kubernetes master is running at https://10.245.1.2
KubeDNS is running at https://10.245.1.2/api/v1/proxy/
namespaces/kube-system/services/kube-dns
```

»

## Open vSwitch и Kubernetes

В основной статье я упомянул, что *Open vSwitch* настраивается установщиком на основе *Vagrant*. Но что именно он делает? *Open vSwitch* (более известный как просто *OVS*) — открытый, готовый к развертыванию многоуровневый виртуальный переключатель. Он достаточно часто используется в публичных облаках и предназначен для поддержки стандартных протоколов управления, в то же время являясь расширяемым. Он делает именно то, что следует из названия: служит виртуальным переключателем. В нашей установке на основе *Vagrant*

у каждого узла есть мост 'kbr0', используемый вместо более обычного интерфейса 'docker0'.

*OVS* находится перед ним и создает туннели GRE (сокращение от 'Generic Routing Encapsulation' — «инкапсуляция общей маршрутизации») между узлами — фактически это подключения «точка-точка», и каждый узел имеет исходящие туннели ко всем остальным узлам. Хотя это прекрасно для небольших кластеров, как тот, что я здесь настроил, в больших сетевых средах рекомендуется использовать *VxLAN*.

*VxLAN*, как показывает название, использует технологии типа *VLAN* для борьбы с проблемами масштабирования, часто наблюдаемыми в облачных инфраструктурах. *OVS* успешно работает как с GRE-, так и с *VxLAN*-сетями. Учитывая широкое распространение *OVS*, на знакомство с ним стоит потратить некоторое время. Сколько инфраструктур еще не станут облачными (даже в локальных системах) или не будут находиться под влиянием облака через несколько лет? Думаю, что *OVS* займет свое место, что бы ни случилось.

```
KubeUI is running at https://10.245.1.2/api/v1/proxy/namespaces/kube-system/services/kube-ui
```

Похоже, всё хорошо, но каковы имена узлов?

```
$vagrant status
master running (virtualbox)
minion-1 running (virtualbox)
```

Чтобы обратиться к двум узлам, можно выполнить обычную команду `$ vagrant ssh <имяузла>` или воспользоваться любезно предоставленной оберткой для командной строки `kubectl`:

```
$. /cluster/kubectl.sh get nodes
```

Иметь всего один узел слегка позорно, но об этом легко позаботиться с помощью переменной окружения:

```
$ export NUM_MINIONS=3
$. /cluster/kube-up.sh
```

`Vagrant` теперь проснется и убедится, что запущены три узла. Можно запустить сколько угодно дополнительных узлов (или — сколько позволят процессор и ОЗУ вашего ПК).

## Запускаем сервисы онлайн

Когда дело касается *Kubernetes*, я предпочитаю употреблять слово «миньоны [minion]», а не «узлы». Возможно, это не слишком «корпоративно» для целевого рынка (но подумайте о дополнительных возможностях и продвижении, говорю я). Какое бы название ни выбрать (кхе, миньоны), когда эти сервисы сидят, ничего не делая, это не слишком впечатляет. К счастью, в репозитории *Kubernetes* (в каталоге `cluster/examples`) есть масса примеров, которые можно попробовать. Начать проще всего, пожалуй, с `simple-nginx`, который делает именно то, что можно ожидать — и запускает старый добрый HTTP-сервер на простых группах контейнеров.

```
$. /cluster/kubectl.sh 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
```

Впечатляет здесь скорость запуска сервисов. Мы опять же передаем аргументы команде `kubectl`, которую вы могли распознать как относящуюся к `Docker`. Было запущено две группы контейнеров

*Nginx* (загруженных из реестра `Docker`), которые слушают порт 80. *Kubernetes* также создает контроллер репликации под названием `my-nginx`, чтобы гарантировать, что запущенных групп контейнеров всегда было две. Состояние этих групп контейнеров можно проверить командой `get pods`:

```
$. /cluster/kubectl.sh get pods
```

Есть и другие команды, которые можно запустить для просмотра, что происходит с моими миньонами. Обратите внимание, что мне на самом деле все равно, где запущены эти группы контейнеров. Если они решат выйти из облака в Огромный Злой Интернет, я буду сожалеть только о том, что не развернул еще больше их копий, чтобы соответствовать требованиям:

```
$ vagrant ssh minion-1 -c 'sudo docker ps'
```

Эту команду можно запустить для всех моих узлов, с целью посмотреть, какие контейнеры *Docker* на них запущены. Где-то в списке миньонов будут два контейнера *Nginx*, радостно ожидающие трафик. При запуске команды для главного узла будут показаны все его соответствующие процессы; `etcd` для восстановления служб, планировщик, сервер API и менеджер контроллера.

Я могу остановить эти довольно бессмысленные группы контейнеров *Nginx* простым запросом к контроллеру репликации:

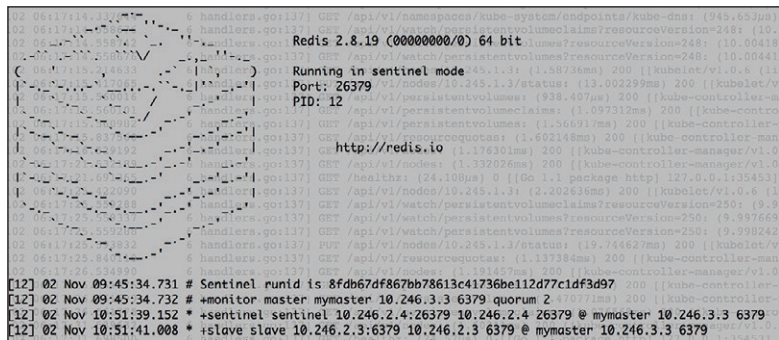
```
$. /cluster/kubectl.sh stop rc my-nginx
```

## Масштабируя высоты

Пока все хорошо. Стоит сравнить работу, предпринятую с *Kubernetes*, с действиями, которые пришлось бы выполнить для разворачивания сервера *Nginx* «старым» способом. Разумеется, мы не рассматриваем конфигурацию самого сервера, но стек *Kubernetes* и его организация очень просты в установке и настройке. А если мы хотим взяться за нечто посложнее и затем расширить это, как удастся Google со своими онлайн-сервисами? Еще один пример установок — хорошая способ иллюстрации того, как это делается. Ранее в этой рубрике я рассказывал о *Redis* — хранилище структур данных в памяти, используемом как база данных, кэш и брокер сообщений. Именно такое и должно быть высокодоступно, а также масштабироваться по запросу. Представьте, что ваш `web-сервис` с поддержкой *Redis* вдруг попадет на главную страницу `Reddit` или о нем расскажет какая-нибудь знаменитость с миллионами подписчиков на `Twitter`? Это, безусловно, приятная проблема, но она также может стать рubeжом, который сервис успешно преодолет или будет отброшен обратно в неизвестность. Но не бойтесь. *Kubernetes* вам поможет!

Тут стоит взглянуть на файлы `YAML` в каталоге `examples/redis`. Они показывают, как спроектировать элементы группы контейнеров или службы (после этого, разумеется, их можно добавить в систему контроля версий), которые можно передать `kubectl` и успокоиться. Первый файл, на который стоит взглянуть — `redis-master.yaml`, в нем определяются два контейнера в группах контейнеров: основной узел *Redis* и соответствующий «страж». Это часть

С помощью *Redis* удобнее всего проверить, как хорошо *Kubernetes* выполняет масштабирование; да он и сам по себе весьма впечатляет.



## Kubernetes и лог-файлы

Думали ли вы, пока читали эту статью, о том, что просматривать лог-файлы из сервиса, запущенного в группах контейнеров — пустая трата времени и расход лишних ресурсов? Если да, 10 очков гуру Linux вам в копилку: это действительно трата времени и расход лишних ресурсов.

При первом запуске *Docker* важно подумать о том, как вести журналы. Очевидно, что хранение лог-файлов локально на хостах с контейнерами едва ли решает проблему. Это еще актуальнее для *Kubernetes*, вся задача которого состоит в том,

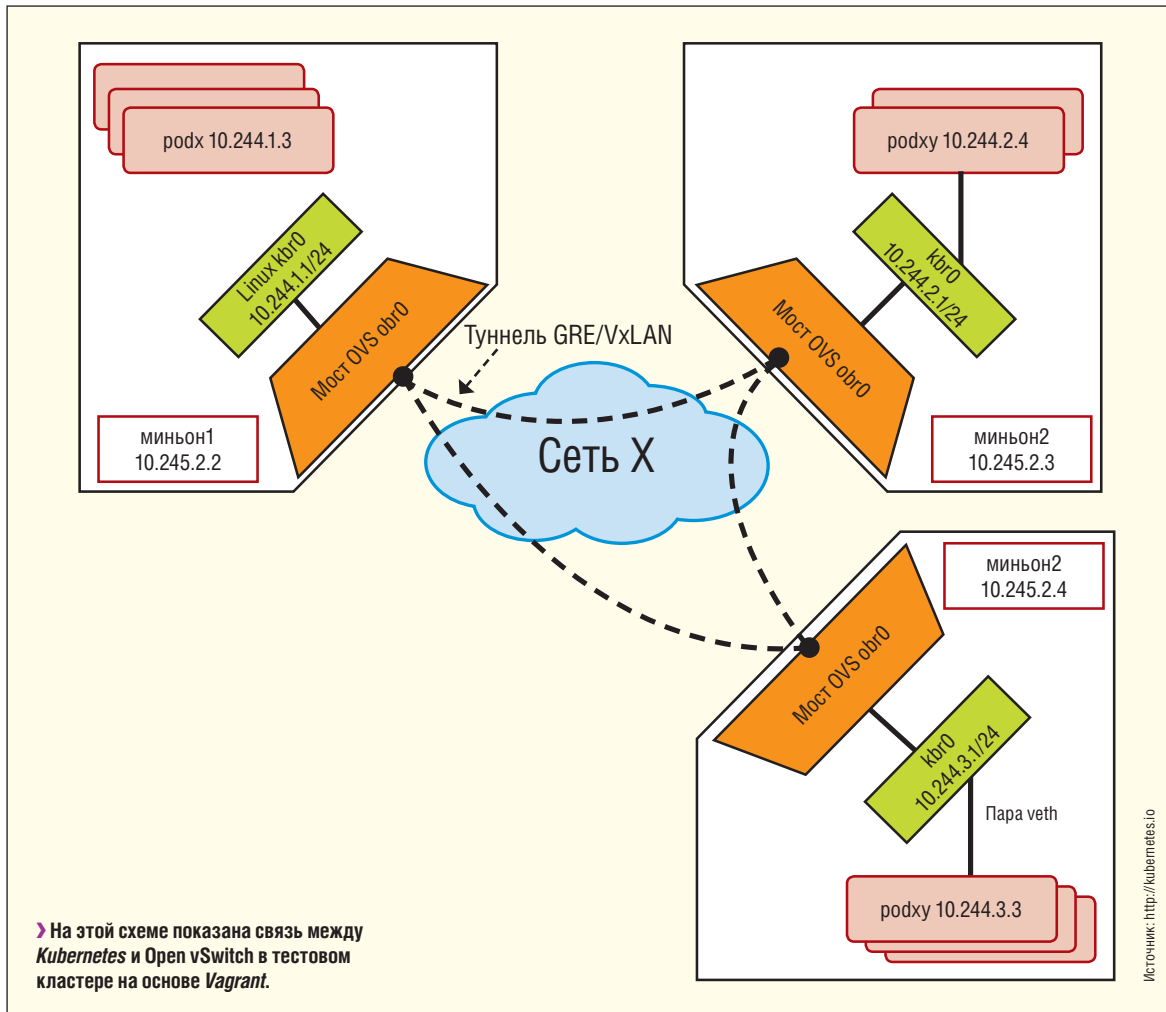
чтобы системный администратор даже не задумывался, где реально разворачиваются его группы контейнеров.

Лучший способ решить эту проблему — настроить централизованный сервер журналирования и сбрасывать лог-файлы туда. Такой сервер соответствует более традиционному серверу `'syslog'` в Linux, где находится `syslog`, и другие записи отправляются в него (это по-прежнему хорошая идея).

Однако старую схему `rsyslogd` превзошли программы вроде *Elasticsearch*. Эта программа,

о которой я расскажу попозже, по сути представляет собой полнотекстовую поисковую машину, написанную на Java. *Kibana*, которая часто используется вместе с ней, предоставляет аналитику и визуализацию для индексов *Elasticsearch*. Код обеих программ открыт.

Хотя для данного урока мне не удалось настроить стек ELK — это сокращение от *Elasticsearch*, *Logstash* (обрабатывает пересылку лог-файлов) и *Kibana* — такая схема очень распространена, и на нее стоит взглянуть.



стека *Redis*, которая обеспечивает здоровую проверку работоспособности и обработку отказа. Ее можно запустить так:

```
$ ./cluster/kubectl.sh create -f examples/redis/redis-master.Yaml
```

На запуск потребуется время — группа контейнеров некоторое время будет находиться в «подвешенном [pending]» состоянии по мере загрузки образов *Docker* из открытого хаба *Docker*. Когда группа контейнеров перейдет в состояние «запущено [running]», я смогу просмотреть лог-файлы, созданные контейнером *Redis*. Первый аргумент команды `logs` — это имя группы контейнеров; второй — контейнер, лог-файлы для которого я хочу просмотреть (если они есть).

```
$ ./cluster/kubectl.sh logs redis-master master
```

Я должен увидеть знакомый лог-файл запуска *Redis*, который означает, что сервер готов к получению трафика на порт 6379. Теперь мы можем определить сервис. Он позволит определять «стражей» *Redis* с помощью API *Kubernetes*. В приведенном примере сервис определяется файлом `redis-sentinel-service.yaml`. Запустить этот сервис можно с помощью той же команды, которая была указана выше. Также с помощью команды `get` можно получить список сервисов (`get services`). Однако сервис *Redis* пока не обладает отказоустойчивостью. Чтобы ее получить, надо создать контроллеры репликации, которые, как я рассказывал вам месяц назад, следят за тем, чтобы было запущено достаточно групп контейнеров. В каталоге `example` есть два файла, определяющих такие контроллеры — один для *Redis* и один для «стража»: `rediscontroller.yaml` и `reis-sentinel-controller.yaml`. После их создания (опять же с помощью команды, указанной выше) можно получить их список командой `rc` (запустите `kubctl.sh get` без аргументов, чтобы узнать, какие еще списки можно получить).

Теперь я могу масштабировать службы *Redis* простой командой

```
$ ./cluster/kubectl.sh scale rc redis --replicas=3
```

```
$ ./cluster/kubectl.sh scale rc redis-sentinel replicas=3
```

*Kubernetes* не слишком многословен в комментариях о завершении этой задачи (об успехе свидетельствует единственное слово «отмасштабировано [scaled]»), но команда `get` покажет новые запущенные группы контейнеров. Для получения дополнительной информации иногда удобно запустить команду `get` с параметром `-o json` (например, `kubectl.sh get -o json pods`). Снова взглянув на лог-файлы основного узла *Redis*, я вижу, что запросы синхронизации пришли и были завершены. При добавлении новых реплик в этих лог-файлах появятся дополнительные записи. Теперь я проверю обработку отказа, «убив» один из узлов *Vagrant* и установив кластер. После этого группы контейнеров *Redis* должны перезапуститься.

```
$ vagrant ssh minion-3
```

```
$ sudo shutdown -h now
```

```
$ ./cluster/kube-up.sh
```

```
$ ./cluster/kubectl.sh get pods
```

Призываю читателей ознакомиться с другими примерами, которые входят в эту установку на основе *Vagrant*. Есть несколько примеров, покрывающих диапазон действий *Kubernetes*: например, из-за нехватки места мы не коснулись темы персистентных хранилищ, а также ряда довольно сложных задач по построению приложений и инфраструктуры. Мне ясно, что простота разворачивания кластера *Kubernetes* — хотя рабочая система, конечно, будет гораздо сложнее по сравнению со всем, что мы здесь рассмотрели — означает, что это сильный игрок в быстро меняющемся мире управления контейнерами и облачной инфраструктурой. **LXF**

## Следственные полномочия

**К**ак меняются времена... Раньше меня заставляли нетерпеливо обновляющим web-страницы для запуска фестиваля или билетов на джаз; а недавно я нажал кнопку обновления в ожидании выхода проекта программного госдокумента, чтобы прочесть проект Закона о следственных полномочиях [Draft Investigatory Powers Bill] (см. <http://bit.ly/DraftIPBill>). Он был опубликован при большой огласке (и, надо сказать, с необычным количеством предварительных брифингов для прессы). Двадцатилетняя версия меня пришла бы в ужас! Как оказалось, этот документ очень велик: в нем 299 страниц, плюс 26 приложений; их суть не опишешь в 300 словах, отведенных на колонку!

В Интернете, естественно, полно аналитических материалов, и ожидается, что законопроект станет законом в конце этого года, пройдя все необходимые этапы (комиссии и т.п.). Все могут высказаться по данному закону, пока его проводят через Парламент, и лично мне кажется, что это стоит сделать независимо от ваших политических взглядов. Меня особо интересовало, насколько тяжким бременем такой закон ляжет на поставщиков услуг и какие из обсуждений о регламентации шифрования, при их наличии, могли проникнуть в опубликованный документ. Но весьма интересны и разделы о «повсеместном вмешательстве в работу оборудования [bulk equipment interference]» (по сути, узаконении взлома) и многочисленных цензурных запретах. Сисадминам ставится задача обеспечить безопасность систем и их устойчивость к атакам. В этом законе многое способно повлиять на наше будущее (например, могут возникнуть ситуации, когда его обсуждение с кем-то приведет вас в тюрьму).

Надо быть в курсе происходящего с этим законопроектом, и я настоятельно рекомендую вам почитать о нем.

# Вышли Docker 1.9 и Swarm 1.0

## Кузов контейнеров пополняется готовым для индустрии *Swarm* и многохостовыми сетями.

**П**еред конференцией DockerCon EU, проведенной в Барселоне в ноябре 2015 г., Docker выпустил несколько радующих широкие массы обновлений своего продукта, который продолжает быть в центре внимания в области Infrastructure/DevOps.

Версия 1.9 контейнерной технологии включает готовый для производства релиз многохостовой сети, обкатка которого прошла в предыдущем релизе. Он позволяет создавать виртуальные сети на нескольких хостах. Это значительно упрощает пользователю полный контроль над топологией сети, и контейнеры могут обращаться друг к другу. Это названо «сетью, определяемой ПО» для контейнеров, и позволяет выгружать лежащий в основе драйвер VXLAN для соответствия нуждам отдельного сайта.

Поступали также новости о полной перестройке системы томов для хранения постоянных данных приложений, что позволяет использовать плагины вроде Serf — большой шаг вперед относительно предыдущих решений.

Помимо этого, Docker анонсировал версию 1.0 своего продукта *Swarm*, предлагающего «родной»

кластеринг для Docker Engine. Это довольно похоже на другие проекты (например, *Mesos* и *Kubernetes*), но с преимуществом в виде использования API Docker. Потенциально это упрощает разработчикам масштабирование их приложений посредством тех же самых постоянных вызовов с рабочего стола в среду, предоставленную большим облаком для хранения кластеров *Swarm*. Docker выпустил подробную информацию о своей работе с 30 000 контейнеров на 1000 хостов, но если необходимы более мощные серверные решения, можно загрузить и такие (для исключительно крупномасштабного производства).

Тем временем OpenStack выпустил еще один релиз (12-й по счету, с кодовым именем Liberty), включивший, помимо массы других функций, улучшенную интеграцию с Docker и *Kubernetes* (в том, что, судя по всему, является постоянно растущим семейством проектов).

Наконец, мы в **LXF** опечалены смертью Тельсы Гуинн [Telsa Gywnne]. Тельса была известным участником множества сообществ открытого кода и членом Совета директоров в GNOME Foundation. Выражаем соболезнования ее семье.



► На DockerCon EU *Swarm* продемонстрировал масштабирование до 50 000 контейнеров — почти столько же, сколько у Джозуа в *Minecraft*.

## 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 тысячи клиентов



+7 812 309 0686

WWW.LINUXCENTER.RU/SHOP/MY\_SQL ● INFO@LINUXCENTER.RU

# Стек ELK

Хватит входить в свою систему через SSH; управляйте журналами с помощью кластера ELK (*Elasticsearch, Logstash, Kibana*).

**К**огда дело доходит до проекта, требующего создания новой инфраструктуры, мне нравится рассматривать сценарий «3 утра». Это когда дежурного (им могу быть и я) будит операционная команда или автоматическое предупреждение и сообщает о проблеме, требующей внимания. Довольно часто подобные беседы коротки, и их получатель, скорее всего, с первого раза не проявит должного понимания. Вопрос в том, быстро ли сможет этот человек понять, к чему относится данный звонок, выяснить, что за проблема возникла, и затем либо начнет поиски ее решения, либо перезвонит для выяснения, что же делать дальше. Или, иными словами, когда я смогу снова улечься спать?

Первой части этого вопроса может и не возникнуть, если как следует продумать мониторинг и ограничить то, что способно спровоцировать звонок. По моему мнению, система может осмелиться пробудить меня от моего чудесного сна — и уж поверьте, сон мне очень нужен — только при действительно срочной проблеме, требующей немедленных действий. Если меня будят ради того, что можно было бы отложить до утра, я буду а) неоправданно усталым на следующий день и б) очень злым. Я дежурил многие годы (и по-прежнему дежурю — правда, гораздо меньше), и героизм, на который вы считаете себя способным, будучи двадцатилетним одиночкой, куда менее привлекателен, когда вам уже за тридцать и по утрам надо отвозить детей в школу. Кстати, если по работе ваше расписание вызовов гарантирует вам жуткую ночь/неделю/месяц с постоянными перебоями сна — завязывайте с этим. Прямо сейчас. Оно того не стоит и предполагает, что либо вы следите за тем, что вряд ли приведет к проблеме, либо ваша инфраструктура/приложение настолько плоха/о, что требует доставки в палату интенсивной терапии. Скажите всей команде, чтобы они притормозили и рассмотрели список вызовов (если у вас нет этого списка, начните его вести). Выясните, что порождает наибольшую головную боль, определите настоящую причину и разберитесь с ней. Прополощите и повторите со вторым пунктом списка. Доводить себя до изнурения из-за постоянного недосыпа — не шутки.

Возвращаясь к сценарию, где сисадмин сидит в пижаме, проклиная команду разработки, компанию хостинга или ISP (т.е. виновников его выползания из кровати) — возможно, из сигнала тревоги уже станет ясно, что делать: например, кончина процесса привела к отказу сервиса; могла переполниться файловая система, и т.д. Однако на случай чего-то нетривиального — а вас не должны дергать по поводу тривиальных вещей, если вы автоматизируете

восстановление и встроите в свой сервис избыточность — вероятно, нужно будет взглянуть на некоторые журналы. Информация, которую мне нужно будет допросить, может являться системным журналом или журналом, генерируемым приложением; но для всего, кроме самых основных сервисов, лог-файлы будут создаваться в самых разных местах.

Последнее, что мне захочется сделать в 3 часа утра — это лезть вручную через SSH в группу разных экземпляров Linux и приниматься запускать команды `less` и `grep`. В зависимости от типа используемой инфраструктуры, я, возможно, попытаюсь отследить ошибки по нескольким web-серверам. Может быть не совсем ясно, у кого именно или у какой группы возникли проблемы; возможно, понадобятся логи с перекрестными ссылками на логи из сервисов приложений среднего уровня. Хуже того, учитывая тенденцию к использованию архитектуры микросервисов, мне, возможно, придется возиться с десятками систем или даже, потенциально, сотнями контейнеров!

В былые дни монолитных и многоуровневых архитектур было (и по-прежнему остается) обычным делом, а также хорошей практикой обеспечения безопасности держать центральный сервер как `syslog`, на который клиентские системы сливают логи (вероятно, используя `rsyslog` и UDP). В наши дни реализация подобных настроек является голым минимумом, который я бы сделал хотя бы для обеспечения безопасности копий live-логов для аудита. Есть ряд опций для сливания логов в пункты назначения `'write once'`, которые варьируются от: дешевых и прикольных до агрегаторов логов корпоративного уровня (т.е. дорогих).

По крайней мере, с подобным подходом я могу искать проблемы в одном пункте — но это по-прежнему означает необходимость вручную продирааться сквозь логи. В одной системе, над которой я работал пару лет назад и которая следовала этой модели, команда постепенно создала набор команд и скриптов для быстрого извлечения информации из объединенных файлов, но все равно слишком часто не оставалось ничего другого, как продирааться через выводы нескольких команд `egrep` и `awk`, собранных вместе.

Я знаю, что вы подумали: «Э, он пытается предложить нам лучший способ работы», не так ли? Именно! Ободритесь, сисадмины! Оставьте Темные Века позади и изучайте свои журналы в совре-



» Логотип *Logstash* вполне уместно смотрелся бы в мультике.

## Могу я заплатить кому-то, чтобы все это сделали за меня?

Компания Elastic (до марта 2015 г. — Elasticsearch) — разработчик стека ELK. Как и многие компании в сфере открытого кода, они предлагают ряд контрактов по платной поддержке своего ПО, плюс другие типы «корпоративных» соглашений для ряда дополнительных программ. Конечно же, солидный аргумент для заключения контракта — то, что его предоставляют люди, написавшие большую часть кода этих продуктов. При самом высоком уровне поддержки (и цене на приложение) они срочно предоставят отладки ошибок. Получив приличное

финансирование, компания так расширилась, что смогла предложить хостинг на основе облака для структур ELK — т.е. «управляемое» решение. Коммерческие предложения компании включают Shield (шифрование, доступ на базе ролей, фильтрация IP и аудит для стека ELK), Watcher (позволяющий ELK обеспечивать, скажем, возможности мониторинга, основанные на аномалиях в данных) и Marvel (дополнительные инструменты для исследования статуса развертывания *Elasticsearch*, возможностей аудита и оптимизации/тонкой настройки). Возможна даже

схема именования в честь супергероев! Проекты с открытым кодом от этой компании (их немало) см. на <https://github.com/elastic>.

Как всегда с открытым ПО, поддержке стека ELK есть альтернативы. Так, поддержку специалистов и хостинг предлагает `Logz.io`; а у Amazon есть свой «Elasticsearch Service», способный ставить кластер *Elasticsearch* прямо из консоли AWS, с обычными прелестями продуктов AWS — автозамещением сбойных узлов, простотой масштабирования и т.д., что поможет заместить «наземные» установки ELK.

менном стиле! В этой и последующих статьях я намерен рассмотреть то, что известно как стек ELK (что означает *Elasticsearch*, *Logstash* и *Kibana*). Эти три компонента составляют действительно мощный инструмент для анализа данных, создаваемых всеми типами систем. Прежде всего, рассмотрим, что каждый элемент стека может предложить.

*Logstash* собирает, обрабатывает и передает логи (и другие типы данных). Проект может похвастаться тем, что умеет «обрабатывать любые данные из любого источника». В подкрепление этого имеется более 200 доступных интеграций для данной программы, позволяющих *Logstash* работать со всеми видами источников данных. *Logstash* написан на Java и требует для работы установленной JRE (версии 7 и новее).

Помимо первичной обработки логов, *Logstash* способен работать с данными по мере их передачи к пункту назначения через нечто, известное как «каналы». Каналы по сути состоят из плагинов ввода, фильтра и вывода — они расщепляют данные, преобразуя их в форму, которая может храниться в *Elasticsearch* (или применяться для иных целей — но нас на этом уроке интересует только стек ELK).

*Elasticsearch* — это конечный пункт высадки для наших обработанных данных в этой структуре. Он тоже написан на Java и, на самом деле, основан на популярном (и достопопеченном) движке поиска и индексирования Java — *Lucene*. Это высоко масштабируемый полнотекстовый поисковый и аналитический движок. Хотя он и способен на многое, но в данном сценарии я хочу использовать его лишь для анализа и поиска логов для получения полезной информации.

*Kibana* написан на JavaScript и предлагает интерфейс для нашей станции по анализу логов. У него открытый код, как и у других компонентов, и его можно использовать для поиска, просмотра и взаимодействия с данными, хранящимися в *Elasticsearch*. Он также способен на большую мощь, причем работает прямо из браузера. А именно здесь мы проводим большую часть нашего времени, как только получим необходимые нам данные.

## Займемся логами

Прежде чем выполнять начальную настройку, я хотел быстро посмотреть работу, которую нужно провести на клиентах (в данном случае — множестве наших систем Linux). Признаюсь, что раньше я противился установке Java, особенно лишь для того, чтобы запустить агента пересылки логов (а *Logstash* написан на JRuby), и особенно в цехах разработчиков, которые к Java и багром бы не прикоснулись. К счастью, агент *Logstash* эволюционировал в проект под названием Beats, это легковесные процессы, написанные на Go. Доступен ряд разных Beats. Лично я рассматриваю *Filebeat*, заменивший старое приложение *logstash-forwarder*. Фактически, эти агенты при необходимости могут выгружать данные прямо в *Elasticsearch* (но тогда мы лишимся крутых преобразований, на которые способен *Logstash*).

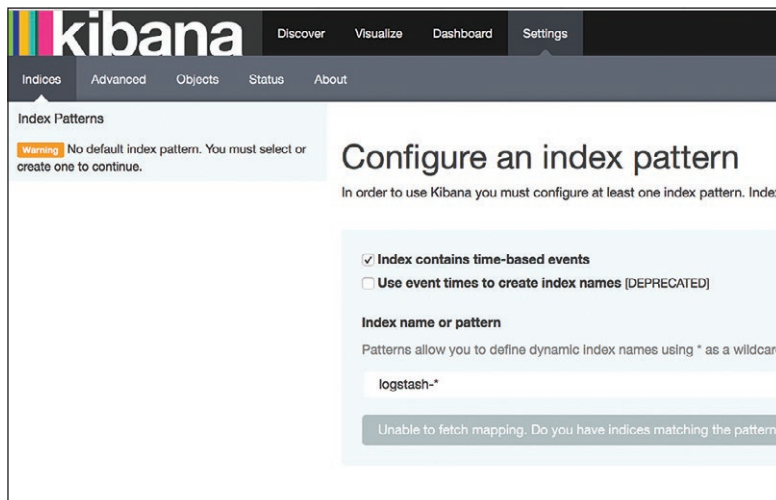
Теперь к главному: готовим все к работе. Elastic содержит свои собственные репозитории пакетов, с выбором для обычных дистрибутивов. Я придерживаюсь моей обычной Ubuntu 14.04, которая, я уверен, уже надоела многим из вас! Для начала я собираюсь установить стек ELK на виртуальную машину и затем заставить несколько клиентов отправить на нее несколько логов; а в следующем номере мы уже рассмотрим всякие «прикольные» штуки.

Взгляните на <http://bit.ly/ElasticRepoSetup>, где подробно сообщается о требуемых публичных ключах и т.п. Вкратце шаги могут быть описаны так:

```
$ wget -qO - https://packages.elastic.co/GPG-KEY-elasticsearch |
sudo apt-key add -
$ echo "deb http://packages.elastic.co/elasticsearch/2.x/debian stable main" |
sudo tee -a /etc/apt/sources.list.d/elasticsearch-2.x.list
$ sudo apt-get update
```

Заметьте, здесь специально избегают использовать `add-apt-repository`, поскольку нет доступного репозитория `deb-src`. Здесь я использую версию 2.x, следуя рекомендациям Elastic. Я не буду рассматривать установку Java (необходимое условие для установки самого *Elasticsearch*), которое идет со стандартной дилеммой «официальные пакеты Oracle» против OpenJDK (все должно быть нормально, по утверждению Elastic, и здесь я это и использовал).

Установка *Elasticsearch* сводится к запуску `$ sudo apt-get install elasticsearch`. Скачается около 28 МБ. Ubuntu установит это в качестве сервиса, как и следовало ожидать при установке через *apt*, что можно заменить на `$ sudo service start elasticsearch`.



➤ Нам понадобится наскоро настроить *Kibana*, чтобы увидеть все наши данные.

## Альтернативы ELK

Возможно, я уже об этом говорил, но что я действительно люблю в открытом коде — это нередкая доступность множества альтернатив, когда речь заходит о выборе программ. Это позволяет идеям со временем развиваться и совершенствоваться тому, что уже существует (ну, я так надеюсь). Например, части пакета ELK можно заменять. Альтернатива *Logstash* — *Fluentd*, написанная на Ruby (и очень удобно, что у нее есть драйверы для Docker), имеющая большую установленную пользовательскую базу (я работаю с рядом клиентов, которые называют эту структуру стеком 'FEK').

Graylog — основанная на открытом коде компания, которая использует *Elasticsearch* (и *MongoDB*) как часть своей структуры, а также собственные альтернативы для части пакета 'LK'. *Grafana* может служить альтернативой *Kibana* (по всей вероятности; лично я ее еще не пробовал), которую, как я замечал, критикуют за излишнюю тяжеловесность (*Kibana* недавно выпустила новую версию, что частенько вызывает беспорядки в пользовательской базе).

Настоящей же 200-фунтовой гориллой на этом поле выступает *Splunk*, который не относится

к открытому коду — но у него есть доступные свободные уровни, если вы хотите запускать небольшие структуры (по моему опыту, продукт отличный — но чрезвычайно дорогой). Другие коммерческие альтернативы SaaS — *SumoLogic* и *Loggly*. Однако помните, что в некоторых средах невозможно отправлять логи третьим лицам (или это вовлекает множество мер безопасности).

Если вы оцениваете эти типы инфраструктуры, дважды проверьте, чтобы не попасть впросак с какими-нибудь промышленными стандартами (например, PCI).

Перескочим в `/var/log/elasticsearch` и взглянем на `elasticsearch.log` — и вы должны увидеть, что все установлено и готово к работе (вывод здесь был сильно урезан):

```
[INFO ][node ] [Screaming Mimi] starting ...
[INFO ][transport ] [Screaming Mimi] publish_address
{127.0.0.1:9300}, bound_addresses [{127.0.0.1:9300}, {::1}:9300]
[INFO ][discovery ] [Screaming Mimi] elasticsearch/
A2FH81ZjTGaSB-HFYioXkQ
[INFO ][cluster.service ] [Screaming Mimi] new_master (Screaming
Mimi){A2FH81ZjTGaSB-HFYioXkQ}{127.0.0.1} {127.0.0.1:9300},
reason: zen-disco-join(elected_as_master, [0] joins received)
[INFO ][http ] [Screaming Mimi] publish_address {127.0.0.1:9200},
bound_addresses {127.0.0.1:9200}, {::1}:9200
[INFO ][node ] [Screaming Mimi] started
[INFO ][gateway ] [Screaming Mimi] recovered [0] indices into
cluster_state
```

Это показывает мой новый кластер *Elasticsearch* (из одного узла, который, естественно, сам провозглашает себя мастером) — готовый и работающий. Обратите внимание, что ему присвоено случайное имя.

На данном этапе я также могу установить *Logstash* (который весит в районе 78 МБ). И опять же, мне нужно добавить его репозитории в `/etc/apt/sources.list`:

```
$ echo "deb http://packages.elastic.co/logstash/2.1/debianstable
main" | sudo tee -a /etc/apt/sources.list
$ sudo apt-get update
$ sudo apt-get install logstash
```

## Сора Kibana

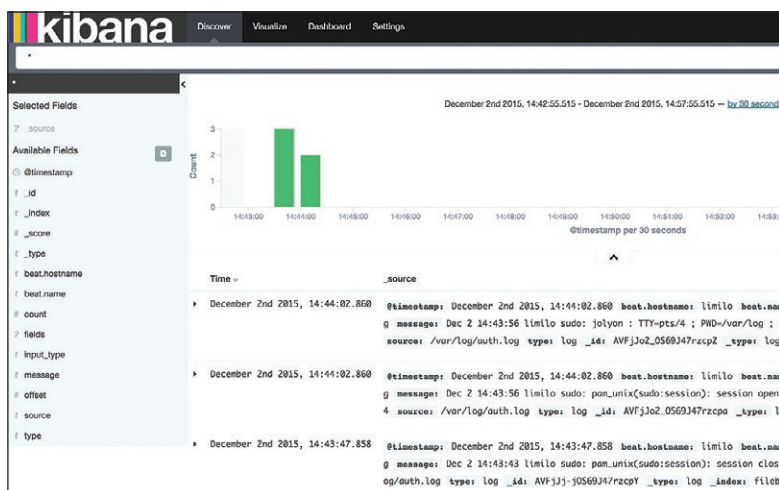
[Игра слов: Сорасабана — пляж в Рио-де-Жанейро, сор — амер. полицейский, — прим. пер.] К сожалению, для *Kibana* ситуация, похоже, включает прямое скачивание файла. Последняя версия (4.3), судя по всему, отсутствует в репозиториях (4.1 там была, но это намекает, что репозиторий не поддерживается — по крайней мере, для *Kibana*; как говорит Elastic, версия, совместимая с *Elasticsearch 2.x* — 4.3). Я бы предложил настроить специального пользователя и группу *kibana* для владения ПО и установить его в `/opt/kibana`, но, возможно, у вас есть свои собственные предпочтения. Я установил 64-битную версию *Kibana* с помощью

```
$ sudo adduser kibana
$ sudo mkdir /opt/kibana
$ sudo chown kibana:kibana /opt/kibana
$ sudo su - kibana
$ cd /opt/kibana
$ wget https://download.elastic.co/kibana/kibana/kibana-4.3.0-
linux-x64.tar.gz
$ tar -zxvf kibana-4.3.0-linux-x64.tar.gz
```

Теперь, как пользователь *kibana*, я могу перейти во вновь созданную структуру директории `kibana-4.3.0-linux-x64/bin` и запустить ее. *Kibana* должна запуститься без проблем и выдать ответы, как в примере (отредактированном) ниже:

```
$ ./kibana
log [13:20:47.173] [info][status][plugin:kibana] Status changed from
uninitialized to green - Ready
log [13:20:47.210] [info][status][plugin:elasticsearch] Status
changed from uninitialized to yellow - Waiting for Elasticsearch
log [13:20:47.323] [info][listening] Server running at
http://0.0.0.0:5601
log [13:20:52.394] [info][status][plugin:elasticsearch] Status
changed from yellow to yellow - No existing Kibana index found
log [13:20:55.949] [info][status][plugin:elasticsearch] Status
changed from yellow to green - Kibana index ready
```

*Kibana* соединяется с экземпляром *Elasticsearch*, работающим на `localhost` по умолчанию, что и подходит для моих тестов здесь. Я могу соединиться с этой (пустой) установкой *Kibana*, направив



браузер на моей VM на порт 5601. Чтобы остановить работу *Kibana*, я просто нажимаю `Ctrl+C`.

Простым тестом для нашего нового кластера ELK является прием логов из его `localhost`. Очень скоро мне понадобилось установить *Filebeat* (используя моего собственного пользователя, а не пользователя *Kibana*):

```
$ curl -L -O https://download.elastic.co/beats/filebeat/
filebeat_1.0.0_amd64.deb
$ sudo dpkg -i filebeat_1.0.0_amd64.deb
```

Теперь я могу взглянуть на `/etc/filebeat/filebeat.yml`. Есть пара разделов, на которые стоит обратить внимание на данном этапе: "paths" определяет, какие лог-файлы должны быть обработаны *Filebeat*. Чтобы добавить *Logstash*, надо раскомментировать строку `#hosts: ["localhost:5044"]` в разделе `logstash`. В этом пункте мне также нужно добавить строку `enabled: false` в разделе `elasticsearch` файла.

Чтобы обрабатывать информацию из *Filebeat*, надо установить плагин для *Logstash*. Команда

```
$ sudo /opt/logstash/bin/plugin install logstash-input-beats
```

живо разберется с этим. Мне также надо настроить *Logstash*, создав файл `etc/logstash/conf.d/config.json` и включив в него следующие строки:

```
input {
  beats {
    port => 5044
  }
}
output {
  elasticsearch {
    hosts => "localhost:9200"
    sniffing => true
    manage_template => false
    index => "%{[@metadata][beat]}-%{+YYYY.MM.dd}"
    document_type => "%{[@metadata][type]}"
  }
}
```

Теперь я могу все запустить:

```
$ sudo service logstash start
$ sudo service filebeat start
```

Переключившись на моего пользователя *kibana*, я могу запустить резервирование *Kibana* для вышеприведенных шагов и переподключиться с помощью моего браузера. В поле `Index name or pattern` я могу заменить `logstash-*` по умолчанию на просто `*`. Это приведет к появлению зеленой кнопки `Create`. Нажав на нее и затем на опцию `Discover` сверху, мы увидим, что моя локальная система запущена и лог сохранен в *Elasticsearch*. Через месяц мы начнем рассматривать, как опрашивать эти данные. **LXF**

» Угадайте, какой идиот (это я...) заблудился в скриншоте работающей системы?

# Samsung вступает в «Тайзен.ру»

Российская ассоциация «Тайзен.ру» призвана содействовать развитию технологии Tizen на территории Российской Федерации.

**S**amsung Electronics, в рамках Национального форума информационной безопасности «Инфофорум», проходящего в здании Правительства Москвы, 4 февраля 2016 г. объявила о вступлении в Российскую Ассоциацию «Тайзен.ру».

Российская Ассоциация «Тайзен.ру» создана с целью продвижения передовых ИТ-решений на основе платформы Tizen, рассчитанных на корпоративный рынок мобильной связи. Ассоциация, в частности, работает над профилем безопасности и выполнением «локальных» требований, удовлетворяющих стандартам российских отраслевых регуляторов. Одна из важнейших целей Ассоциации — глобальное развитие Tizen для всех сегментов и сфер применений, от ИТ-инфраструктуры до мобильных устройств и развлечений.

## Основная цель сотрудничества — обмен ключевой информацией.

Основой Ассоциации стали российские компании, занимающиеся информационной безопасностью (НИИ СОКБ, Инфотекс, Элвис Плюс и др.), которые объединили усилия с целью сделать Tizen наиболее защищенной ОС для доверенного использования.

Помимо ведущих отраслевых игроков, в Ассоциацию вошли академические институты (Институт системного программирования РАН, Институт проблем информационной безопасности МГУ и др.) и компании прикладной сферы, такие как «Новые облачные технологии», медиа-компания SPB TV и многие другие.

Российская версия Tizen 2.4 установлена на новый смартфон Samsung Z3, который уже прошел сертификацию ФСТЭК. В ноябре 2015 г. компания «Инфотекс» представила VPN-клиент для организации защищенного доступа к корпоративным сетям — VipNET Client для Tizen. В свою очередь, НИИ СОКБ снабдила Tizen системой управления мобильными данными SafePhone — с ней телефон защищен как от физического взлома, так и от несанкционированного проникновения.

Согласно подписанному соглашению, Samsung является основным технологическим партнером РТА. Основная цель сотрудничества — обмен ключевой технической информацией, в том числе информацией об изменении стандартов и программного кода, условий продвижения ОС Tizen для нужд мобильной связи, телевидения и Интернета вещей.

Samsung Electronics Co., Ltd превращает в жизнь передовые идеи и внедряет инновационные технологии, вдохновляя людей и формируя будущее потребительского рынка в мире телевизоров, смартфонов, носимых устройств, планшетов, камер, бытовых приборов, принтеров, медицинских устройств, сетевых систем, полупроводников и светодиодных решений. Компания является одним из лидеров в области развития инновационной индустрии «Интернета вещей [Internet of Things]», в частности, благодаря реализации концепции «Умного дома [Smart Home]» и «Цифрового здравоохранения [Digital Health]». В 84 странах у Samsung работают свыше 319 тысяч человек, а объем годовых продаж составляет \$ 196 млрд. Для получения более подробной информации посетите сайт [www.samsung.com](http://www.samsung.com) и официальный блог [global.samsungtomorrow.com](http://global.samsungtomorrow.com). **ЛХФ**



**ВЛАДИСЛАВ ШЕРСТЮК,**

директор Института проблем информационной безопасности МГУ им. М. В. Ломоносова, советник секретаря Совета безопасности Российской Федерации, председатель Наблюдательного совета Ассоциации «Тайзен.ру»: «Консолидация усилий лидеров ИТ-отрасли на решении конкретных задач создания конкурентоспособных информационных и коммуникационных технологий будет содействовать укреплению информационной безопасности Российской Федерации».



**АНДРЕЙ ТИХОНОВ,**

президент Российской ассоциации «Тайзен.ру»: «Миссия Ассоциации Tizen.ru заключается в максимальном ускорении внедрения инновационного решения, которое представляет собой ОС Tizen, в реально продаваемые продукты и системы на мобильном рынке и рынке Internet of Things. Мы планируем сделать Tizen OS системой номер один на рынке с точки зрения сегментов, предъявляющих повышенные требования к защите информации».



**МАРАТ ГУРИЕВ,**

директор по работе с государственными организациями Samsung Electronics: «Несомненно, сфера корпоративного управления ждет мобильную технологию именно на основе Tizen и после пилотной проверки практического решения нас ожидает ударный труд по тиражированию на десятках направлений с десятками заказчиков и многими сотнями тысяч проданных мобильных устройств. Мы очень рады, что российский проект Tizen возглавляет Андрей Тихонов. За последние три года Андрею удалось создать очень сильное направление B2B в российском подразделении Samsung Electronics; кроме того, он с самого начала лично руководил запуском проекта на основе Tizen — Samsung Z3. Это действительно здорово, что Андрей продолжает работу в рамках данного амбициозного проекта, который призван сделать Tizen одной из самых популярных и надежных ОС в России».



Наши эксперты помогут вам с любым приложением Linux!



**ЕВГЕНИЙ БАЛДИН**  
Подтвердивший  
свою квалификацию  
физик.

## Облако на коленке

Фаррел: Да вы сбили  
тачку вертолет!  
Мак-Клейн: Кончились патроны.  
*Крепкий орешек 4.0*

**Ч**его-чего, а облаков вокруг сейчас хватает. И я имею в виду не видимые на небе невооруженным глазом продукты конденсации водяного пара. Я про облачные хранилища данных. Потихоньку инфраструктура сети доросла до возможности переместить свои данные за пределы основного рабочего компьютера. Как минимум актуальную копию; и неплохо бы устроить синхронизацию между разными рабочими компьютерными устройствами, включая телефон.

К сожалению для подавляющего большинства успешных на этом рынке компаний, есть одна проблема: для доступа к их услугам необходимо поставить на свои устройства закрытый уникальный для выбранного сервиса клиент. Кроме того, если компания перестает заниматься такой деятельностью, как это было сделано с Ubuntu One, то приходится менять приобретенные привычки.

Поэтому пора потихоньку создавать свои личные облака. Благо соответствующее программное обеспечение уже готово. В зависимости от целей и привычек можно выбирать из *OwnCloud* (классическое PHP-приложение), *Syncthing* (активно разрабатываемый децентрализованный свободный «аналог Dropbox, но на языке Go») и *git-annex assistant* (создан фанатом Git для фанатов Git и на деньги фанатов Git). Вполне себе задача для «домашнего сервера». [E.m.Baldin@inp.nsk.su](mailto:E.m.Baldin@inp.nsk.su)

## В этом месяце вы научитесь...



### Рассылать письма ..... 56

Новый год давно прошел, но **Ник Пирс** справедливо рассудил, что он не последний в нашей жизни и навыки рассылки еще пригодятся.



### Управлять разделами ..... 60

Потрясенный мощью LVM, **Нейл Ботвик** решил употребить эту мощь по новой схеме и принялся добавлять и удалять логические тома.



### Рудить дисками ..... 62

Если вы уже успели поднакопить не один терабайт личных архивов, поучитесь организовывать хранилища у **Нейла Ботвика**. Рудит RAID.



### Размечать диск ..... 64

Разметка диска бывает полезна для самых разнообразных целей, и **Ник Пирс** рекомендует изучить этот вопрос доскональнейшим образом.



### Клонировать ОС ..... 68

Создав всего одну рабочую станцию, **Маянк Шарма** берется мигом обставить такими целую лабораторию.



### Шифровать диски ..... 70

Стоит ли мелочиться, шифруя файлы поодиночке? **Маянку Шарме** по плечу зашифровать и весь диск целиком.



### Звонить в город ..... 72

**Максиму Черепанову** оказалось мало связи через мобильного оператора, и он покусился на городские номера.



### Ставить MySQL Fabric ..... 76

**Лада Шершова** устанавливает и запускает ПО для управления фермой серверов при высокой их доступности.

## АКАДЕМИЯ КОДИНГА



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

### Пишем скрипты на Lua ..... 80

Без лишних проволочек, **Михалис Цукалос** воплотил теорию в практику: написал скрипты для мониторинга системы.

### Изучаем новый Perl ..... 84

Пришествие Perl 6 неизбежно, смекнул **Михалис Цукалос** — и засел за разбирательство его особенностей.

# LibreOffice. Как делать рассылки

Ник Пирс предпочитает экономить время и осваивает массовую рассылку писем в LibreOffice, используя комбинацию Base, Calc и Writer.



Наш эксперт

Ник Пирс занимается компьютерами более 30 лет и баулется Linux лучшей частью десятилетия.



У вас полная версия LibreOffice? Некоторые дистрибутивы Ubuntu не включают его по умолчанию, в этом случае apt-get install libreoffice выполнит установку и добавит недостающие биты.

Все, кому приходилось писать одно и то же послание множеству разных людей, добросовестно вставляя имена и адреса каждого из них, оценит то, что способна предложить рассылка писем. (А также те, кому лень писать имена и адреса на конвертах на Новый год). На их счастье, в LibreOffice Writer есть инструмент Рассылка писем..., способный упростить процесс. Рассылка писем получает список контактов и автоматически вставляет необходимую информацию из этого списка в определенные части документа — например, адрес в письме или личное приветствие. Список контактов хранится отдельно и может быть получен из любого из трех источников: базы данных, правильно форматированной таблицы или CSV-файла.

LibreOffice даже сумеет взять за источник списка контактов существующую адресную книгу Thunderbird или Icedove, хотя придется преобразовать ее в базу данных только для чтения выбором пункта меню Файл > Мастер > Источники данных адресов.... Впрочем, на нашем уроке мы будем основывать список контактов на правильно форматированной таблице, которую затем можно напрямую импортировать в Рассылку писем (см. Рассылка писем через Writer и Calc внизу) или дополнить ею базы данных Base, что придает большую гибкость ее использованию.

## Настроим список контактов

Список контактов должен быть организован в общем для всех баз данных виде — полей и записей. Конкретные контактные данные формируют основу каждой записи, а части этих записей разнесены в поля, например, имя, фамилия, адрес и т.д. Эти данные представлены в виде таблицы, где каждому полю отведен свой столбец, а каждой записи — своя строка.

1	A	B	C	D	E	F	G
1	first_name	last_name	address	city	country	postal	phone1
2	France	Andrieu	8 Moor Place	East Southbourne and Tuckton W	Bournemouth	BH4 3BE	01327-368222
3	Alyssa	Ansboro	85 Hero St	Slaptope	County Durham	DL13 2TZ	01602-881825
4	Paco	Aschoff	135 Cople St	Howard Town Ward	Derbyshire	SK13 3BD	01296-951144
5	Margaret	Bain	3 August Rd	Maudsley and Sherwood Ward	Surrey	GU21 5QL	01870-813697
6	Kiera	Basell	5152 Sophia St	Woodstock Ward	South Gloucestershire	BS15 8DS	01652-519837
7	Maria	Brace	459 Lake St	Malmesbury	Wiltshire	SN16 0XZ	01777-469402
8	Nehra	Burch	24 Bute St	Broomham, Spittal and Winchburg	West Lothian	EH45 5TL	01874-859050
9	Sibehana	Cartea	33 Yipond St	Woodhall Farm Ward	Hertfordshire	HP2 7JP	01424-348876
10	Regina	Chisam	652 Grove Park	Butwell Ward	Nottinghamshire	NG6 8NG	01332-534521
11	Tina	Chisam	5922 William Mount St	Aylshury	Ruckinghamshire	HP15 8PP	01567-296442
12	Clara	Cobbley	7524 Kempton Rd	Cockerton West Ward	Derbyshire	DL3 0JF	01223-647922
13	Marybelle	Compslate	5410 Lawton St	Torquay Community	Rhodesia, Cymon Tall	CF42 6PL	01303-5647025
14	Meredith	Compton	29 Wye St	Wakenfield North Ward	West Yorkshire	WF5 1DR	01924-808218
15	Soi	Cousser	6440 Tilgast St	Constrough and Denaby Ward	South Yorkshire	DN12 4JF	01412-526394
16	Abraham	Critch	41 Berendict St	Aldersnagh Ward	Greater London	IG2 7QJ	01599-245408
17	Ladette	Crocz	484 Blary St	Stoney Row Ward	County Durham	DH4 7NS	01913-620249
18	Ella	Denise	260 Savon St	Brooklands East Ward	Greater Manchester	M33 4BP	01537-525550
19	Elbert	Drone	9 Cypress St	Washington East Ward	Tyne and Wear	NE38 9AQ	01763-784273
20	Jessie	Dulle	87 Eversal St	Seaton	Cambridgeshire	CB22 2BB	01208-954981
21	Luis	Ear	2 Birchfield Rd	Whittington	Shropshire	SV11 4PH	01462-546669
22	Rayen	Ekan	67 Pufford St	Pittiswell Ward	Essex	SS2 8NA	01368-416887
23	Charlotta	Em	2 Hagden St	Louthby Green Ward	Derbyshire	S40 4LY	01273-838006
24	Eloisa	Faustoe	7033 Mitchell St	Blintsee	Norfolk	NR20 5HQ	01337-655353
25	Stephany	Frensch	638 Portland Place	Gresham Ward	Middlesbrough	TS1 4RS	01743-977277
26	Blaine	Fuchs	222 Susanic St	Aspley Bridge Ward	Greater Manchester	B11 5PP	01451-821447
27	Ruben	Castellum	4 Forest St	Weston-Super-Mare	North Somerset	BS23 3HG	01974-755279
28	Tony	Gasthae	274 Alcar Ave	Catnick	North Yorkshire	DL10 7NJ	01585-580015
29	Lenny	Geddes	6 Rowley St	Walthamstow Ward	Tyne and Wear	NE8 4TH	01312-523253
30	Mimi	Grampick	7457 Cowe St	Barnes Ward	Southampton	SO14 3TY	01865-592516
31	Margret	Guest	2732 Rockock St	St. Ender	Cornwall	TR9 8N4	01863-310794

» Проще всего записать список контактов в таблицу Calc, а затем импортировать ее в Base.

Первая строка любой таблицы должна содержать заголовки столбцов, помогающие определить каждое поле. Если вы начинаете с самого начала, откройте Calc и создайте пустую базу данных. Введите имена всех полей в столбцы строки A — вот пример того, что включать: имя, фамилия, адрес, город, область, страна, почтовый индекс, домТелефон, рабТелефон, мобТелефон, элктроПочта, сайт, категория.

Очевидно, вам может потребоваться добавить больше информации — например, некоторые адреса требуют второй строки; поэтому включите адрес1 и адрес2. Или, если у вас нет необходимости в стране или сайте, смело исключайте их. Один из особо

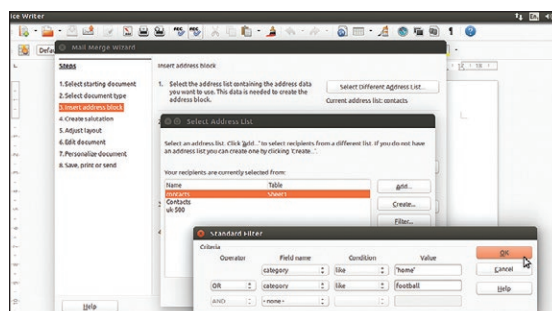
## Рассылка писем через Writer и Calc

Если вам некогда разбираться в базах данных, то рассылку писем можно осуществить, используя в качестве источника таблицу контактов; можно даже установить фильтры, чтобы включить только часть записей.

Открыв заполненную таблицу Calc, создайте пустой документ в Writer и выберите Сервис > Рассылка писем.... Следующие действия практически идентичны пошаговому руководству на стр. 59, но есть одно принципиальное различие: дойдя до выбора списка адресов, нажмите кнопку Добавить..., чтобы найти свой файл электронной таблицы. Когда он появится в списке,

выделите его и нажмите кнопку Фильтр... Можно добавить до трех фильтров, чтобы точно указать записи в списке.

Добавление фильтра включает выбор трех элементов: имя поля, условие и значение. Чтобы отправить письмо, например, всем вашим рабочим контактам, после нажатия на кнопку Фильтр... выберите поле категории в Имя поля. Задайте условие «совпадение» и введите категорию («Работа») в поле Значение. Желая добавить другие условия, выберите имя поля, затем выберите, надо ли условиям срабатывать по отдельности (выберите ИЛИ) или одновременно (И).



» Добавьте несколько фильтров с оператором ИЛИ, чтобы соответствовать более одной категории в вашем списке контактов.

важных заголовков столбцов — категории. Он позволяет привязать каждый контакт к определенной группе, такой как «дом» или «работа». Чтобы узнать больше об этом поле и добавить его в имеющийся список контактов, прочитайте врезку «Подготовка базы данных Calc для переноса» (см. стр. 58).

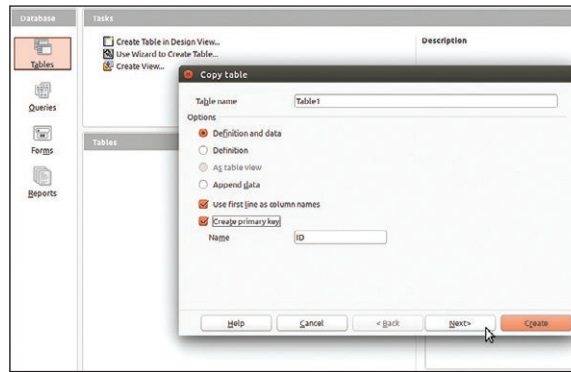
Присвоив заголовки столбцам, можно вносить контактные данные. Помните, каждый контакт занимает строку в базе данных, так что первая запись занимает строку 2, вторая — строку 3, и т.д. Не забывайте сохранять таблицу по мере ввода; как только закончите, поздравляем — вы создали вашу базу данных контактов. Не обязательно вводить записи в алфавитном порядке: если вы хотите, чтобы все было красиво и аккуратно, просто выберите столбец с фамилиями и нажмите кнопку Сортировать по возрастанию, расположенную на панели инструментов.

## Перенос в Base

Теперь вы готовы непосредственно перейти к применению инструментов рассылки писем в *Writer* (во врезке на стр. 58 показано, как это сделать, используя таблицу *Calc*). Но, хотя *Calc* достаточен для выполнения этой задачи, есть основания пойти дальше и использовать данные таблицы, чтобы создать полноценную базу данных с помощью компонента базы данных LibreOffice — *Base*.

Во-первых, *Base* поддерживает более сложные базы данных, включая реляционные, которые позволяют создать несколько таблиц данных и связать их вместе. Если, например, у вас большая коллекция DVD, которые вы одалживаете друзьям, вы могли бы это отслеживать, используя три таблицы: ваш список контактов; вторую таблицу со списком всех ваших DVD; и, наконец, третью таблицу, позволяющую связать DVD с людьми, которые эти DVD позаимствовали.

Во-вторых, *Base* поддерживает запросы, позволяющие создавать подмножества списка контактов на основе ряда фильтров. На нашем уроке мы используем запросы, которые позволяют объединенные с почтой документы связать непосредственно с контактами, находящимися в указанной группе или категории контактов; но можно пойти дальше. Можно настроить запрос для



» Быстро перенесите данные из *Calc* в *Base*, используя копирование и вставку, чтобы открыть мастер копирования таблиц.

фильтрации списка по городу проживания или по фамилии (*Base* позволяет отбирать контакты по нескольким критериям: например, можно включить три или более разных фамилий, чтобы охватить всех своих родственников).

Через *Base* также удобнее вводить данные в список контактов — в окне формы записи легче ориентироваться, чем в таблице. И, наконец, можно использовать список контактов для генерации отчетов о данных с помощью инструментов отчетов *Base* — это, например, изящный способ создать отформатированную распечатку ваших контактов.

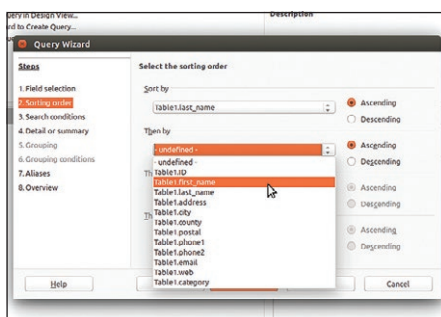
Хорошей новостью будет то, что переход от *Calc* к *Base* не означает создания своего списка контактов с нуля. Вот как импортировать данные из таблицы: во-первых, откройте *Base*, и появится Мастер базы данных. Оставьте пункт Создать новую базу данных с выбранным по умолчанию параметром HSQLDB Embedded и нажмите кнопку Далее. Оставьте настройки по умолчанию и нажмите Готово. Найдите требуемую папку — как правило, **Документы**, затем дайте базе данных имя и нажмите кнопку Сохранить.

Теперь переключитесь в *Calc* и откройте таблицу. Выберите данные для копирования — включая крайне важный заголовок столбца — и выберите Правка > Копировать. Вернитесь в *Base*, убедитесь, что включен вид таблицы, и выберите Правка > Вставить, »

**Скорая помощь**

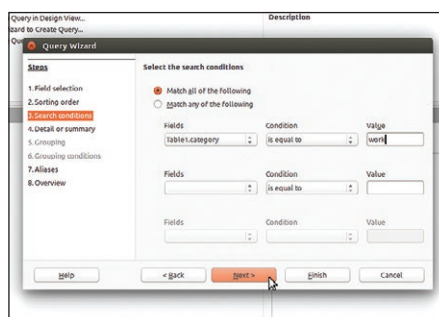
Для удаления базы данных выберите Сервис > Параметры, разверните раздел *Base* и выберите Базы данных. Выделите ненужную базу и нажмите кнопку Удалить, чтобы отключить ее.

# Создание запросов базы данных для отбора записей



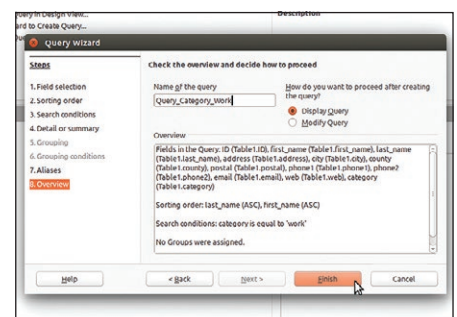
### 1 Создайте простой запрос

Вам потребуется создать отдельные запросы для каждой группы контактов в *Base*. Переключите на Запросы в левой панели, затем дважды щелкните Использовать Мастер для создания запроса. Нажмите кнопку >>, чтобы добавить все доступные поля в запрос. Нажмите кнопку Далее, затем в выпадающем меню выберите сортировку результатов в нужном порядке (как правило, по фамилиям, а затем по именам). Снова нажмите Далее.



### 2 Уточните запрос

Оставьте выбранным Совпадение по всем, затем щелкните выпадающее меню Поля, чтобы выбрать группу или категорию из списка. Оставьте условие «равно» и введите требуемые значения группы (например, «дом» или «работа») в поле Значение. Нажмите кнопку Далее, оставьте пункт Подробный запрос выбранным и нажмите кнопку Далее еще раз. Оставьте псевдонимы как есть, затем опять нажмите кнопку Далее.



### 3 Именованное, проверка и повторение

Наконец, дайте запросу значащее имя, например 'ЗапросПоРабота'. Нажмите Готово, чтобы увидеть отобранный список контактов. Закройте окно, затем повторите шаги, чтобы настроить отдельные запросы для каждой группы в списке контактов. Процедура идентична, за исключением того, что в первом шаге мастера нужно выбрать таблицу, а не запрос в выпадающем меню.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)

## Скорая помощь



При просмотре этикеток или документов почтовой рассылки нажмите F4 или кнопку Источники данных, чтобы раскрыть удобную панель, позволяющую просмотреть текущий источник данных — нажмите кнопку Изменить в панели инструментов Источники данных, и вы можете даже внести изменения в базу данных, не покидая *Writer*.

чтобы открыть окно копирования таблицы. Оставьте выбранным Definition and data, затем убедитесь, что включены оба флажка — в том числе Создать первичный ключ — и нажмите кнопку Далее. Проверьте, какие поля вы хотите импортировать; нажмите кнопку >>, чтобы импортировать их все, или выберите нужные по очереди и нажмите кнопку >, чтобы добавить их в список. Нажмите кнопку Создать. Вы должны увидеть Таблица 1 в разделе Таблицы: дважды щелкните по ней, чтобы проверить, что все записи были импортированы. Закройте окно, когда закончите. Теперь база данных создана; пришло время добавить к ней такие запросы, которые позволят отфильтровать контакты по друзьям, коллегам или другим установленным критериям. Для этого следуйте пошаговой инструкции (см. стр. 57).

## Подготовим письма для рассылки

Подготовив список контактов и запросов, воспользуйтесь их преимуществами в *Writer*. Хорошая новость: предоставляется Мастер рассылки писем для писем и электронных сообщений, что сильно упрощает процесс. Мастер создает один документ, содержащий несколько страниц или разделов, по одному на каждого человека в вашем списке контактов.

Весь мастер можно пройти в шесть шагов: выберите источник данных — один из запросов к базе данных для отбора списка, затем настройте два почтовых элемента, а именно — блок адреса и личное приветствие. Затем создайте документ, добавив личные штрихи каждому письму, и, наконец, сохраните, распечатайте или отправьте по электронной почте готовый результат. Прежде чем перейти к седьмому шагу мастера, где можно при желании отредактировать письмо, убедитесь, что вас удовлетворяет универсальный вариант письма. Если вернуться назад и изменить основной документ, то персонализация будет потеряна.

По окончании создания письма для рассылки вам будут доступны четыре варианта: первый — сохранить начальный, т.е. основной документ. Второй — Save merged document, это возможность сохранения единого многостраничного документа или индивидуальных документов по отдельности — выберите имя файла, а *Writer* добавит цифры в конце каждого из них. Можно также сохранить определенный ряд документов, введя начальный и конечный номера.

Третий вариант позволяет распечатать документы, подготовленные к рассылке — снова можно напечатать все или указать диапазон. Наконец, можно использовать поле электронной почты в списке контактов для отправки документов по почте (плюс вручную введите адреса в поля копии и скрытой копии, нажав кнопку Копировать в...). Укажите тему письма, выберите способ отправки сообщения в виде обычного текста, HTML или во вложении

(поддерживается *OpenOffice*, PDF и *Word*) и снова выберите между отправкой всем или только конкретным людям в списке. Нажмите кнопку Отправить документы, а затем — при необходимости — введите данные своей учетной записи электронной почты в *LibreOffice*, которые она будет использовать для отправки электронной почты.

## Печать этикеток и конвертов

И последнее, где можно использовать базу данных контактов в *Writer* — распечатать серию конвертов (Вставка > Конверт) или этикеток (Файл > Создать > Этикетки). Оба работают по одинаковому принципу: в случае этикетки, оставьте поле Адрес не отмеченным — оно просто указывает обратный адрес, определенный в разделе Сервис > Параметры > LibreOffice > Сведения о пользователе. Далее, выберите список контактов из выпадающего меню базы данных, затем укажите таблицу или запрос, содержащий адреса, которые вы хотите использовать, через выпадающее меню Таблица.

Далее необходимо выбирать по порядку все поля базы данных и жать кнопку со стрелкой влево, чтобы добавить их в текстовое поле этикетки. Тут могут возникнуть некоторые сложности с размещением — например, не забудьте добавить пробел между полями имени и фамилии, плюс перевод строки между частями адреса.

После этого используйте параметры форматирования, чтобы задать размеры этикетки. В выпадающем меню Марка выберите подходящую этикетку — там есть Avery, Herma, Leitz, Sigel и Tower. Если подходит, откройте выпадающее меню Тип и уточните лист; если нет, переключитесь на вкладку Формат, чтобы ввести эту информацию вручную, используя этикетки в качестве ориентира. Наконец, вкладка Параметры позволяет распределить этикетки по странице или распечатать их по отдельности, а также настроить принтер, используя кнопку Настройка...

Когда будете готовы, нажмите кнопку Создать документ и подождите, пока *Writer* создаст этикетки. Фактически вы не будете видеть имена и адреса в списке контактов — просто ряд имен полей. Это нормально — когда вы выберите Файл > Печать, то увидите предупреждение, что ваш документ содержит поля адреса и спросит, хотите ли вы распечатать бланк письма; нажмите Да.

Появится диалоговое окно печати с выбранным вами контактами — можно распечатать их все, выбрать пронумерованный диапазон или, удерживая левую клавишу Ctrl, щелкнуть по отдельным строкам рядом с контактами, чтобы указать и выбрать, какие именно адреса печатать. Затем можете выбрать вывод файла на принтер или сохранить его как файл, при открытии которого поля информации заменяются реальными именами и адресами выбранных вами контактов.

## Подготовка базы данных Calc для переноса

Прежде чем импортировать таблицу списка контактов в *Base*, откройте ее в *Calc* и сначала убедитесь, что у вас есть заголовки столбцов (такие как имя, фамилия, адрес и так далее) в строке 1 — *Base* создаст соответствующие поля из этих заголовков. Если вы не сделали этого, выделите первую строку, нажав '1' слева от нее, затем по правой кнопке мыши выберите Вставить строки выше. Введите имена полей в новую пустую строку.

Вводя или редактируя имена полей, следите, чтобы между словами не было пробелов —

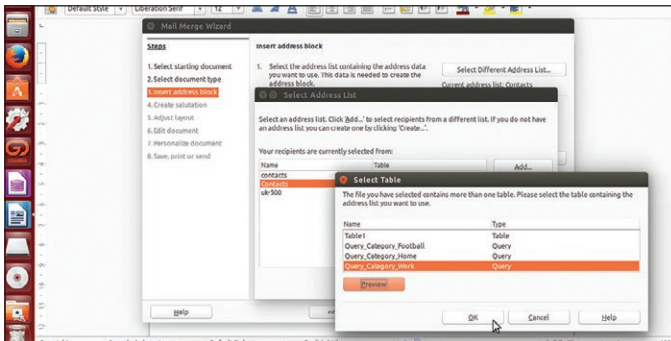
в качестве замены сгодится знак подчеркивания (\_), например, 'дом\_адрес'.

Наконец, нужно добавить один или два дополнительных поля в список: назовите первое поле 'категория' или 'группа', которое позволяет фильтровать контакты соответственно в группы, такие как 'дом', 'семья' или 'работа'. Второе поле применяется, только если чей-то пол является важным при написании приветствия в письме (например, «Дорогая миссис Смит»). Назовите его 'пол' и назначьте M для мужчин и Ж для женщин.

При большом количестве контактов заполнение их категории и пола вручную будет утомительным процессом, поэтому ускорим этот процесс путем ввода первой категории — скажем, 'дом' — в первую ячейку, где она появляется. Теперь щелкните правой кнопкой мыши ячейку и выберите Копировать. Далее, удерживая левую клавишу Ctrl, щелкните на каждой ячейке, где вы хотите, чтобы появился 'дом'. Выделив их все, щелкните правой кнопкой мыши и выберите Вставить, и вы должны увидеть, что эти клетки мигают.

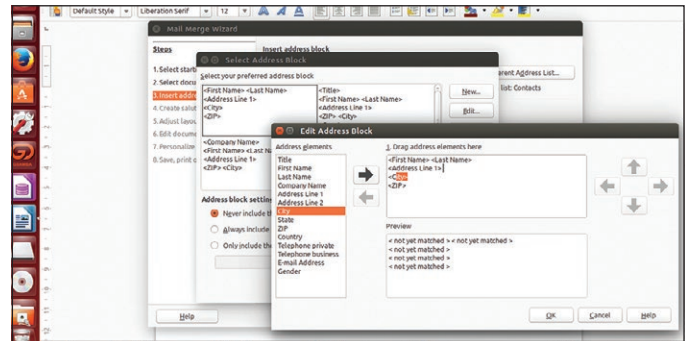
» Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

# Создайте письмо для почтовой рассылки



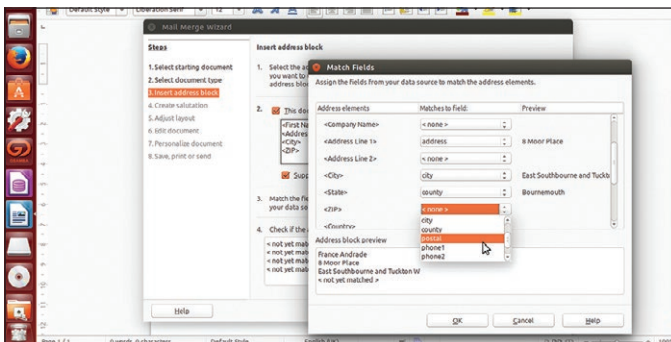
## 1 Выберите контакты

Откройте пустой документ в *Writer* и выберите Сервис > Рассылка писем. Оставьте Использовать текущий документ выбранным и нажмите Далее. Убедитесь, что выделено Письмо, и нажмите Далее еще раз. Сперва выберите вашу базу данных и нажмите кнопку Изменить таблицу, чтобы выбрать отфильтрованный список из имеющихся запросов, который вы хотите использовать (нажмите кнопку Предпросмотр, чтобы убедиться в правильности списка). Дважды нажмите кнопку ОК.



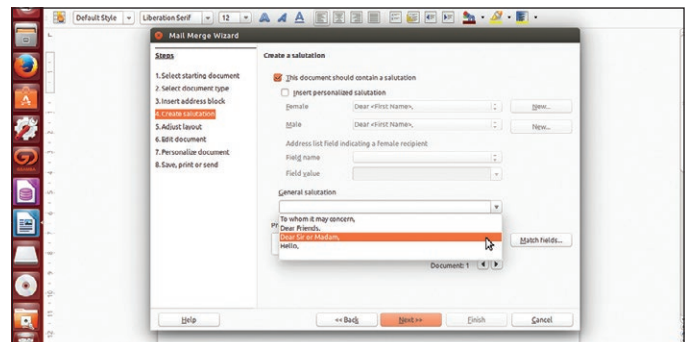
## 2 Отредактируйте блок адреса

Нажмите Подробнее..., чтобы выбрать адрес блока, который вы хотите использовать — выберите наиболее подходящий для вас и нажмите кнопку Изменить... Добавьте новые элементы, перетаскивая их из левой панели. Щелкните и перетащите имеющиеся элементы внутри списка, или выберите ненужный элемент и, нажав кнопку <, удалите его. Когда вы со всем этим покончите, нажимайте кнопку ОК два раза.



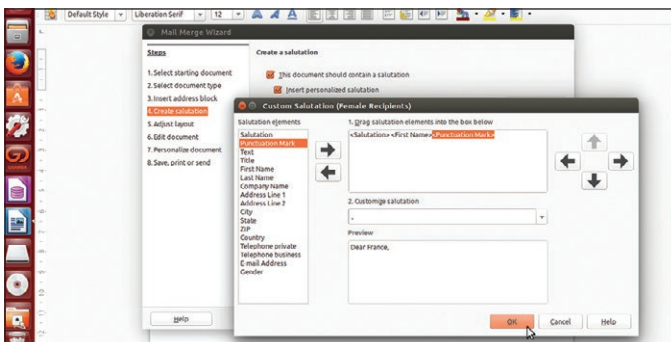
## 3 Соответствие полей

Далее, нажмите кнопку Соответствие полей.... Используйте выпадающее меню рядом с каждым элементом, чтобы сопоставить их с полями в базе данных — не надо сопоставлять каждый, любой элемент, который вы не планируете использовать, оставьте помеченным <Нет>. Используйте окно предпросмотра адресного блока для проверки соответствия полей; закончив, жмите кнопку ОК, затем Далее.



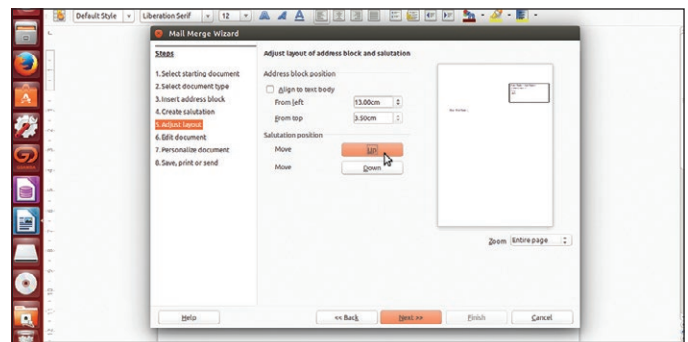
## 4 Создайте обращение

Чтобы дать общее обращение, уберите галочку у Вставить личное приветствие и выберите из выпадающего меню Общие приветствия. Или оставьте галочку, чтобы применить личное приветствие, используя список контактов. Если ваши приветствия зависят от пола, выберите поле пола в выпадающем меню Имя поля, затем выберите значение (как правило, «Ж» или «Жен») указывающее, что это женщина.



## 5 Создайте личное приветствие

Нажмите кнопку Создать... рядом с Женщина. Перетащите элемент Приветствие в верхнее поле и введите подходящее данному случаю приветствие (например, «Дорогая» или «Уважаемая») в поле Настройка приветствия. После этого перетащите Имя в список и затем, наконец, перетащите знак препинания. Выберите и вставьте запятую (,) в поле Настройка приветствия. Используйте окно предварительного просмотра с целью убедиться, что все выглядит нормально, и затем нажимайте кнопку ОК.



## 6 Составьте и отправьте

Нажмите кнопку Далее, чтобы выбрать место для адреса и приветствия на странице. Нажмите Далее, затем Редактировать документ..., чтобы написать само письмо. Когда закончите, нажмите Вернуться в Мастер рассылки писем. Нажмите Далее, и отдельные документы будут создаваться как отдельные страницы. Вы получите возможность персонализировать отдельные письма; затем нажмите кнопку Далее, чтобы сохранить, распечатать или отправить по электронной почте подготовленный к рассылке документ. LXF

# LVM. Содержимое дисков в порядке

Нейл Ботвик норовит воспользоваться мощностью LVM, чтобы получить максимум от жестких дисков и разделов.



Наш эксперт

Нейл Ботвик отлично разбирается в загрузке, ведь у него по компьютеру в каждой комнате; а вот в перезагрузке он не силен, поскольку перешел на Linux.

Раньше, когда жесткие диски были маленькими и дорогими, управление разделами не было большой проблемой. После установки разделов **root**, **swap** и **home** для чего-либо еще оставалось не много места, а несколько дисков были роскошью. Теперь, когда жесткие диски стали такими большими, быстрыми и дешевыми, нужен лучший способ их организации. Да, можно просто иметь **root** и **swap** и валить все в корневой раздел, но у отдельных файловых систем есть много преимуществ.

Файловые системы т.н. «следующего поколения» [см. *Сравнение LXF203*], такие как Btrfs и ZFS, будут поддерживать всё, включая RAID, управление томами, снимки и многое другое. А если вы хотите придерживаться испытанных ФС, таких как ext4 или более быстрой для больших файлов XFS, и все-таки хотите работать с несколькими дисками и разделами? Тогда вам нужен менеджер логических томов (LVM). Если у вас Fedora, вы, вероятно, уже используете LVM, и здесь мы покажем, как всё это работает и как извлечь из этого лучшее.

## Основы LVM

Есть три основных составляющих LVM:

» **Физические тома (physical volumes, PV)** Это физические блочные устройства, используемые в качестве строительных блоков. Как правило, это разделы диска, но могут быть и устройства dm-crypt для шифрования или файлы виртуального диска, если вы хотите поэкспериментировать с LVM, или даже целые диски или любые их комбинации.

» **Группа томов (volume groups, VG)** Группа томов состоит из одного или более физических томов и может рассматриваться как LVM-эквивалент диска — своего рода виртуальный диск. Физические тома могут быть добавлены (или удалены) в группу томов, чтобы изменить ее размер.

» **Логические тома (logical volumes, LV)** Это устройства, на которых создаются файловые системы. Их можно считать разделами на группе томов, но у них нет ни физических границ раздела жесткого диска, ни ограничений, связанных с ними.

Начнем с простого примера, с одним PV. В примерах будем предполагать, что **/dev/sdb1** — это раздел на внешнем устройстве. (Пока не трогайте свой системный диск, а посмотрите варианты во врезке «Пробная установка» на с. 61). Если этот раздел уже содержит файловую систему, он будет перезаписан. Все команды должны выполняться от имени **root** или предваряться **sudo**. Начните с `$ pvcreate /dev/sdb1`. Это просто инициализирует готовность раздела. Теперь создадим группу томов, содержащих его:

```
vgcreate MyFirstVG /dev/sdb1
```

Дадим VG значащее имя — возможно, содержащее имя хоста компьютера. Не используйте что-нибудь вроде VG00: если позже вы подключите этот диск к другому компьютеру, который имеет VG с тем же именем, то поймете, почему уникальные имена важны. Следующий шаг заключается в создании логического тома и установке на нем файловой системы:

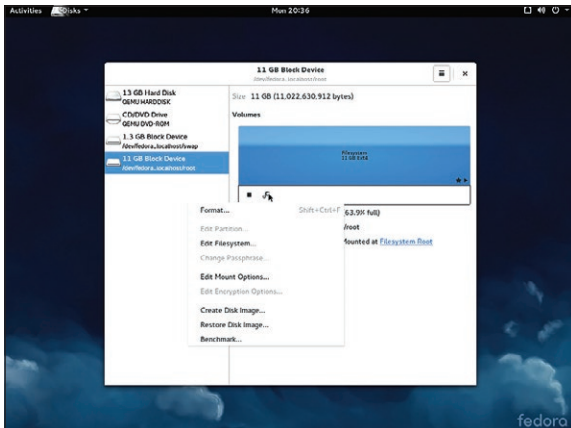
```
lvcreate --size 1G --name TestLV MyFirstVG  
mkfs.ext4 /dev/MyFirstVG/TestLV
```

Во второй команде можно видеть, что ваши группы томов отображаются как каталоги в каталоге **/dev**, содержащем входы каждого логического тома; вы также найдете их всех в **/dev/mapper/vgname-lvname**. После создания логического тома можно делать всё, что вы обычно делаете с разделом — например, создать и смонтировать файловую систему; но можно покуситься и на нечто поинтереснее.

Для начала, больше нет ограничений на размер раздела — можно создать сколько угодно LV, с легкостью организовывая жесткие диски. Беспокойтесь о скачивании всех этих ISO-образов, заполняющих ваш домашний каталог? Создайте для них отдельный LV: будет наполняться только он, и не вызовет проблем с загрузкой, переполнив более важные файловые системы (попробуйте войти в систему, если **/home** или **/var** заполнен на 100%). Также можно свободно добавлять, удалять и изменять размеры логических томов; поступив так с разделами, вы вызовете значительную перетасовку данных, необходимость дополнительных операций **GParted** и, обычно, перезагрузку. Из-за этого при организации диска важно сменить менталитет. Думать надо не обо всем доступном пространстве, а только о том, что вам нужно. Сделайте логический том достаточно большим для ожидаемых потребностей, плюс достаточный запас, чтобы избежать фрагментации. Вообще это хорошая идея — по возможности заполнять меньше 80%: деля файловую систему, скажем, на 50% больше, чем вы будете использовать, вы получаете пространство для маневра. А почему бы не сделать его еще больше? А потому, что вам этого не надо,

» **lvdisplay** и его собратья **vgdisplay** и **pvdisplay** обеспечивают массу информации о компонентах системы LVM.

```
Activities | MATE Terminal | Mon 20:40  
netz@localhost:~/home/netz  
File Edit View Search Terminal Help  
[root@localhost netz]# lvdisplay  
--- Logical volume ---  
LV Path                /dev/fedora_localhost/swap  
LV Name                swap  
VG Name                fedora_localhost  
LV UUID                0de1b2c-M9dB-Yony-ekcp-fSJh-h0Lz-Yi0x9B  
LV Write Access        read/write  
LV Creation host, time localhost, 2015-02-26 22:59:30 +0000  
LV Status               available  
# open                 2  
LV Size                1.26 GiB  
Current LE             307  
Segments               1  
Allocation              inherit  
Read ahead sectors     auto  
- currently set to    256  
Block device           253:0  
--- Logical volume ---  
LV Path                /dev/fedora_localhost/root  
LV Name                root  
VG Name                fedora_localhost  
LV UUID                Z2EY6E-Ux0e-LKW5-8V8B-dS3c-aqCU-W6aIPp  
LV Write Access        read/write  
LV Creation host, time localhost, 2015-02-26 22:59:31 +0000
```



► Fedora использует LVM по умолчанию и обеспечивает минимальные графические инструменты, но использование всей группы томов для `swap` и `root` значительно снижает предлагаемую LVM гибкость.

но может понадобиться пространство для чего-то другого, и придется менять модель использования. Кроме того, увеличение логического тома и раздела на нем очень легко, обратное же далеко не верно. Золотое правило изменения размеров файловых систем: увеличение гораздо, гораздо легче, чем уменьшение. Увеличение размера вышеуказанного пробного тома до 2 ГБ очень просто:

```
lvresize --size 2G MyFirstVG/TestLV
```

или

```
lvresize --size +1G MyFirstVG/TestLV
```

затем

```
resize2fs /dev/MyFirst/VG/TestLV
```

Это можно сделать с файловой системой не только смонтированной, но даже работающей. Уменьшение размера — это намного больше работы и времени, поскольку предполагает сначала отмонтирование файловой системы, затем уменьшение с `resize2fs`, которое часто требует проверки `fsck`, затем использование `lvresize` для уменьшения размера LV и, наконец, запуска `resize2fs`, чтобы снова создать файловую систему LV, поскольку придется дополнительно уменьшить файловую систему еще немного, чтобы убедиться, что при уменьшении LV вы ничего не потеряете. Если том содержит системные каталоги, возможно, вам придется воспользоваться LiveCD.

```
umount /dev/MyFirst/VG/TestLV
```

```
fsck -f /dev/MyFirst/VG/TestLV
```

```
resize2fs /dev/MyFirst/VG/TestLV 950M
```

```
lvresize --size 1G MyFirst/VG/TestLV
```

```
resize2fs /dev/MyFirst/VG/TestLV
```

## Несколько дисков

Что происходит при исчерпании места на диске? Менять размеры можно только пока группа томов не заполнится; а что потом? На настольном компьютере или сервере ответ довольно прост — добавить еще один жесткий диск и назвать его `/dev/sdc`:

```
pvcreate /dev/sdc1
```

```
vgextend MyFirstVG /dev/sdc1
```

Вот и все: группа томов теперь имеет много больше свободного пространства, и вы можете увеличивать или добавлять логические тома, когда понадобится. Говоря о свободном пространстве — как узнать, сколько свободного места у вас в VG и чем занято остальное? Чтобы увидеть подробности своей VG, запустите `vgdisplay MyFirstVG`.

Если опустить имя VG, будут показывать данные по всем группам. Если вы всего лишь хотите увидеть, сколько места используется и свободно, лучше подойдет более короткий вывод `vgfs`. Есть эквивалентные команды `pvdisplay`, `pvs`, `lvdisplay` и `lvs` для

## Пробная установка

Если вы хотите поэкспериментировать с LVM, можно использовать внешний жесткий диск, но через USB будет меньше скорость. Однако простое решение для быстрого тестирования — использование файлов образов диска, если, конечно, у вас есть свободное пространство:

```
dd if=/dev/zero of=somefile0 bs=1 count=1 seek=10G
```

```
losetup /dev/loop0 somefile0
```

```
dd if=/dev/zero of=somefile1 bs=1 count=1 seek=10G
```

```
losetup /dev/loop1 somefile1
```

При этом создадутся два файла, а затем блочные устройства для доступа к этим файлам. Теперь можно использовать `/dev/loop0` и `/dev/loop1` вместо `/dev/sdb1` и `/dev/sdc1` в примерах. При перезагрузке вы потеряете устройства `loop`, но не файлы, поэтому в этом случае просто запустите команды `losetup` снова.

физических и логических томов. Без параметров каждая из этих команд выводит все соответствующие объекты, или можно дать им список того, что вы хотите видеть.

Мы увидели, как легко добавить новое устройство; а его удаление требует чуть больше работы, но не намного. Скажем, вы добавили гораздо больший диск `/dev/sdc1` и теперь намереваетесь удалить `/dev/sdb` — быть может, потому, что он устарел и тормозит. Во-первых, перетащите все данные со старого диска на новый: `pvmove /dev/sdb1` переместит все данные из указанного PV в другие тома в группе — можно указать, в какие именно, если их там много, но обычно в этом нет необходимости, разве что вы хотите удалить больше одного диска. Теперь с помощью `vgreduce` удалите PV из группы томов: `vgreduce MyFirstVG /dev/sdb1` или `vgreduce MyFirstVG --all`. Второй вариант удаляет все неиспользуемые физические тома.

Многие инсталляторы дистрибутивов теперь умеют применять LVM при разметке диска — в Fedora это по умолчанию уже несколько лет. Если вы хотите загрузиться с корневой файловой системы на LVM, надо будет использовать `initramfs`, и установщик дистрибутива позаботится об этом. Если вы устанавливаете самостоятельно, *Dracut* [см. Учебники, стр. 78 LXF196] позволит создать `initramfs`, которая загружается с LVM. Еще много чего наберется об LVM, о чем свидетельствует несколько map-страниц, но мы рассмотрели достаточно, чтобы быть в состоянии использовать и понимать его. LXF

Module Config Search Docs..














### Logical Volume Management

LVM version 2.02.111

**Volume Groups   Physical Volumes   Logical Volumes**

A logical volume is a virtual partition created from the combined disk space of a volume group. Each can have a filesystem which is then mounted to store files.

Create a logical volume in **bulk**. | Create a snapshot in **bulk**. | Create a logical volume in **huge**. | Create a snapshot in **huge**.  
 | Create a logical volume in **fast**. | Create a snapshot in **fast**. | Create a logical volume in **safe**. | Create a snapshot in **safe**.

 <b>backup</b> 1000 GB	 <b>bulk</b> 976.56 GB	 <b>data</b> 80 GB	 <b>home</b> 195.31 GB
 <b>home</b> 80 GB	 <b>root</b> 19.53 GB	 <b>rootbck</b> 5 GB	 <b>storage</b> 1.17 TB
 <b>swap</b> 11.72 GB	 <b>tmp</b> 10 GB	 <b>usr</b> 9.77 GB	 <b>var</b> 19.53 GB
 <b>www</b> 30 GB			

Create a logical volume in **bulk**. | Create a snapshot in **bulk**. | Create a logical volume in **huge**. | Create a snapshot in **huge**.  
 | Create a logical volume in **fast**. | Create a snapshot in **fast**. | Create a logical volume in **safe**. | Create a snapshot in **safe**.

► *Webmin* предоставляет некоторые web-инструменты для графического администрирования LVM.

# RAID: Создаем массивы дисков

Не опускаясь до каламбуров про спреи от насекомых, **Нейл Ботвик** показывает, как заставить несколько дисков летать и как исправлять в них неполадки.



**Наш эксперт**

Нейл Ботвик имеет большой опыт работы с загрузкой, ведь у него по компьютеру в каждой комнате, и гораздо меньше — с перезагрузкой, поскольку он перешел на Linux.



**М**ы рассмотрели использование LVM [см. стр. 60] для управления разметкой и несколькими дисками. Имеется и другая технология управления несколькими дисками, называемая RAID. Эта аббревиатура расшифровывается как Redundant Array of Inexpensive Disks — избыточный массив недорогих дисков, объединяющий несколько дисков в один блок для повышения производительности или избыточности на случай

```
[neil@hactar ~]$ sudo mdadm --detail /dev/md6
/dev/md6:
  Version : 1.2
  Creation Time : Wed Nov 11 17:00:18 2015
  Raid Level : raid1
  Array Size : 10477568 (9.99 GiB 10.73 GB)
  Used Dev Size : 10477568 (9.99 GiB 10.73 GB)
  Raid Devices : 4
  Total Devices : 4
  Persistence : Superblock is persistent

  Update Time : Wed Nov 11 17:02:59 2015
  State : clean, resyncing
  Active Devices : 4
  Working Devices : 4
  Failed Devices : 0
  Spare Devices : 0

  Resync Status : 68% complete

  Name : hactar:6 (local to host hactar)
  UUID : c592ada0.a40b94a7:5973f52:d6e831da
  Events : 11

   Number Major Minor RaidDevice State    active sync   /dev/loop0
    0       7       0       0   active sync   /dev/loop1
    1       7       1       1   active sync   /dev/loop2
    2       7       2       2   active sync   /dev/loop3
    3       7       3       3   active sync   /dev/loop3
[neil@hactar ~]$ cat /proc/mdstat
Personalities : [raid0] [raid1] [raid10] [raid5] [raid6] [raid4]
md6 : active raid1 loop3[3] loop2[2] loop1[1] loop0[0]
      64232640 blocks [64232640/64232640] 64232640x4
```

» Команда `mdadm --detail` дает много информации о массиве, а `/proc/mdstat` более лаконичен.

отказа. Простейший RAID — два диска в массиве RAID 1. В этом случае два диска являются зеркалами друг друга. Все записи делаются на обоих дисках (производительность обеспечивается буферизацией), а чтение выполняется с того диска, который находится в большей готовности, чем данный процесс, и ускоряется. Это означает, что вы получите небольшой прирост в скорости чтения, незаметную разницу в скорости записи и тот же объем, что и у одного диска. Выгода — в безопасности данных. Поскольку все записывается на оба диска, то при неисправности одного данные уцелеют на другом. Кроме того, когда вы удалите неисправный диск и поставите в массив новый, данные автоматически скопируются в фоновом режиме, и безопасность данных в виде двух копий быстро восстановится.

## Типы RAID

Существуют различные уровни RAID, характеризующие способ распределения данных по дискам (см. врезку *Уровни RAID вкратце*). Есть также три различных способа реализации RAID: аппаратный RAID, программный RAID и FakeRAID. Аппаратный RAID, как следует из названия, реализуется чисто аппаратно: либо платой контроллера, либо на материнских платах некоторых серверных систем. Независимо от количества дисков, подключенных к аппаратному RAID, в ОС отображается только один. Аппаратный RAID быстр, но имеет два недостатка: он дорог и использует собственный формат диска, т.е. при выходе контроллера из строя понадобится совместимая замена, чтобы иметь возможность читать ваши диски.

Программный RAID реализован программно и включен в ядро Linux. На современном оборудовании его производительность сравнима с аппаратным RAID, но он предлагает гораздо более гибкое управление и возможность чтения дисков на другой системе. Здесь мы рассмотрим именно этот вариант.

## Пробная установка

Если вам хочется немного поэкспериментировать с RAID, не затрагивая разделы своего жесткого диска, можно создать несколько виртуальных дисков следующим образом.

```
for i in {0..3}; do
  dd if=/dev/zero of=diskfile$i bs=1 count=1
  seek=10G
  losetup /dev/loop$i diskfile$i
done
```

done  
При этом сперва создаются четыре файла, а затем — блочные устройства для доступа к этим файлам. Теперь можно использовать в примерах `/dev/loop0–/dev/loop3` вместо дисков. При перезагрузке вы потеряете устройства `loop`, но не файлы, и в подобном случае просто запустите команды `losetup` снова.



## Уровни RAID вкратце

Существует несколько способов объединения дисков в RAID, и у каждого свои преимущества и недостатки. Далее N означает количество устройств в массиве, а S — объем каждого из них. Наиболее распространенные уровни:

» **RAID 0** Не совсем RAID, поскольку это просто объединение нескольких дисков. В таком случае лучше использовать LVM. Отсутствует отказоустойчивость. Общий объем хранилища —  $N \cdot S$ .

» **RAID 1** Два или более дисков, являющихся полными копиями друг друга. Способен выдерживать отказ N-1 дисков. Общий объем хранилища составляет S.

» **RAID 5** RAID из трех или более дисков с таким распределением данных и информации о четности, что при отказе одного из дисков (любого) данные сохраняются. Общий объем хранилища в этом случае равен  $(N-1) \cdot S$ .

» **RAID 6** Четыре или более дисков с распределением данных и информации о четности так, что при отказе любых двух дисков данные сохраняются. Общий объем хранилища —  $(N-2) \cdot S$ .

» **RAID 10** Массив RAID 1 массивов RAID 0, требующий как минимум четырех дисков. Может выдержать несколько отказов, пока RAID 1 не потеряет все свои диски. Здесь общий объем хранилища —  $(N/2) \cdot S$ .

Если для материнской платы заявлена поддержка RAID, а плата дешевле сотен фунтов, это скорее всего будет так называемый «ложный RAID», он же fakeRAID. Контроллер выполняет часть операций для RAID, загружает драйвер с дисков, потом становится программным RAID. Это, как правило, работает только с Windows.

Мы несколько раз упоминали диски, но программный RAID может работать с любым блочным устройством и часто реализуется на уровне разделов, а не дисков. Хватит разговоров — давайте создадим RAID 1 на `/dev/sda3` и `/dev/sdb3` (измените имена устройств в соответствии с вашей системой). Эти разделы должны быть запасными, или можно использовать файлы образов (как описано во врезке *Пробная установка*, с. 62). Поскольку мы работаем с файлами устройств в каталоге `/dev`, вы должны быть суперпользователем, так что или откройте `root`-терминал, или предваряйте каждую команду `sudo`. Основная команда для работы с устройствами с программным RAID — `mdadm`.

```
mdadm --create /dev/md0 --level=raid1 --raid-devices=2 /dev/sda3 /dev/sdb3
```

Для экономии на наборе можно использовать

```
mdadm -C /dev/md0 -l 1 -n 2 /dev/sd{a,b}3
```

но здесь для наглядности мы будем придерживаться длинных вариантов. Вы создали блочное устройство в `/dev/md0` (программные RAID-устройства, как правило, называются `/dev/mdN`), которое можно форматировать и потом монтировать, как любое другое блочное устройство:

```
mkfs.ext4 /dev/md0
```

```
mount /dev/md0 /mnt/somewhere
```

### Когда что-то не так

Надеюсь, это и всё, что нужно сделать — ваш массив создан и используется как обычный диск средствами ОС; а если у вас отказывает привод? Состояние RAID можно увидеть в любое время любой из этих команд:

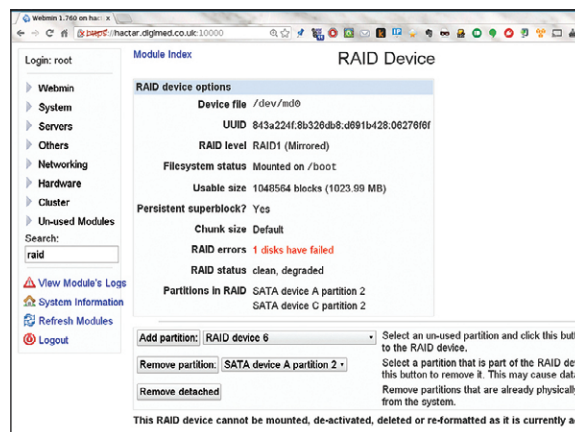
```
cat /proc/mdstat
```

```
mdadm --detail /dev/md0
```

Предположим, что отказал второй диск, а у вас есть другой ему на замену. Удалите старый диск из массива:

```
mdadm /dev/md0 --fail /dev/sdb3 --remove /dev/sdb3
```

Затем выключите компьютер, замените диск и перезагрузите. Массив по-прежнему будет работать, но `/proc/mdstat` покажет, что он деградировал, поскольку устройство отсутствует. Теперь запустите `mdadm /dev/md0 --add /dev/sdb3` и снова взгляните в `/proc/mdstat`. Он покажет, что массив теперь опять с двумя устройствами и что уже идет синхронизация данных на новом диске. Можно продолжать использовать компьютер, хотя пока выполняется синхронизация, производительность диска падает. Если у вас есть диск, который даже больше не отображается, как в случае полного отказа, нельзя удалять `/dev/sdb3`, потому что его больше не существует: используйте слово `missing` вместо имени диска, и `mdadm` удалит все диски, которые не сможет найти. Если у вас уже имеется резервный диск в компьютере, скажем, `/dev/sdc`,



» Обычно RAID управляют с помощью `mdadm` в командной строке, но есть и модуль RAID для *Webmin*, если вам нужен графический вариант.

можете добавить его в массив в качестве запасного: `mdadm /dev/md0 --add-spare /dev/sdc3`. Если `sdb` откажет и будет удален, как указано выше, `sdc3` автоматически добавится в массив на его место и синхронизируется. Все эти примеры касаются использования RAID 1, но процедуры, кроме первоначального создания массива, одинаковы для всех более высоких уровней RAID.

### Контроль неисправностей

А как узнать, что диск отказал — надо постоянно смотреть `/proc/mdstat`? Конечно, нет: у `mdadm` есть также режим контроля массивов, работающий как сервис. Все, что вам нужно сделать — это настроить его в `/etc/mdadm.conf`: найдите строку, содержащую `MAILADDR`, укажите свой email и удалите символ `#` из начала строки. Теперь установите запуск службы `mdadm` при загрузке, и он будет следить за состоянием RAID и уведомлять вас о всех проблемах. Файл настройки `/etc/mdadm.conf` также определяет, какие устройства принадлежат к какому массиву. По умолчанию при запуске диски сканируются, чтобы определить компоненты массива, но можно указать их явно в строке `ARRAY`. Создается эта строка командой `mdadm --examine --scan`. Это может быть полезно, если к вашей системе подключены одно или несколько медленных устройств, замедляющих процесс сканирования.

В приведенных выше примерах мы создали RAID из разделов, но можно также создать массив из целых дисков — например, такой RAID 5 из трех дисков:

```
mdadm --create /dev/md0 --level=raid5 --raid-devices=3 /dev/sd{a,b,c}
```

Создав такой массив, можно использовать `gdisk` или `gparted` для его разметки, подобно физическому диску; разделы будут обозначаться как `/dev/md0p1` и т.д. Имейте в виду, что BIOS потребует директорию `/boot` на файловой системе, которую она может читать, поэтому RAID из целых дисков может не подойти для диска с вашей ОС. RAID также хорошо работает с LVM. Создайте RAID и затем используйте его как физический том для LVM. Тогда вы получите гибкость LVM и безопасность данных RAID. **LXF**

# GParted: Как разметить диск

Ник Пирс откроет, как добавить разделы на жестком диске, чтобы защитить свои данные и упростить работу в Linux.



Наш эксперт

Ник Пирс подсел на Linux — сначала на Ubuntu, а теперь Minibian — вот уже 10 лет как. Раньше он предпочитал графические инструменты, но теперь в нем нарастает любовь к терминалу.



Разметка позволяет разбить один физический жесткий диск на несколько меньших по размеру виртуальных дисков, также именуемых разделами или томами (хотя тут есть разница, как мы объясним позже). Каждый том существует независимо от других, определяя уровень избыточности для данных, имеющихся на диске. Если с одним из разделов возникают проблемы, его можно восстановить, не затрагивая остальную систему.

Такого рода избыточность — первая из причин, зачем нужна разметка диска: она позволяет перемещать ваши личные данные (как правило, папку **home**) на другой том, защищая ее от изменений системного раздела. Тогда вы сможете превратить свою установку Linux в тестовую среду, откатывая любые нежелательные изменения без вреда для ваших данных.

Разметку также часто применяют для запуска нескольких операционных систем на одном компьютере. Можно, например, установить параллельно две версии Linux, или сделать машину с двойной загрузкой Windows/Linux в целях совместимости, и т. д. Это возможно, поскольку каждый раздел можно отформатировать под свою файловую систему — ext3 или ext4 для Linux, или NTFS для Windows. Затем можно создать раздел с данными, видимый для обеих операционных систем, чтобы иметь доступ к самым свежим версиям ваших документов и других файлов независимо от ОС, в которой вы работаете.

Можно использовать разметку диска и для других целей, например, создание специального раздела для обмена по сети или независимых разделов для временных файлов, файла подкачки

или даже запуска некоторых сервисов, например, web-сервера. В чем бы ни состояла ваша цель, разделы могут сыграть решающую роль для улучшения производительности, а также защиты различных частей вашего диска друг от друга.

## Типы разделов

Разбиение жестких дисков производится по определенной схеме. Есть два основных типа схем: та, что постарше, называется Master Boot Record (MBR), она подходит для дисков объемом до 2 ТБ, а та, что поновее — GUID Partition Table (GPT). В обоих случаях информация о разделе физически записывается в первом секторе диска в таблице разделов — GPT-диски заодно дублируют эту информацию на другом конце диска, в последнем секторе, для обеспечения избыточности.

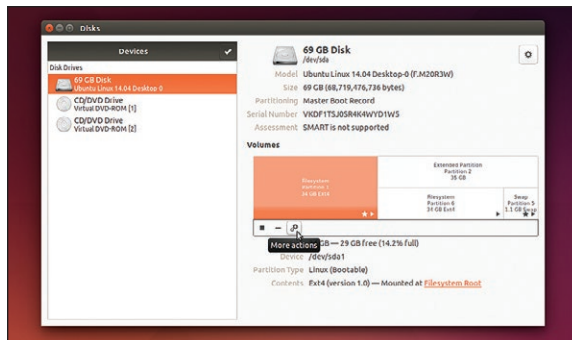
Один из недостатков старой схемы MBR в том, что в ней допускается создавать максимум четыре раздела; правда, один из них может быть расширенным и вмещать несколько логических томов, что позволяет преодолеть этот лимит. В GPT отсутствует деление на основные и расширенные (логические) разделы и поддерживается до 128 разделов на диске.

GPT разработан в соответствии с современной прошивкой UEFI, но в некоторых случаях имеет обратную совместимость с более старыми унаследованными системами BIOS. Если объем вашего диска более 2 ТБ, то это принудительно, хотя в конечном итоге схема разделов, используемая у вас, определяется самой системой. Для простоты лучше придерживаться того, что есть, и если вы планируете полностью очистить диск и установить Linux с нуля, пусть ваш дистрибутив сам решит, какая схема лучше подходит для текущей установки; с нею вы и будете работать после инсталляции.

На нашем уроке мы рассмотрим разметку с применением MBR, но в GPT процесс аналогичен. Мы используем Ubuntu 14.04.3 LTS,

## Скорая помощь

Если вы планируете пользоваться разделом совместно с Windows, то это должен быть раздел NTFS, чтобы Windows могла с ним работать. Убедитесь, что установлен пакет *ntfs-3g*, позволяющий Linux писать на разделы такого типа.



Утилита **Диски** представляет удобный обзор существующих разделов.

## Работа с LVM

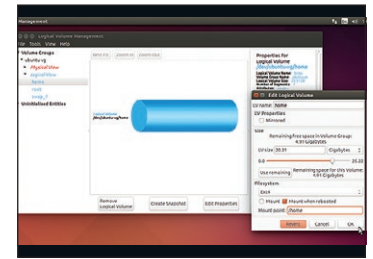
Если вы установили свою систему Linux с LVM, то при попытке изменить структуру разделов с помощью *GParted* вам придется малость понервничать — это просто не получится. Хотя это и не проблема: нужен только правильный инструмент.

Можно, конечно, настроить разделы из командной строки, но лучше всего будет воспользоваться удобной утилитой *LVM*. Чтобы обзавестись ею, загрузитесь со своего Live CD, откройте *Software Center* и наберите в поиске 'LVM'. Выберите Logical Volume Management и щелкните Подробнее > Использовать этот источник > Установить.

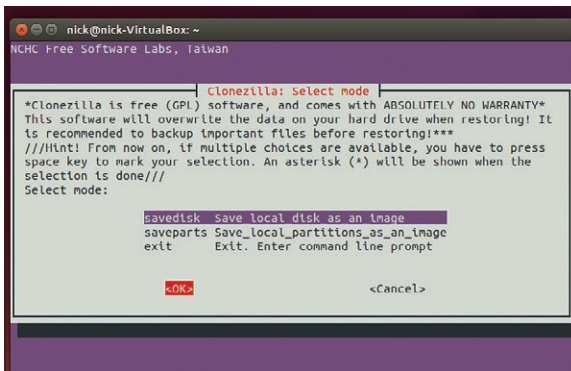
После установки запустите *LVM* — и вы окажетесь в более приятной рабочей среде.

Начните с изменения размера корневого раздела, чтобы освободить доступное пространство: разверните Логический вид, выберите 'корневой' и нажмите Изменить свойства. Используйте элементы управления, чтобы уменьшить раздел до нужного размера, нажмите кнопку ОК и подождите, пока идет изменение. Потом можете переключиться на Логический вид и нажать Создать логический том, чтобы добавить столько разделов, сколько хотите, не забывая давать им понятные имена (например, «дом» для вашего раздела *home*).

Настраивать точку монтирования на данном этапе не надо — вместо этого выполните шаги с четвертого по шестой из пошагового руководства (см. стр. 67), чтобы завершить настройку вновь созданного раздела.



» Утилита *LVM* (или *system-config-lvm*, по названию пакета) — самый простой инструмент разметки для систем с функцией *LVM*.



» Перед разметкой не забудьте сделать резервную копию всего вашего диска. Попробуйте *Clonezilla*.

но то же должно воспроизвестись на большинстве дистрибутивов Linux.

То, как ваш компьютер разбит на диски, зависит от индивидуальных факторов, ну и, конечно, от используемого дистрибутива. Чтобы увидеть текущую разметку и узнать, какой тип схемы разделов (MBR или GPT) применяется у вас, откройте утилиту *Диски* из меню Dash. Выберите свой диск из левой панели и откройте вкладку Разделы, чтобы увидеть, какая применяется схема.

Ниже расположен график Тома — визуальное представление вашего диска со всеми его разделами. В установке Ubuntu 14.04, например, по умолчанию, таких разделов два: основной раздел Linux, загрузочный, со всеми вашими приложениями, данными и настройками в дополнение к дистрибутиву Linux. И другой, расширенный раздел, гораздо меньший по размеру, в котором содержится файл подкачки.

О каждом разделе на вашем жестком диске есть отдельная запись в папке */dev*, где хранятся ссылки на все компоненты вашего компьютера. Каждый физический жесткий диск представлен тремя буквами: *hda*, *hdb* и *hdc* и т.д. для накопителей, подключенных к более старым контроллерам IDE, и *sda*, *sdb*, *sdc* для тех, что подцеплены к SCSI и новым контроллерам SATA.

Разделы всех физических дисков пронумерованы — например, как *sda1* или *sdb5*. Чтобы узнать, какой номер был закреплен за выбранным томом, посмотрите в Дисках на пункт Устройство. Сложный момент: возможно, при установке вам предлагалось использовать LVM (Logical Volume Management) — если вы согласились, то на традиционных установках на базе MBR вы увидите небольшой первичный раздел, обозначенный как Linux (Загрузочный), и внутри него другой, расширенный раздел, гораздо больше — это том Linux LVM. Первичный раздел нужен потому, что

расширенные разделы не являются загрузочными, так что необходимые загрузочные файлы находятся на этом небольшом разделе, а сама система — внутри расширенного тома (см. *Работа с LVM*, вверху).

## Структура раздела

Доступ к разделам осуществляется через файловую систему путем монтирования раздела на определенном уровне. Linux организует свои файлы, папки и разделы в виде дерева [снизу вверх, — прим. ред.]. При загрузке Linux сначала монтирует основной раздел в */* — корне (или стволе) дерева. Другие разделы могут затем монтироваться в определенных точках под ним — например, разделы, предназначенные для совместного использования, неплохо смонтировать в папки */mnt* или */media*. Там же автоматически монтируются внешние разделы — например, другой ОС — если файловая система такого раздела распознается Linux.

При необходимости, разделы можно также монтировать прямо в ключевых каталогах, таких как */tmp* (временные файлы) или */home* (домашние папки каждого пользователя на компьютере) — в этом случае их присутствие совершенно не ощущается, независимо от того, создаете вы специальные разделы для ключевых папок, или нет. Точки монтирования хранятся в файле */etc/fstab*.

Лучший момент для разметки диска — это первичная установка Linux на пустом жестком диске (о том, как это легко, см. *Создание разделов с нуля*, стр. 66).

Также не обойтись без создания разделов при настройке системы с двойной загрузкой. В этом случае все будет гораздо проще, если ваша существующая ОС обнаружится в процессе установки — тогда вся трудная работа по разметке диска будет выполнена за вас, поскольку инсталлятор сам предложит подходящую структуру разделов, которые вы можете подстроить по своему усмотрению, смотря сколько у вас свободного места на диске.

Для большинства людей это будет не самый разумный вариант — и к счастью, есть специальные инструменты, позволяющие изменить существующую структуру разделов без потери данных, хотя важно отметить, что элемент риска всегда есть. Вот почему стоит создать образ вашего диска сейчас, чтобы потом, если что-то пойдет не так, можно было бы все снести и начать заново. Используйте инструмент, подобный *Clonezilla* (<http://clonezilla.org>), или сделайте моментальный снимок системы [snapshot], если работаете в виртуальной машине. Также стоит сесть и разобраться, что именно вам нужно и сколько места на диске есть в вашем распоряжении. В первую очередь, решите, какие дополнительные разделы вы хотите создать и зачем. Если для того, чтобы защитить их от переустановки или повреждения основного диска, то можно разместить их на том же самом разделе; помните также, что хотя

## Скорая помощь

Еще одно преимущество LVM перед обычной разметкой в том, что можно добавлять дополнительные физические диски к вашему компьютеру, приплюсовывая место к существующим томам. То есть увеличивать объем памяти без переноса и копирования данных с диска на диск.

## Скорая помощь



Можно добавить новый раздел в `fstab`, напрямую его не редактируя — для этого откройте утилиту Диски, выберите раздел и затем Другие действия > Редактировать опции монтирования... Однако при использовании этого метода нельзя добавить поле 'пропустить', чтобы диск при старте системы проверялся на наличие ошибок.

➤ Если система, которую вы хотите установить параллельно, распознается Ubuntu, процесс разметки идет легко.

это и обеспечивает некоторую защиту от потери данных, такая операция не является заменой резервного копирования.

Если вы хотите улучшить производительность за счет перемещения ключевых папок — файла `swap`, или, возможно, папки `usr`, где хранятся программы — тогда их нужно будет разместить на физическом диске, отдельно от того, на котором работает система Linux.

Также стоит учесть размер каждого раздела — это зависит от его типа и объема свободного места на диске. Для начала, щелкните правой кнопкой по нужной папке в файловом менеджере и выберите Свойства. Это технически минимальное пространство, которое надо выделить на раздел, но обдумайте, что еще может храниться в папке, и прибавьте соответствующий объем. Учтите, пространство для нового тома вы отнимаете от вашего основного раздела, потенциально ограничивая то, сколько места вы можете выделить.

Другие факторы, которые следует учесть: будет ли этот раздел основным или расширенным? Если вы планируете создать только один дополнительный раздел, тогда просто измените размер системного раздела и создайте вместо него новый основной раздел (как показано в пошаговом руководстве, стр. 67). Однако если вы планируете создать ряд разделов на основном диске, рекомендуется создать расширенный раздел (см. *Создание разделов с нуля*, внизу).

Вы также должны решить, какая файловая система будет использоваться в новом разделе. Если вы создаете отдельные разделы для папок вроде `home` или `var`, то система должна быть той же, что и в основном разделе Linux (как правило, `ext3` или `ext4`). То же касается разделов, где вы планируете папки для совместного использования через сеть. На самом деле, менять тип файлов надо только в том случае, если вы хотите параллельно установить систему Windows на том же ПК; в этом случае выберите FAT32 или NTFS. Когда дело доходит непосредственно до разметки вашего

диска, лучший инструмент для этого уже есть на Ubuntu Live CD, и это *GParted*, дружелюбный графический интерфейс, позволяющий легко менять размер разделов (причем с удобным графическим представлением структуры вашего жесткого диска). И коли вам все равно придется загружаться с него, чтобы изменить размер системного раздела, можете заодно создать разделы на свой вкус, находясь вне своей базовой установки Linux.

## Инструменты разметки

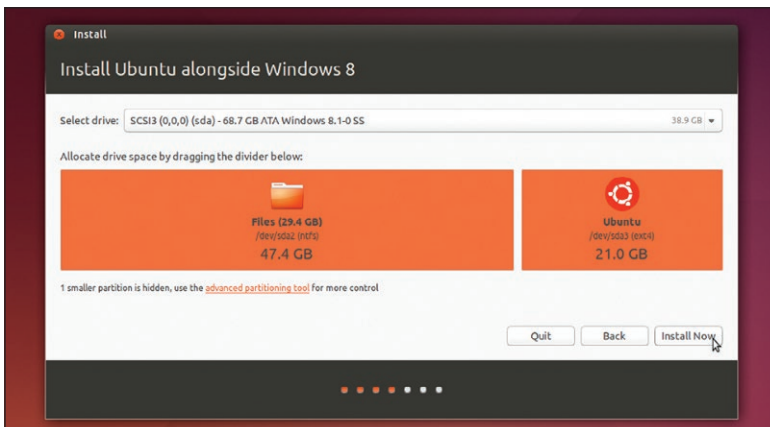
Вы также можете запустить *GParted* из его собственного загрузочного образа на CD или USB с [www.gparted.org/livecd.php](http://www.gparted.org/livecd.php). Убедитесь, что выбрана правильная архитектура — как правило, это `i586` для более старых, 32-разрядных компьютеров, и `amd64` для новых 64-битных машин с UEFI вместо BIOS. Пользователи 32-битных ПК могут также поэкспериментировать со сборкой `i686-PAE`, если покажется, что `i586` малость притормаживает.

В пошаговом руководстве (справа), показано, как уменьшить свой основной раздел, прежде чем создавать второй, параллельно с ним, с помощью *GParted* из Ubuntu Live CD. Если же вы хотите создать новый раздел или разделы внутри расширенного, процедура будет несколько иной. Во-первых, измените размер системного раздела, как описано в руководстве. После этого надо увеличить малый расширенный раздел, содержащий ваш файл `swap`, чтобы он занял все оставшееся свободное место.

Этого нельзя сделать в момент, когда файл `swap` используется, поэтому щелкните правой кнопкой мыши на раздел подкачки в *GParted* и выберите Отключить. После этого правым щелчком можете выбрать в расширенном разделе Изменить размер/Переместить. Введите 0 (ноль) в поле Свободное место перед разделом предшествующего окна и нажмите Изменить размер/Переместить, чтобы быстро выделить все доступное пространство для раздела. Теперь можете разделить весь свободный объем на сколько угодно разделов — только обязательно рассчитайте размер каждого раздела и распределите его соответственно, а закончив, не забудьте нажать Применить.

Инструменты, упоминавшиеся ранее, в целом довольно умны, но потенциально это опасная операция, поэтому важно предварительно сделать полную резервную копию системы. Если возникнут проблемы, обратитесь к справке по *GParted* на <http://gparted.org/help.php>. Там вы найдете полезные ссылки на FAQ, а также, среди прочего, руководство по решению проблем с *Grub*.

Вы можете столкнуться с проблемами при попытке монтирования новых разделов в ключевых папках, таких как ваша папка `home`. Если Linux выдает ошибку на старте, нажмите `m` для восстановления системы вручную, а затем введите `sudo nano /etc/fstab`, чтобы изучить файл и проверить, что в нем нет ошибок, препятствующих правильному построению раздела. Если проблем не обнаружится, удалите строку, сохраните файл и перезагрузитесь, прежде чем с этим разбираться.



## Создание разделов с нуля

Лучше всего размечать диск во время новой установки системы. В Ubuntu, когда будет предложено стереть диск, выберите Дополнительно и нажмите Продолжить. Вы увидите, что на данный момент есть только одно устройство — `/dev/sda`. Нажмите кнопку Изменить таблицу разделов..., прочтите предупреждение и нажмите кнопку Продолжить. Теперь выберите свободное пространство и нажмите кнопку + рядом с Изменить, чтобы создать раздел.

Начнем с главного системного раздела. Он может занимать всего 15–20 ГБ, но при необходимости

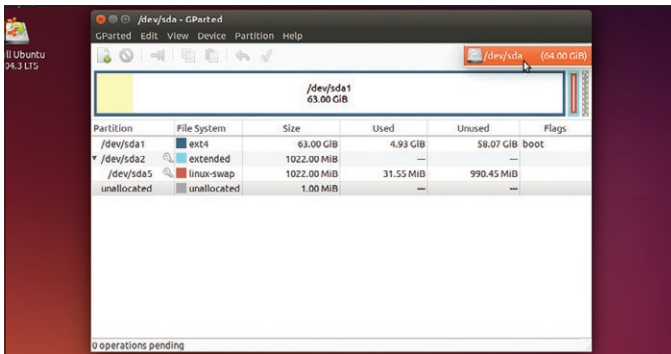
можно сделать больше. Подсчитав, сколько гигабайт вам требуется, умножьте это на 1024 и введите полученную цифру в поле Размер. Оставьте помеченными Основной, Начинать отсюда и Файловая система `Ext4`, выберите / как точку монтирования и нажмите OK.

Выберите оставшееся свободное пространство и снова нажмите +. Нажмите кнопку Использовать как и в выпадающем меню выберите «область подкачки». Установите тип раздела как Логический, но расположение укажите как Конец этого

пространства. Размер его (в МБ) может быть разным — как правило, он должен соответствовать объему ОЗУ вашего компьютера (увеличьте эту цифру вдвое, если у вас 1 ГБ или менее). Затем нажмите OK.

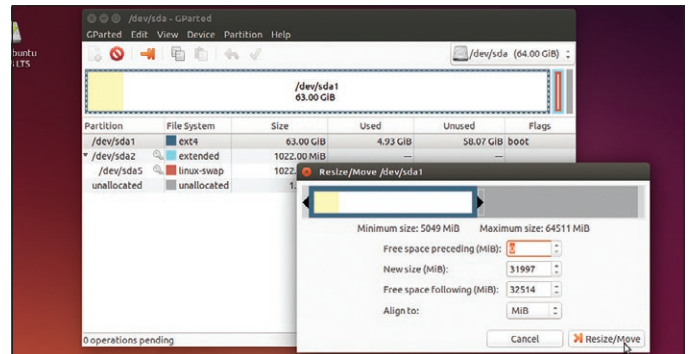
Наконец, снова выберите свободное пространство и еще раз нажмите +. Оставьте размер, тип, расположение и Использовать как неизменными и установите Точка монтирования в `/home`. Нажмите Установить сейчас и далее Продолжить, чтобы создать свои разделы и установить Ubuntu.

## Создаем отдельный домашний раздел с GParted



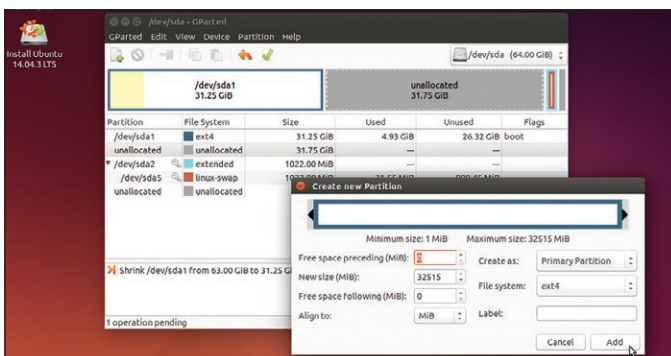
### 1 Запускаем GParted

Перезагрузитесь с вашего Ubuntu Live CD или съемного диска, выбрав вариант Попробовать Ubuntu, когда это будет предложено. Загрузив рабочий стол, запустите *GParted*, введя его имя в Dash. Если к вашему компьютеру подключено несколько физических дисков, позаботьтесь о том, чтобы выбрать правильный диск (скорее всего, этим диском будет `/dev/sda`), нажав на кнопку в правом верхнем углу окна *Gparted*.



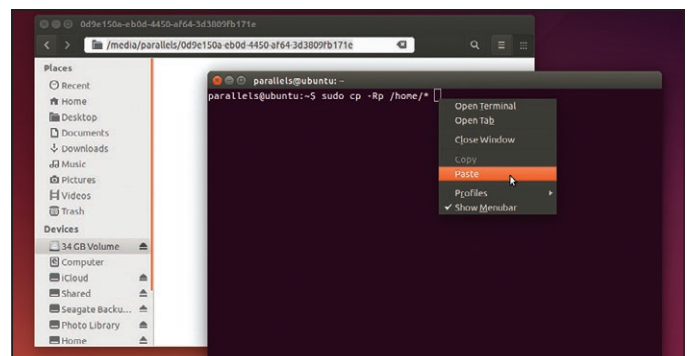
### 2 Изменяем объем системного раздела

Щелкните правой кнопкой на значке основного раздела (`/dev/sda1`) и выберите Изменить размер/Переместить. Нажмите и перетащите правую границу полосы, уменьшив размер примерно так, как вам надо, смотря сколько места вы планируете выделить под свой раздел `home`. При необходимости укажите размер точнее, через Свободное пространство после раздела (MiB). Нажмите Изменить размер/Переместить.



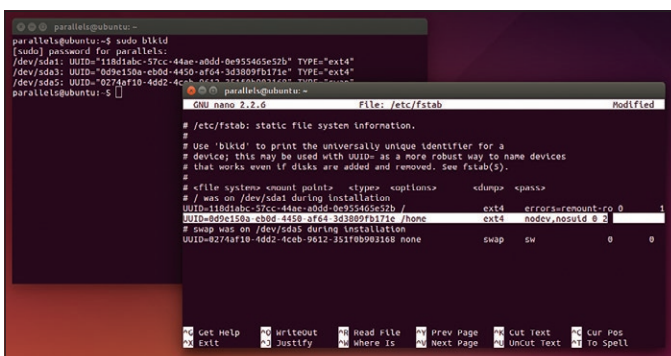
### 3 Создаем домашний раздел

Далее, щелкните правой кнопкой мыши на незанятом пространстве и выберите Новый раздел. Если это будет единственный добавочный раздел, оставьте настройки по умолчанию как есть и нажмите кнопку Добавить. Проверьте изменения, пока они еще не внесены, и если всё вас устраивает, нажмите на зеленую галочку и затем Применить. Ждите, пока *GParted* изменит основной раздел, а затем создаст новый.



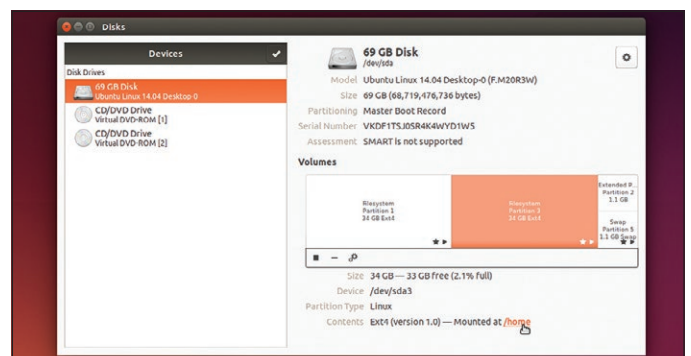
### 4 Копируем файлы в новый раздел

Перезагрузитесь в Ubuntu. Откройте Файлы и нажмите на новый том, появившийся под полем Устройства. Выберите Перейти > Указать расположение, затем выберите текущее расположение в адресной строке и далее правым щелчком Копировать. Откройте терминал, введите `sudo cp -Rp /home/*`; снова правый щелчок и затем Вставить, чтобы указать новый раздел. Нажмите Enter и игнорируйте все ошибки.



### 5 Обновляем fstab

После этого в разделе должны появиться файлы из вашей домашней папки. Далее, введите `sudo blkid` и нажмите Enter. Запомните UUID, введите `sudo nano /etc/fstab` и снова Enter. Введите следующую под первой строкой вида `UUID=<location>`, подставив UUID своего раздела: `UUID=<location>/home ext4 noudev,nosuid 0 2`.



### 6 Монтируем новый раздел

Внесите изменения в файл `fstab`, сохраните обновленный файл и закройте его, а затем введите в терминале следующую последовательность команд и нажмите Enter: `cd / && sudo mv /home /home_old && sudo mkdir /home`. Теперь надо перезагрузить компьютер, вновь открыть Диски и убедиться, что ваш новый раздел правильно установлен в `/home`. **LXF**

# Образ системы: Клон и запуск

Маянк Шарма создает образ системы и почти мгновенно развертывает его на нескольких компьютерах.



## Наш эксперт

**Маянку Шарме** некогда работать сисадмином: он слишком занят написанием статей для верных читателей LXF — о том, как настроить и управлять Linux для любых задач на многих системах.

Управление сетью компьютеров — процесс сложный. Прежде чем браться за проблему активного мониторинга машин, надо установить операционную систему на каждой из них. И даже для небольшой сети с десятком компьютеров это отнимет немало времени. Клонирование компьютера включает установку операционной системы, драйверов, программного обеспечения и данных на одном компьютере и автоматическое воспроизведение этого на других компьютерах. Этот метод, известный как репликация образов, используется системными администраторами для развертывания нескольких одинаковых машин через сеть без особых усилий. Fog, который мы будем использовать, является одной из самых популярных систем клонирования с открытым исходным кодом.

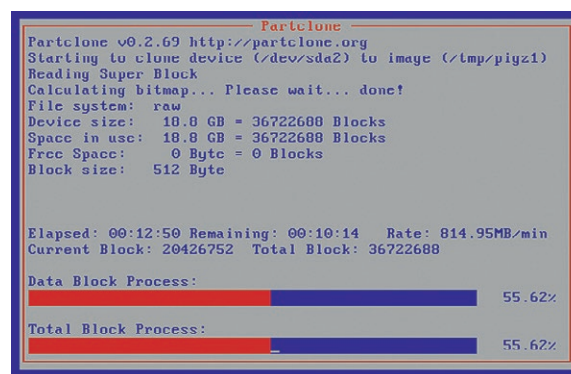
Чтобы использовать Fog, надо, во-первых, настроить сервер репликации. Проект официально поддерживает несколько релизов Ubuntu, Fedora, Debian и CentOS, но известно, что он работает и на других дистрибутивах. Перед установкой Fog позаботьтесь, чтобы сервер имел статический IP-адрес (устанавливаемый в разделе администрирования маршрутизатора). Для этого урока мы считаем, что адрес сервера Fog — 192.168.3.51. Также убедитесь, что все машины в вашей сети настроены на загрузку с сетевой карты. Наконец, убедитесь, что все DHCP-сервера в сети отключены, а сервер Fog настроен как DHCP-сервер и раздает адреса всем компьютерам сети.

Настроив свою сеть, скачайте последний стабильный релиз Fog (<http://sourceforge.net/projects/freeghost/files/FOG/>) на машину, которая будет сервером развертывания. Затем запустите терминал и распакуйте скачанный архив:

```
tar xvf Fog_1.2.0.tar.gz -C /opt
```

После этого переберитесь в каталог bin, извлеченный из архива, и запустите установочный скрипт:

```
sudo ./installFog.sh
```



» Fog зависит от нескольких зрелых инструментов с открытым исходным кодом, таких как partclone для создания образа компьютера.

Скрипт предложит вам бит-другой информации: используемую вами версию Linux, тип установки, IP-адрес сервера, роутер и DNS-сервер, и хотите ли вы установить DHCP-сервер самого сервера Fog. В большинстве случаев лучше оставить настройки по умолчанию, предложенные установщиком; только введите правильные IP-адреса для сервера.

Скрипт установит всякие необходимые компоненты. После этого отобразится URL панели управления Fog (наподобие [192.168.3.51/fog/management](http://192.168.3.51/fog/management)). Откройте ссылку в web-браузере и войдите в систему с учетными данными по умолчанию (fog:password). При первом запуске надо загрузить параметры по умолчанию в базу данных сервера, нажав на кнопку на странице. Первым делом на должной административной панели надо создать нового пользователя. Для этого воспользуйтесь User Management > Create New User.

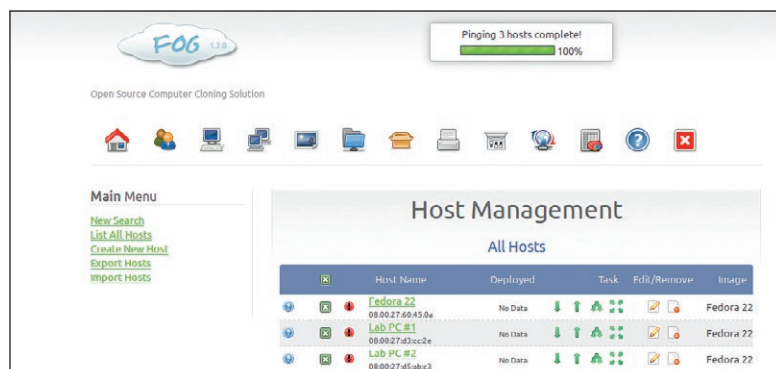
## Создание базового образа

Установив сервер репликации, применим его для создания образа компьютера. Когда образ будет готов, его можно будет развернуть на других компьютерах одним щелчком.

Чтобы начать этот процесс, запустите браузер на сервере репликации, откройте панель управления Fog и войдите с учетными данными по умолчанию. Затем используйте Image Management > Create New Image. Заполните поля формы описания образа: например, предположим, что мы создаем образ установки Fedora Workstation 22, который затем установим на все компьютеры в научной лаборатории. Тогда можно назвать образ 'Fedora для научной лаборатории'.

Далее, используйте выпадающее меню Operating System, чтобы указать операционную систему этого образа — например, Linux.

» Web-ориентированная панель управления сервера Fog облегчает управление сложными сетевыми развертываниями.



## Расширенные возможности Fog

*Fog* является сложным программным обеспечением, и хотя мы рассмотрели основные функции сервера, у него их несколько больше. Сервер *Fog* масштабируем и может управлять большими сетями в целом здании или по всему миру. Он позволяет разбить хосты на несколько групп для более легкого управления. Одной из наиболее полезных функций сервера *Fog*, особенно для администраторов крупных сетей, является возможность многоадресной рассылки. С помощью этой функции можно единым духом развернуть несколько машин.

Но чтобы успешно использовать эту возможность, необходимо убедиться, что у вашего хоста *Fog* достаточно вычислительных и сетевых ресурсов

на передачу нескольких образов одновременно. Для крупных сетей можно иметь несколько инсталляций *Fog*, настроенных как серверы хранения данных. Эти серверы хранения обмениваются образами и берут на себя нагрузку главного сервера *Fog* при клонировании компьютеров. Распределенные серверы хранения также ускоряют одноадресные передачи и вводят избыточность данных.

Помимо двух важнейших задач сервера *Fog*, рассмотренных на нашем уроке (а это загрузка образов), можно создать несколько разных задач для любого из хостов в репозитории *Fog*: например, запустить задачу отладки, которая загружает образ Linux в окружение *Bash* для исправления

любых ошибок при загрузке. Также можно создать задачу удаленной очистки хостов, для восстановления файлов с помощью *TestDisk* или сканирования на наличие вирусов с помощью *ClamAV*.

Сервер *Fog* может также установить и управлять принтерами в сети. В зависимости от операционной системы на хосте, можно также использовать сервер для отслеживания доступа пользователей к компьютерам по именам пользователей в Windows, автоматически отключать сеансы пользователей и выключать компьютер по истечении заданного периода бездействия. *Fog* может также устанавливать и удалять приложения через snapins — модули оснастки.

Наконец, выберите правильную схему разметки диска из выпадающего меню *Image Type*. Наша установка Fedora — на одном диске с несколькими разделами, поэтому выберем второй вариант. Предполагая, что вы уже установили Fedora на один из компьютеров в сети, запустите этот компьютер. Поскольку компьютер настроен на загрузку с сетевой карты, он покажет загрузочную среду PXE [Preboot eXecution Environment] сервера *Fog*. Прокрутите меню *Fog* и выберите пункт *Quick Registration and Inventory*. Сервер *Fog* теперь просканирует компьютер и добавит его в репозиторий известных хостов.

## Загрузка образа

Все подготовив, выключите компьютер с Fedora и вернитесь к серверу *Fog*. Откройте панель управления и выберите *Host Management* > *List All Hosts*. В списке будет отображен сервер Fedora. По умолчанию *Fog* определяет каждый хост по MAC-адресу, вместо которого можно задать более осмысленное имя (например, 'Fedora 22'), нажав на значок *Edit*: здесь можно изменить имя компьютера и добавить краткое описание, чтобы идентифицировать этот компьютер. Главное, используйте выпадающее меню *Host Image* и выберите ранее созданный образ Fedora 22.

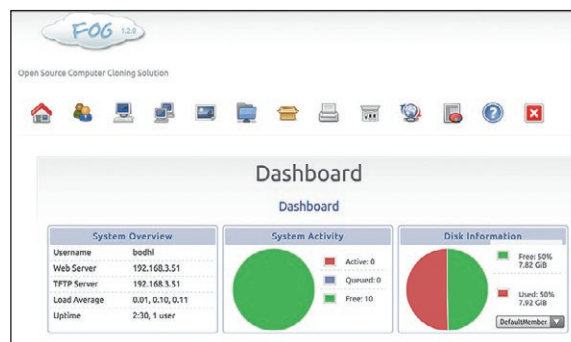
Итак, основа подготовлена; пора создавать образ установки Fedora. Вернемся в *Task Management* > *List All Hosts*, где имеется установка, переименованная в Fedora 22. В разделе *Task*, соответствующем этому образу, нажмите на зеленую стрелку загрузки. *Fog* даст вам несколько вариантов для планирования задачи загрузки. Можно изучить эти варианты, приобретая некоторый опыт с *Fog*, но сейчас лучше оставить параметры по умолчанию, для быстрого развертывания.

Затем вернитесь к машине с Fedora и загрузитесь. Снова обнаружится PXE *Fog*, будет автоматически создан образ машины и закачан на сервер *Fog*. Этот процесс займет некоторое время, зависящее от размера клонируемого диска, производительности используемых компьютеров и скорости локальной сети.

```
Host is registered as Fedora 22
```

```
-----
Boot from hard disk
Run Memtest86+
Update Product Key
Quick Image
Quick Host Deletion
Client System Information (Compatibility)
```

➤ Когда хост зарегистрирован, можно запросить об его оборудовании и получить информацию о совместимости перед снятием образа.



➤ Можно развернуть и создать образ ваших компьютеров, заходя в панель управления *Fog* с мобильного устройства вроде планшета.

После загрузки образа компьютер с Fedora перезагрузится. Теперь можно использовать *Fog*, чтобы развернуть этот образ Fedora на всех лабораторных компьютерах одним нажатием кнопки! Аналогичным образом можно клонировать любой другой компьютер в сети, включая установки новой Windows 10.

Прежде чем вы сможете развернуть образ, нужно зарегистрировать целевые машины как хосты на сервере *Fog*. Процесс регистрации такой же, как и раньше. Загрузите новый компьютер из сети, что должно обнаружиться средой PXE *Fog*. Когда это произойдет, выберите пункт *Quick Registration and Inventory*.

Добавив компьютер в репозиторий компьютеров, известных *Fog*, войдите в панель управления *Fog* и выберите *Host Management* > *List All Hosts*. Нажмите на иконку 'edit', соответствующую добавленной машине, и задайте ей более информативное имя, что-нибудь вроде *Lab PC #1*. Опять же, не забывайте использовать выпадающее меню *Host Image* для выбора образа Fedora 22, который мы только получили с другого компьютера. Повторите процедуру, чтобы зарегистрировать на сервере *Fog* все компьютеры в лаборатории. Затем отредактируйте их в панели управления *Fog*, чтобы дать им значащие имена, и выберите образ Fedora в качестве образа хоста.

Теперь нам надо реплицировать образ Fedora на другие компьютеры лаборатории, что мы и сделаем, выбрав *Task Management* > *List All Hosts*. Просмотрите список хостов, чтобы найти компьютер, на который требуется развернуть систему, и выберите стрелку вниз при соответствующей опции *Deploy image*. Когда задача развертывания создается, включите лабораторный компьютер. Он автоматически определит задачу на сервере *Fog* и скопирует образ с сервера на локальную машину.

Когда это будет выполнено, вы получите зеркальную копию установки Fedora на компьютере в лаборатории. И, наконец, необходимо будет повторить процедуру развертывания Fedora на других компьютерах. LXF

# ZuluCrypt: Шифруем диски

Изолируйте свои данные новым способом, советует Маянк Шарма.



Наш эксперт

Маянк Шарма — человек очень скрытный. Он каждое утро шифрует свой заказ на латте в кафе местному бармену. А живет, может, и в Нью-Дели — кто его знает.

Вы можете контролировать доступ к данным на вашем компьютере через учетные записи пользователей и права доступа к файлам, но чтобы не позволить целенаправленному злоумышленнику добраться до ваших личных файлов, этого мало. Единственный надежный способ сохранить свои данные для себя — это использовать шифрование. Конечно, работать с зашифрованными данными сложно, но это сильно укрепит безопасность и сохранность ваших данных. *ZuluCrypt* — графическое приложение шифрования с интуитивно понятным и простым интерфейсом. С помощью *ZuluCrypt* можно создать зашифрованный диск в пределах файла, раздела и даже USB-дисков. *ZuluCrypt* также можно применять для шифрования отдельных файлов с помощью GPG.

Чтобы установить *ZuluCrypt*, откройте <http://mhogomchungu.github.io/zuluCrypt> и прокрутите страницу вниз до раздела двоичных пакетов. Приложение доступно как устанавливаемые файлы Deb-пакета для Debian и Ubuntu. Скачайте пакет для своего дистрибутива и распакуйте его командой `tar xf zuluCrypt*.tar.xz`. Внутри распакованной папки перейдите в папку, соответствующую вашей архитектуре (i386 для старых 32-битных машин и amd64 для новых 64-битных). Обе папки содержат четыре бинарника, устанавливаемых одной командой — `sudo dpkg -i *.deb`. В других дистрибутивах придется установить *ZuluCrypt* вручную. Скачайте архив приложения и следуйте подробным инструкциям в файле `build-instructions` для скачивания зависимостей из репозитория вашего дистрибутива.

Одно из первых дел, которые надо сделать после установки — создать зашифрованные версии всех файлов, которые вы считаете конфиденциальными. Запустите приложение и выберите `zC > Encrypt A File`. В появившемся диалоговом окне нажмите на кнопку рядом с полем `Source` и выберите файл, который требуется зашифровать. *ZuluCrypt* использует эту информацию, чтобы создать файл с тем же именем, и добавит в конце расширение `zC`; или же

сохраните его в другом месте, нажав на значок папки рядом с полем `Destination` и выбрав новое местоположение.

Далее введите пароль для шифрования файла в поле `key`. Пароль должен представлять собой смесь из букв и цифр, чтобы его было сложно угадать. Также помните, что нет средства восстановления пароля, если вы его забудете, и нет возможности расшифровки файла — это информация к размышлению! После подтверждения пароля нажмите кнопку `Create` для шифрования файлов. Время, требуемое на этот процесс, зависит от типа и размера шифруемого файла. По окончании у вас будет зашифрованная версия с расширением `.zC` в указанном вами месте. Раз файл был зашифрован, позаботьтесь удалить его оригинал.

Теперь, чтобы читать этот файл и вносить в него изменения, его надо будет расшифровать. Для этого запустите *ZuluCrypt* и выберите `zC > Decrypt A File`. Укажите зашифрованный файл в поле `Source` и измените расположение разблокированного файла в поле `Destination`. Затем введите пароль, с которым файл шифровался, и нажмите на кнопку `Create`. В указанном месте назначения создастся расшифрованный файл. Чтобы снова заблокировать файл, зашифруйте его, выполнив описанные выше процедуры.

## Склад зашифрованных данных

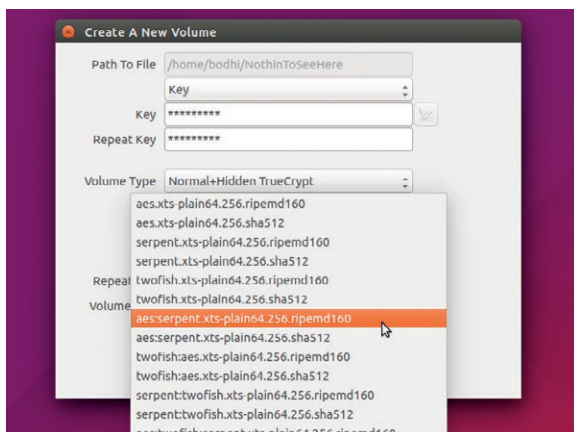
Индивидуальное шифрование файлов работает адекватно, если вы хотите взять под защиту всего лишь пару файлов. Вообще же это трудоемкий процесс и подходит только для файлов, которые не надо читать или изменять регулярно. Если вам необходимо защитить несколько файлов, к которым нужен частый доступ, лучше держать их в зашифрованных областях хранения.

*ZuluCrypt* умеет шифровать блочные устройства, т.е. шифровать все, что на определенном блочном устройстве записано. Блочным устройством может быть весь диск, раздел или даже файл, примонтированный как `loopback`-устройство. При шифровании блочного устройства пользователь создает на этом устройстве файловую систему, и слой шифрования прозрачно шифрует данные перед записью их на фактически нижестоящее блочное устройство. А зашифрованная область хранения просто выглядит как большое скопление случайных данных и не раскрывает даже свою структуру каталогов.

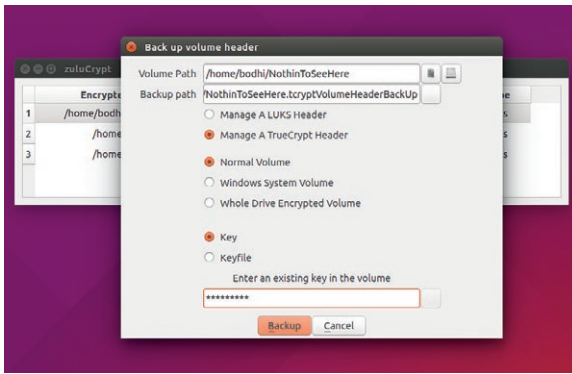
Для создания зашифрованного устройства хранения внутри файла запустите *ZuluCrypt* и выберите `Create > Encrypted Container In A File`. В появившемся окне надо будет ввести имя и полный путь к каталогу, где разместятся ваши конфиденциальные данные. Это называется файл, потому что в зашифрованном виде будет выглядеть как единственный файл. Также укажите размер каталога в зависимости от размера помещаемых в него файлов и доступного пространства на вашем диске.

По нажатию кнопки `Create` в *ZuluCrypt* всплывает еще одно окно. Сначала нужно будет указать пароль для шифрования файла;

» *ZuluCrypt* также поддерживает каскадное шифрование, т.е. шифрование уже зашифрованного сообщения, по тому же или по другому алгоритму.







► **Том без заголовка разблокировать нельзя. Если оригинал будет поврежден, создайте резервную копию, щелкнув правой кнопкой мыши на смонтированном томе и выбрав соответствующую опцию.**

затем выбрать Volume Type. По умолчанию это LUKS (Linux Unified Key Setup) — спецификация шифрования диска, разработанная специально для Linux. Кроме LUKS, *ZuluCrypt* может также создавать и открывать тома *TrueCrypt*, *VeraCrypt* и Plain. Тома Plain — это беззаголовочные зашифрованные тома, а информация о шифровании обеспечивается *ZuluCrypt*. Из-за этого тома Plain зависимы от приложения и не очень переносимы. Тома *TrueCrypt* или *VeraCrypt* — более предпочтительные альтернативы, если зашифрованный том должен быть доступен с компьютеров с Linux, Windows и OS X.

Определившись с типом тома, выберите шифр — алгоритм, который собственно и выполняет шифрование и дешифрацию. Сопутствующим атрибутом шифра является ассоциированный размер ключа. При увеличении размера ключа увеличивается и сложность его нахождения методом перебора, вплоть до практической невозможности взлома шифрования напрямую.

Наиболее популярным алгоритмом шифрования является Advanced Encryption Standard (AES), основанный на шифре Rijndael. Широко используется AES с длиной ключа 256 бит, поскольку предлагает правильный баланс скорости и безопасности. Он является шифром по умолчанию в *ZuluCrypt*. Однако приложение поддерживает большое количество шифров, в том числе алгоритм Twofish и Serpent. Эти два, по мнению Национального института стандартов и технологий США, имеют более высокую безопасность, чем AES, но они медленнее.

Можно спокойно оставить для каждого поля значения по умолчанию, в т. ч. файловую систему по умолчанию для тома (ext4), и нажать на кнопку Create. По завершении процесса вы увидите файл с именем, заданным для зашифрованного контейнера, с неразборчивым содержанием и размером, равным тому, что вы указали ранее.

Прежде чем вы сможете хранить файлы внутри этого зашифрованного тома, вам сначала придется расшифровать и смонтировать его. Выберите Open > PLAIN,LUKS,TrueCrypt Container In A File. По кнопке File во всплывающем окне вы перейдете в зашифрованный файл-контейнер, который только что создали. При желании можно изменить имя точки монтирования тома, или просто введите пароль и нажмите Open. Переключите флажок, если хотите только читать содержимое зашифрованного тома.

После монтирования том появится в вашей файловой системе, аналогично любой другой смонтированной файловой системе. В главном окне *ZuluCrypt* будет также список томов вместе с полным путем точек монтирования. Теперь можно создавать внутри этого смонтированного объема каталоги и файлы, как на любом обычном подключенном устройстве. Закончив, щелкните правой кнопкой мыши на смонтированном томе в интерфейсе *ZuluCrypt* и выберите пункт Close. Том размонтируется и зашифруется — и все, с чем вы останетесь, будет одним зашифрованным файлом

с неразборчивым содержанием. Чтобы раскрыть это содержание, снова смонтируйте файл по процедуре, описанной выше.

Если вам необходимо управление несколькими паролями, *ZuluCrypt* дает возможность создавать случайные файлы ключей, которые затем можно использовать для шифрования файлов и томов. Чтобы сгенерировать файл ключей, выберите Create > Keyfile. Теперь введите имя для файла ключа и путь хранения. Безопасности ради, следует хранить файлы ключей не на том же жестком диске, что и файлы или тома, которые они шифруют. На самом деле, лучше держать их на внешнем диске — это гарантирует, что зашифрованные данные останутся в безопасности, даже если кто-то доберется до вашего диска, содержащего зашифрованные файлы и тома.

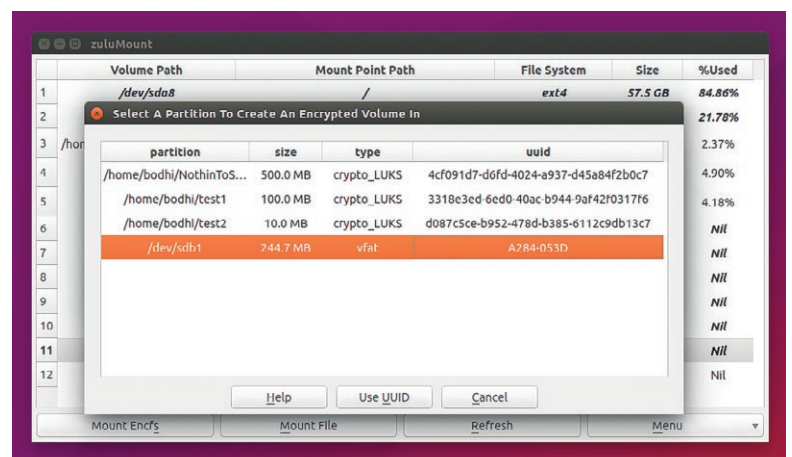
Чтобы использовать вместо пароля файл ключа, выберите опцию файла ключа в выпадающем меню при создании зашифрованного тома или шифровании файла. Выбрав эту опцию, вы также должны указать приложению на файл ключа, который будет затем использоваться для зашифрования ваших данных.

## Шифруем разделы и диски

Если вы хотите шифровать большие объемы данных, лучше всего поместить зашифрованный контейнер в собственный раздел или даже на съемный USB-накопитель. Обратите внимание, что при создании такого контейнера *ZuluCrypt* забирает весь раздел или диск целиком, поэтому не забудьте сделать резервную копию имеющихся данных.

Также убедитесь, что целевой раздел или диск не примонтирован. Список всех смонтированных разделов покажет команда mount. Если раздел, который вы хотите использовать, скажем, /dev/sdb1, смонтирован, надо сперва размонтировать его: `sudo umount /dev/sdb1`. Потом запустите *ZuluCrypt* и выберите Create > Encrypted Container In A Hard Drive. Во всплывающем окне *ZuluCrypt* будет список всех доступных разделов, которые можно использовать для размещения зашифрованного тома. Обратите внимание, что устройства перечислены как по именам, так и по соответствующим UUID. Если вы создаете контейнер на съемном диске, удостоверьтесь, что включена опция Use UUID — это гарантирует, что *ZuluCrypt* всегда правильно идентифицирует устройство. Дважды щелкните на диске/разделе, на котором хотите создать том; теперь вы можете создать зашифрованный том на диске по тем же процедурам, которые мы использовали ранее для создания зашифрованного тома внутри файла.

Сначала *ZuluCrypt* может показаться сложным в использовании, но со временем, по мере ознакомления, это приложение станет вам ближе. Для озабоченных конфиденциальностью более легкого способа сохранить свои данные в безопасности просто нет. **LXF**



► **Инструмент ZuluMount, включенный в ZuluCrypt, умеет расшифровывать все зашифрованные тома, поддерживаемые ZuluCrypt, а также выступает как обычный инструмент монтирования.**

# IP-телефония: Алло, город!

Не у всех счастливицков есть *Asterisk* — Максим Черепанов задумался о тех, кто вынужден обходиться обычным телефоном.



**Наш эксперт**

**Максим Черепанов** — заместитель директора в небольшой коммерческой фирме. По образованию железнодорожник-управленец. Linux занимается с 2008 г. Чем дольше это делает, тем дальше от Windows. За эти годы разучился чистить реестр и забыл слово «антивирус».

**И**так, если вы продолжали следовать за нами по пятам (см. LXF201–LXF203), то вы уже имеете свою собственную АТС с подключенными клиентами в виде софтофонов и выход во внешний мир с помощью USB-модема и сотового оператора. Также мы очень верим в то, что вы своевременно и постоянно делаете бэкапы вашей АТС и равнодушны к вопросам питания (диета для электронных устройств в нашем случае неприемлема, кормить их надо хорошо).

Вернемся к реалиям повседневной жизни: IP-телефония еще победила не везде. Вернее, ее победу тормозят официальные поставщики проводной связи, телефонных сетей общего пользования (ТФОП, привыкайте, это общепринятое сокращение). Иначе каким образом им зарабатывать деньги? Не мешая их бизнесу, будем приспосабливаться к тому, что нас окружает. Если проще — почти в каждом офисе есть проводной телефон с городским номером. Да и дома у многих. Давайте задействуем и эту возможность общаться.

## Шлюз

Как вы понимаете, воткнуть сразу телефонный провод в компьютер не получится. Нужно искать устройство, позволяющее электрические сигналы из медного/алюминиевого провода перевести в дороге нашему сердцу нули и единицы.

Я знаю несколько попыток использовать для этого старые компьютерные модемы 56К, имеющие функцию голоса. Но не знаю из них ни одной успешной. То ли авторы забыли отписаться на форумах, то ли ничего не получилось. Поэтому пойдём проторенной дорогой — приобретем VoIP-шлюз.

Выбор такого оборудования достаточно велик, от именитых Cisco до клонов с непроизносимыми названиями. На самой заре своего углубления в IP-телефонию мне довелось пообщаться



» Рис. 2. Linksys SPA3000 — комплект поставки.

с представителями компаний-интеграторов, и мнение у них одно: для бизнеса допустимы исключительно продукты из верхних строчек списка, отсортированного по цене. Причину называют тоже одну: низкая надежность. Но цена... За такую сумму иногда хочется пойти пешком к абоненту и лично рассказать всё то, что хотел сказать по телефону. Поэтому я попытался найти оптимальное для себя соотношение цена/качество, и выбрал не новую, но популярную модель Linksys SPA3000 (с учетом того, что компания Linksys в 2003 г. была поглощена Cisco, это неплохой выбор). Документации по данной модели достаточно, примеров использования в сети тоже, поэтому я надеялся на нормальную работу устройства. Кроме всего, у модели достаточно грамотный дизайн (рис. 1), ее можно поставить вертикально или горизонтально, либо повесить на стену, и она не испортит вам интерьер.

Тяга к подстиланию соломки нашенстала мне в ухо, и я купил на [ebay.com](http://ebay.com) 2 экземпляра такого шлюза. С учетом того, что получилось выторговать немного на доставке, обошлись они мне в 29\$ за штуку. В комплекте (рис. 2) сам шлюз со съемной подставочкой, блок питания, патч-корд и инструкция на 1 листочке.

Но нас больше интересуют разъемы (рис. 3), потому как нужно понимать, что мы имеем для исполнения своей задачи.

Шлюз имеет обычный разъем для подключения в локальную сеть, разъем питания и 2 линии для подключения телефонного кабеля — FXO и FXS. Давайте кратко познакомимся с ними.

Интерфейс FXS — это порт, который дает возможность подключения абонента к аналоговой телефонной линии. Другими



» Рис. 1. Linksys SPA3000 — общий вид.



► Рис. 3. Linksys SPA3000 — разъемы.

словами, «розетка в стене» выдает сигнал станции и обеспечивает батарейное питание линии, а также напряжение, необходимое для звонка.

Интерфейс FXO — это разъем, в который включается аналоговая телефонная линия. Это разъем на телефонном или факсимильном аппарате или разъем/разъемы на аналоговой мини-АТС. Такой порт имеет индикацию состояния трубка снята/трубка на телефоне (замыкание цепи). Так как порты (разъемы) являются частью устройства, например, телефона или факса, такое устройство часто называют «устройством FXO» или «аналоговым устройством».

Разъемы FXO и FXS всегда парные, то есть обладают «вилкой» и «гнездом». Если коротко, на самом аппарате — FXO, в стенке — FXS, и наш шлюз будет включен в разрыв этого провода.

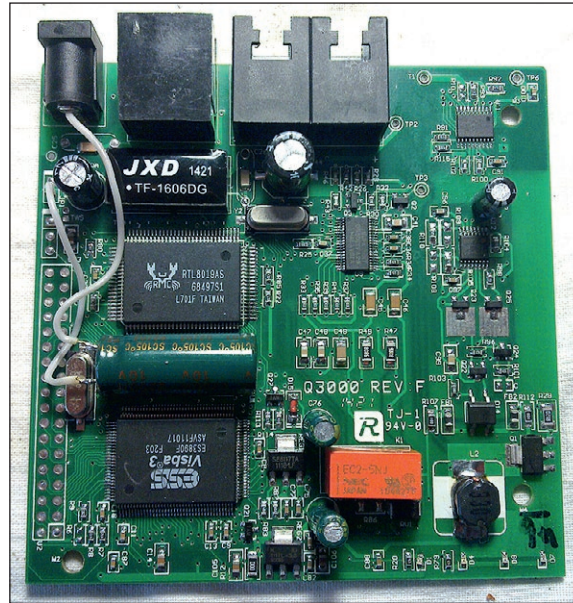
Смекаете? Мы получим не только шлюз: у нас будет как клиент (абонент) подключен аналоговый телефон.

Забегая вперед, скажу, что у данной модели огромное количество возможностей. Мы с вами рассмотрим очень небольшую их часть, но имея немного желания, вы без труда отыщете море информации и сможете превратить шлюз+АТС в мощнейший комбайн для коммуникаций.

## И снова о питании

Но вначале я был сильно раздосадован, были даже поползновения выбросить один из шлюзов на помойку: он упорно не желал работать. Сброс на заводские настройки, заливка всех возможных прошивок не помогала — шлюз упорно перезагружался через непредсказуемые промежутки времени, от 20 секунд до 40 минут. Все настройки сохранялись, но кому нужен такой непредсказуемый выход в ТФОП?

Пришлось брать себя в руки, а в сами руки вкладывать мультиметр, осциллограф и паяльник. Схему найти не удалось, пришлось разбираться прямо по плате, благо разбирается шлюз легко, за 1 минуту. Неисправностей на первый взгляд не было обнаружено. На второй тоже. После двухдневного мониторинга всех возможных точек проблема была локализована — по главной шине питания проскакивала помеха, которая срывала ШИМ-стабилизатор на 3 вольта. Проследовав по шине от стабилизатора, я обнаружил, что внешнее питание, подаваемое через разъем от блока питания, ничем не резервируется — конденсатора даже не предусмотрено на плате! Видимо, разработчики понадеялись на хорошее питание в сети и идеальную работу внешнего блока питания (кстати, его замена не дала ничего). Поэтому пришлось вынуть из неисправной материнской платы конденсатор на 2200 мкФ



► Рис. 4. Linksys SPA3000 — ангрейд своими силами.

6,3 вольта и припаять его параллельно разъему питания (и приклеить на двусторонний скотч в свободном месте) — см. рис. 4.

С этого момента шлюз перестал перезагружаться. Я специально оставлял его на несколько недель для тестирования — всё работает стабильно! Стоит ли говорить, что второй был доработан таким же образом.

Допускаю, что последние два абзаца имеют лишь приблизительное отношение к теме нашей беседы. Но они лишний раз подтверждают, что

- программные проблемы часто по сути аппаратные;
- питание — основа основ;
- идеальных устройств не существует;
- паяльник в руках сисадмина — вовсе не чуждый предмет, особенно в нашей стране.

А теперь, наконец...

## Откроем дверь в город!

Итак, шлюз приведен в чувство и включен, в телефонные разъемы включены провода от телефона и внешней АТС (телефонная розетка). Пока не включен сетевой разъем, телефон включен напрямую в линию, для этого внутри есть реле. Имейте это в виду, т. е. вы сможете пользоваться телефоном даже при неработающем Asterisk'e (что тоже немаловажно — соломка!), проверьте это.

Включаем сетевой кабель, шлюз получает IP-адрес по DHCP, в случае отсутствия DHCP шлюз назначает статический адрес по умолчанию [192.168.0.100](http://192.168.0.100).

Чтобы выяснить адрес шлюза, можно посмотреть на модем/маршрутизаторе в таблице ARP (MAC-адрес шлюза указан на корпусе), или — подключив аналоговый телефон и набрав \*\*\*\* и затем 110#, можно услышать IP-адрес.

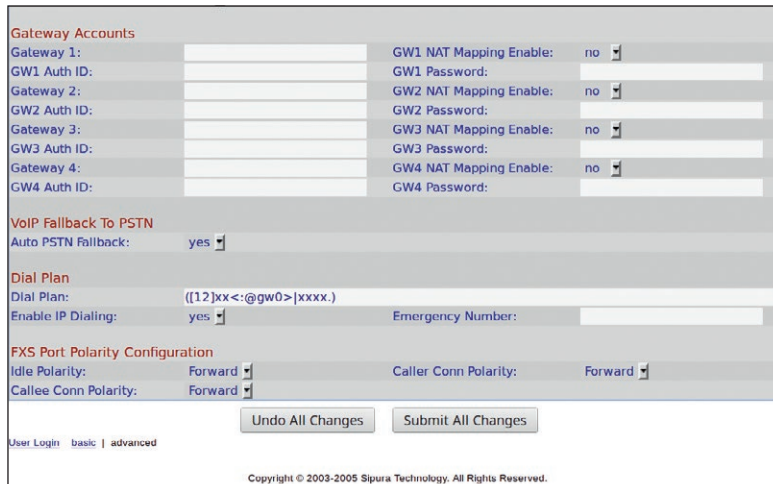
Заходим на шлюз по выясненному IP-адресу, выполняем Admin Login и выбираем режим конфигурирования Advanced.

Для начала пройдите и посмотрите все опции во всех вкладках. Если вы не специалист по IP-телефонии, у вас должно перехватить дух от огромного количества настроек. Не стоит пугаться: мы будем использовать очень небольшое количество, все прочее оставим по умолчанию. Но если вы вооружитесь терпением и проникнетесь идеологией построения шлюза, то сможете ювелирно управлять вашими звонками и предоставить клиентам необыкновенный по качеству сервис.

Пройдемся по общим важным вкладкам:

- Info — тут изменить ничего не получится, это краткая сводка о системе.

»



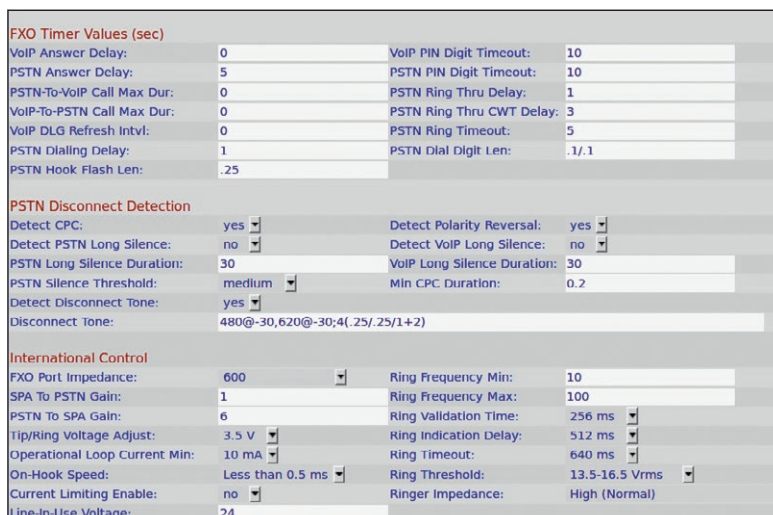
➤ **Рис. 5. Диал-план аналогового телефона.**

- В закладке System можно настроить статический IP-адрес, шлюз, NTP-серверы, пароль администратора, DHCP. Все просто и стандартно.
- Закладки SIP, Provisioning, Regional пока не трогаем. И вообще не трогаем, если все будет работать нормально.
- Line 1 — вот тут начало того, ради чего все затевалось.

Будем настраивать подключение аналогового телефона как SIP-клиента к Asterisk. Что-то я буду давать в текстовом виде, что-то — как скриншот, они на шлюзе длинные, и давать их полностью — расточительство. Итак, Line Enable ставим в yes, SIP Port назначаем 5060 (как мы делали в Asterisk'e), для Proxy задаем адрес Asterisk'a (в примере 192.168.0.100), для Subscriber Information — заполняем поля на одного из наших пользователей, заданного в меню Applications > Extentions нашей АТС Free-PBX, это аналог полей пользователя в софтовых клиентах. Я отключил софтового клиента 1004 и прописал его параметры в шлюзе. Дальше я должен признаться, что аналоговая линия у меня подключена к офисной АТС, имеющей трехзначную нумерацию. Помня про это, я сделал в Asterisk+FreePBX четырехзначную. Т.е. у меня один аналоговый телефон будет иметь 2 номера: один из 3-х цифр — обычная офисная станция, из четырех (1004) — то, что мы только что написали, это Asterisk. А как же тогда звонить? Вот в этом и заключается вся прелесть: шлюз сам разберется, куда и как ему звонить!

Включаем Dial plan и вписываем в него приведенную на скриншоте рис. 5 строку. Она указывает шлюзу, что все трехзначные номера, начинающиеся с 1 или 2, надо направлять на аналоговую линию (gw0); четырех- и более значные номера, набранные на аналоговом телефоне, направлять на Asterisk.

➤ **Рис. 6. Шлюз — временные задержки.**



Нажимаем кнопку Submit All Changes, шлюз перегружается, и в командной строке Asterisk (которую вы уже включили на термине, без напоминаний) видим сообщение:

```
[2016-01-12 10:26:13] NOTICE[31930]: chan_sip.c:23739 handle_response_peerpoke: Peer '1004' is now Reachable. (10ms / 2000ms)
```

Мы готовы звонить и принимать звонки с помощью аналогового телефона.

Следующим шагом необходимо подключить городскую линию к Asterisk'у. Здесь надо быть особо внимательным. Начнем с сервера, открываем панель FreePBX, добавляем новую линию: Trunk > Add SIP Trunk. Заполняем поля Trunk Name (я вписал Linksys), Maximum Channels: 1 и PEER Details.

```
disallow=all
allow=ulaw
canreinvite=no
context=from-trunk
dtmfmode=rfc2833
host=192.168.0.115
incominglimit=1
port=5061
qualify=yes
type=peer
```

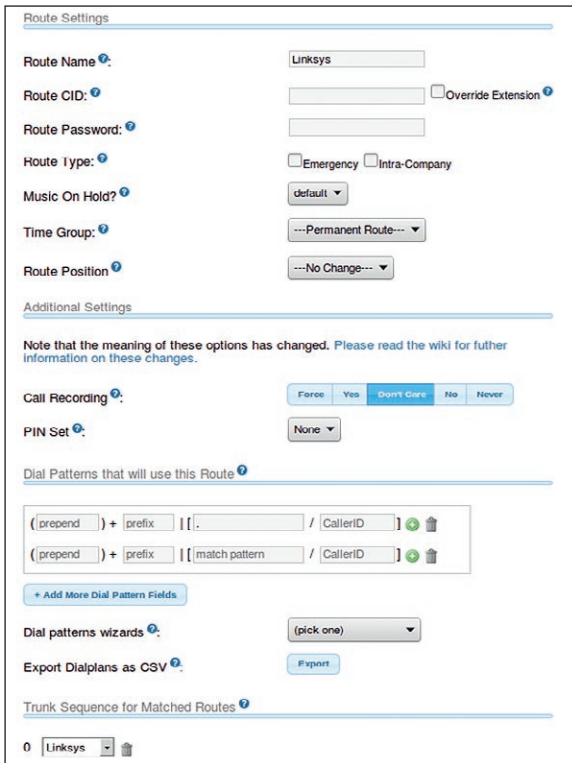
192.168.0.115 — это адрес моего шлюза, у вас будет свой. Обратите внимание на порт! Он выбран другим, нежели подключение SIP-клиентов (в Extentions).

Далее переходим на шлюз, на закладку PSTN Line (ТФОП), надо обязательно разрешить анонимные звонки:

```
Make Call Without Reg = Yes
Ans Call Without Reg = Yes
Register = No
```

Остальные параметры (на некоторых из них я остановлюсь ниже, более подробно):

```
Network Jitter Level = low
Jitter Buffer Adjustment = disable
SIP Port = 5061
Proxy = 192.168.0.100
Register Expires = 300
Preferred Codec = G711u
DTMF Process INFO = Yes
TMF Process AVT = Yes
DTMF Tx Method = Auto
Echo Canc Enable = yes
Echo Canc Adapt Enable = yes
Echo Supp Enable = yes
Dial Plan 2 = (S0<:12345678@192.168.0.100)
VoIP-To-PSTN Gateway Enable = yes
VoIP Caller Auth Method = None
VoIP PIN Max Retry = 3 ;
One Stage Dialing = Yes ;
Line 1 VoIP Caller DP = none
VoIP Caller Default DP = none
Line 1 Fallback DP = none
PSTN-To-VoIP Gateway Enable = Yes
PSTN Caller Auth Method = none
PSTN Ring Thru Line 1 = no
PSTN Pin Max Retry = 3
PSTN CID for VoIP CID = Yes
PSTN CID Number Prefix = (Leave Blank)
PSTN Caller Default DP = 2
Off Hook While Calling VoIP = Yes
Line 1 Signal Hook Flash To PSTN = Disabled
Detect CPC = yes
Detect Polarity Reversal = yes
Detect PSTN Long Silence = no
Detect VoIP Long Silence = no
```



► Рис. 7. Исходящий маршрут на шлюз.

PA To PSTN Gain = 1

PSTN To SPA Gain = 6

Теперь прокомментирую выделенное.

» SIP Port = 5061; порт, должен совпадать с указанным в PEER Details во *FreePBX*.

» Proxy = 192.168.0.100; это адрес Asterisk.

» Dial Plan 2 = (\$0<:12345678@192.168.0.100);12345678 — DID входящего маршрута, по нему будем идентифицировать шлюз. Теоретически этого можно не делать, но если вы захотите направлять в разные места звонки с разных входящих маршрутов (с сотового USB-модема, с ТФОП или с транка SIP-провайдера), без этого не обойтись.

» PSTN Caller Default DP = 2; диал-план, который направляет звонок со шлюза на *Asterisk*, должен совпадать с предыдущей строкой.

На рис. 6 показан скриншот с временными задержками, у меня стабильно заработало с такими настройками.

Возможно, у вас будут свои, тут все дело в реакции аналоговой линии — нужен ли префикс (у меня выход в город — 9, я его набираю вручную, чтобы можно было набирать внутренние номера), и насколько быстро линия дает ответ после набора этого префикса. Это все возможно запрограммировать в *Asterisk+FreePBX*; оставляю этот момент творчества для вас.

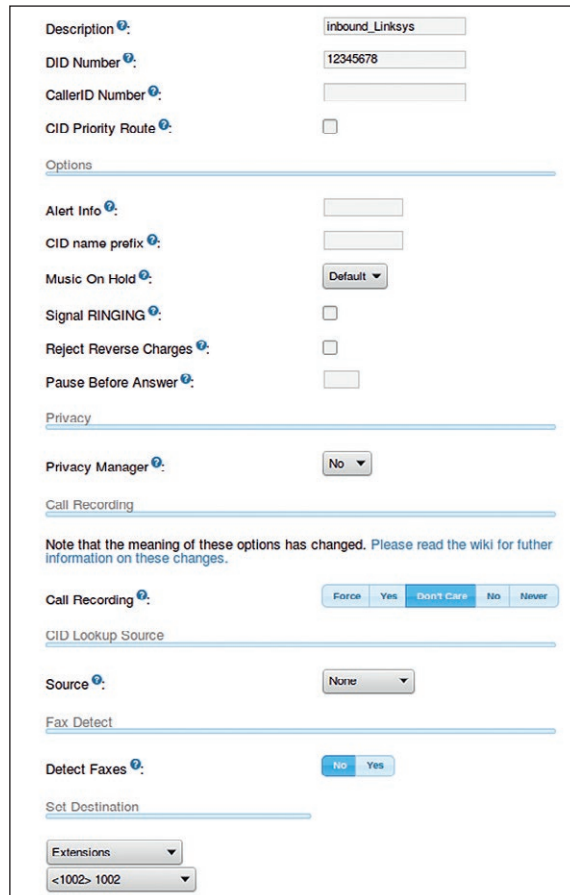
Возвращаемся во *FreePBX*: транк мы прописали, осталось задать маршруты (не забывая сохранять и применять все, что вы сотворили — помните просьбу о красной кнопке?).

Организовываем исходящий маршрут Outbound Routes > ADD Route (рис. 7).

И в секции Trunk Sequence for Matched Routes указываем созданный нами транк.

Организовываем входящий маршрутизатор — Inbound Routes > ADD Route (рис. 8).

Обратите внимание, в поле DID указывается тот набор цифр (назовем его паролем), который вы задали в настройках шлюза в поле Dial Plan 2. После успешной настройки попробуйте менять его в АТС и в шлюзе. Вы убедитесь, что несовпадение приведет к невозможности совершать звонки.

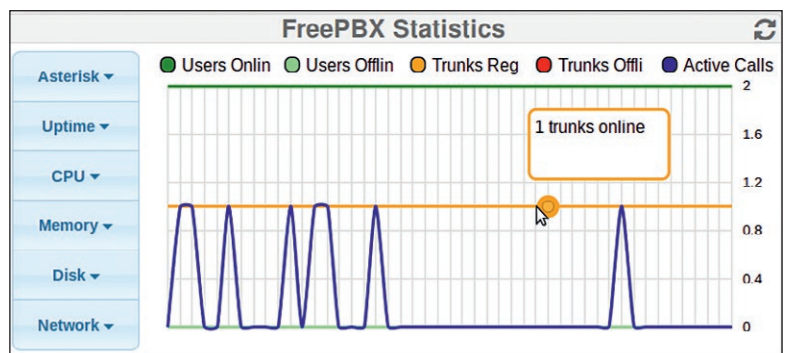


► Рис. 8. Входящий маршрут со шлюза.

А внизу указываем, куда нам направлять звонки, пришедшие по этому маршруту; я направил его просто на абонента 1002.

В последней фразе скрывается вся мощь, которая становится доступной после всех произведенных манипуляций. Вы можете отправлять пользователя на IVR-меню, давать ему прослушать информацию, соединять с группой или ставить в очередь при занятости нужного абонента... Если честно, я завидую тем, кто еще не пробовал все эти «вкусности» IP-АТС, потому что вас ждет буря эмоций и ощущение необыкновенного могущества от полного контроля за системой. В подтверждение моих слов перейдите во *FreePBX* в меню Reports > System status; кроме всего прочего, вы сможете в графическом виде наблюдать за количеством активных пользователей, транков и совершенных через АТС звонков (рис. 9). Пока я писал все это для вас (а писалось все с живой системы), я сделал несколько звонков и выделил мышью активный транк.

Пожалуй, на сегодня все. Желаю приятного содержания ваших разговоров через *Asterisk!* **LXF**



► Рис. 9. Графики всегда наглядны!



# MySQL Fabric:

# Введение

Лада Шерышова устанавливает, настраивает и запускает для вас программу управления фермами серверов.



**Наш эксперт**

**Лада Шерышова** долгие годы работала на коммерческие корпорации, создавая промышленные высоконадежные информационные системы. Но пришло время сбросить оковы и применить свои знания и опыт на работу со свободным ПО.

**П**ервое наше знакомство с *MySQL Fabric* состоялось в прошлом месяце (стр. 14 **LXF204**). Теперь мы изучим этот программный продукт подробнее.

Сначала немного теории. *MySQL Fabric* имеет две области управления серверами, которые помогают администраторам баз данных: высокая доступность (ВД) и шардинг данных.

Для обеспечения высокой доступности *MySQL Fabric* позволяет создавать группы MySQL-серверов, которые рассматриваются как отдельные логические единицы. Каждая такая логическая единица состоит из главного [primary] сервера, которому доступны операции записи и чтения [Read-Write], и подчиненных [secondary] серверов, которым доступны только операции чтения [Read-Only]. Они также используются для масштабирования и могут принять роль главного сервера в случае его отказа. Наборы серверов в *MySQL Fabric* называются высокодоступными (ВД) группами. Информация обо всех серверах и группах находится в ведении отдельного экземпляра MySQL, который не является членом ВД-группы, а функционирует как самостоятельный узел. Этот экземпляр называется узлом *MySQL Fabric*.

*MySQL Fabric* может быть сконфигурирован для автоматического мониторинга статуса серверов в группе. Если текущий главный сервер в группе отказывает, *MySQL Fabric* автоматически назначает новый главный сервер из имеющихся подчиненных.

Для обеспечения репликации данных *Fabric* управляет набором MySQL-серверов, которые имеют т.н. глобальные идентификаторы транзакций — Global Transaction Identifiers (GTIDs). GTID — это уникальный идентификатор, который создается и привязывается к каждой транзакции, фиксируемой на главном сервере. Этот идентификатор является уникальным не только для сервера, который породил транзакцию, но и для всех серверов, входящих в группу репликации. Таким образом, GTID'ы позволяют осуществлять проверки и поддерживать непротиворечивость между серверами.

Для работы с шардингом данных *MySQL Fabric* использует специальные коннекторы. Можно управлять шардами и без коннекторов, используя командную строку утилиты *mysqlfabric*. Однако в этом случае приложение придется дополнять логикой, задающую правила выбора шардов при запросах. При использовании коннекторов между приложением и *MySQL Fabric* устанавливается соединение либо на основе протокола XML-RPC, либо протокола MySQL Client/Server. Коннекторы кэшируют маршрутную информацию, полученную от *Fabric*, и направляют запросы на запись и чтение от клиентского приложения в соответствующую шардированную таблицу. *MySQL Fabric* позволяет выполнять различные операции над шардами: создание, удаление, перемещение и разделение.

*MySQL Fabric* написан на языке Python и включает специальную библиотеку, которая реализует все предоставляемые продуктом функциональные возможности. Для взаимодействия с *Fabric*

имеется специальная утилита *mysqlfabric*, которая предоставляет набор команд, используемых для создания и управления группами, определения и работы с шардингом данных, и многого другого.

Итак, разбираемся с *MySQL Fabric*.

## Основные концепции

Прежде чем приступить к изучению *MySQL Fabric*, познакомимся с концепциями, лежащими в его основе.

Высокодоступная группа (ВД-группа) или просто группа — пул серверов, управляемых *MySQL Fabric*. Сервера, входящие в ВД-группу, являются экземплярами сервера MySQL (*mysqld*). Каждый сервер может быть членом только одной группы и быть привязаным только к одному главному серверу.

Идентификатор группы — наименование группы, представляющее собой регулярное выражение [a-zA-Z0-9\_-]+ (например, group-1, my\_group и т.д.). Каждая ВД-группа должна иметь свой идентификатор.

Глобальная группа — специальная группа, участвующая в шардинге. Она хранит изменения, которые должны распространяться на все шарды.

Узел *MySQL Fabric* — запущенный экземпляр *MySQL Fabric*. Это программа на языке Python, которая, помимо прочего, служит XML-RPC-сервером, используемым коннекторами и утилитой *mysqlfabric* для работы в командной строке. На узле *MySQL Fabric* также д.б. запущен экземпляр *mysqld*. Этот экземпляр предназначен для хранения информации, необходимой для управления серверами, и называется хранилищем состояния *MySQL Fabric*.

Шард — горизонтальный сегмент данных в таблице. Сегментирование (шардирование) происходит на уровне *MySQL Fabric*. При этом экземпляр *mysqld* ничего не знает о сегментах, и для него шард — это просто таблица.

PRIMARY — сервер, который назначен на роль главного, и принимает транзакции как чтения, так и записи. В каждой ВД-группе может быть только один главный сервер.

SECONDARY — подчиненный сервер, ему доступны только операции чтения. Он может быть назначен на роль главного в случае отказа этого последнего, а также применяться для масштабирования операций чтения.

## Требования к установке

*MySQL Fabric* работает с MySQL-серверами не ниже версии 5.6.10. Утилита *mysqlfabric* требует наличия Python 2 (2.6 или выше). Для работы *Fabric* с приложениями необходимы коннекторы: Connector/Python 1.2.1 (и выше) или Connector/J 5.1.27 (и выше). Для хранилища состояния *MySQL Fabric* требуется MySQL-сервер версии 5.6 или выше. На данном уроке мы не будем рассматривать процесс установки и запуска *mysqld*, предполагая, что сервер MySQL уже установлен и запущен на узле *MySQL Fabric*.

## Шаг за шагом: Установим и запустим сервер MySQL Fabric

### 1 Скачиваем и устанавливаем

В настоящее время для свободного (некоммерческого) использования доступна версия *MySQL Fabric 1.5.6*, которая устанавливается из набора утилит *MySQL Utilities* версии 1.5.6. Скачиваем ее по ссылке <http://dev.mysql.com/downloads/utilities/>, выбрав соответствующую платформу (на данном уроке мы, естественно, будем использовать ОС семейства Linux).

Распаковываем дистрибутив:

```
>> gunzip mysql-utilities-1.5.6.tar.gz
>> tar -xvf mysql-utilities-1.5.6.tar
>> cd mysql-utilities-1.5.6
```

Устанавливаем Connector/Python:

```
mysql-utilities-1.5.6>> python setup.py build
mysql-utilities-1.5.6>> sudo python setup.py install
```

Connector/Python можно также установить, скачав его по ссылке <http://dev.mysql.com/downloads/connector/python/>.

Проверяем, что *MySQL Fabric* установился:

```
>> mysqlfabric --version
mysqlfabric 1.5.6
```

### 2 Создаем пользователя MySQL Fabric

Подключаемся к серверу (и при необходимости указываем порт и сокет-файл):

```
>> /bin/mysql -u root --port=3311 --host=localhost --socket=/tmp/mysql1.sock --protocol=tcp
```

Создаем пользователя (например, с именем *fabric* и таким же паролем):

```
mysql> CREATE USER 'fabric'@'localhost' IDENTIFIED BY 'fabric';
```

Назначаем ему привилегии суперпользователя :

```
mysql> GRANT ALL ON *.* TO 'fabric'@'localhost';
```

И перезагружаем таблицу привилегий, чтобы изменения вступили в силу:

```
mysql> FLUSH PRIVILEGES;
```

### 3 Создаем пользователей на MySQL-серверах

На каждом сервере MySQL, управляемом *Fabric*, должны быть созданы 3 типа пользователей: *server user* — для доступа к серверу; *backup user* — пользователь, отвечающий за выполнение операций резервирования; *restore user* — пользователь, отвечающий за операции восстановления.

Запускаем *mysql*. Отключаем запись в двоичный лог (это делается для того, чтобы при репликации избежать повторного создания этих пользователей):

```
mysql> SET SESSION SQL_LOG_BIN = 0;
```

Создаем пользователей вышеперечисленных типов: *server user* (например, *group*), *backup user* (например, *fabric\_backup*) и *restore user* (например, *fabric\_restore*):

```
mysql> CREATE USER 'group'@'localhost' IDENTIFIED BY 'group';
```

```
mysql> CREATE USER 'fabric_backup'@'localhost' IDENTIFIED BY 'fabric';
```

```
mysql> CREATE USER 'fabric_restore'@'localhost' IDENTIFIED BY 'fabric';
```

Назначаем им привилегии суперпользователей:

```
mysql> GRANT ALL ON *.* TO 'group'@'localhost';
```

```
mysql> GRANT ALL ON *.* TO 'fabric_backup'@'localhost';
```

```
mysql> GRANT ALL ON *.* TO 'fabric_restore'@'localhost';
```

Перезагружаем таблицу привилегий:

```
mysql> FLUSH PRIVILEGES;
```

И разрешаем запись в двоичный журнал:

```
mysql> SET SESSION SQL_LOG_BIN = 1;
```

### 4 Настраиваем конфигурационный файл Fabric

Конфигурационный файл содержит всю необходимую информацию для запуска узла *MySQL Fabric*. Кроме того, он служит и как конфигурационный файл для внешних утилит. Расположение конфигурационного файла *Fabric* варьируется в зависимости от установленной операционной системы и от того, как устанавливался *MySQL Fabric*. В случае Linux-систем этот файл находится в */etc/mysql/fabric.cfg*. Для Windows — *UTILITIES\_INSTALLDIR/etc/mysql/fabric.cfg*. Конфигурационный файл необходимо дополнить информацией обо всех созданных пользователях. Для этого открываем файл *fabric.cfg* и в раздел *[storage]* вносим следующую информацию:

» **address** Адрес, по которому будет располагаться хранилище состояния *Fabric* в формате *host:port* (например, *localhost:3311*).

» **user** Имя пользователя, которое используется при подключении к хранилищу состояния *Fabric* (например, *fabric*).

» **password** Его пароль.

» **database** Наименование базы данных, которая будет хранить информацию о состоянии *Fabric* (например, *fabric*).

Дополнительно можно настроить:

» **auth\_plugin** — плагин аутентификации, используемый при подключении к резервному хранилищу. Значение этого параметра передается коннектору при подключении к хранилищу.

» **connection\_timeout** — таймаут подключения к хранилищу *Fabric* (в секундах). Этот параметр также передается коннектору при подключении к хранилищу. Это максимальное время, в течение которого *MySQL Fabric* ожидает доступ к хранилищу.

» **connection\_attempts** — количество попыток подключений к хранилищу *Fabric*.

» **connection\_delay** — задержка между попытками подключений к хранилищу *Fabric* (в секундах).

В раздел *[servers]* надо внести информацию о созданных пользователях:

» **user** — имя пользователя для подключения к серверу MySQL, управляемому *Fabric* (в нашем примере — *group*).

»

```
*fabric.cfg x
[storage]
address = localhost:3311
user = fabric
password = fabric
database = fabric
auth_plugin = mysql_native_password
connection_timeout = 6
connection_attempts = 6
connection_delay = 1
```

» Рис. 1. Настройка конфигурационного файла: раздел *[storage]*.

```
*fabric.cfg x
[servers]
user = group
password = group
backup_user = fabric_backup
backup_password = fabric
restore_user = fabric_restore
restore_password = fabric
unreachable_timeout = 5
```

» Рис. 2. Настройка конфигурационного файла: раздел *[servers]*.

► Рис. 3. Настройка конфигурационного файла: раздел [protocol.xmlrpc].

```
*fabric.cfg x
[protocol.xmlrpc]
address = localhost:32274
threads = 5
user = client
password = client
disable_authentication = no
realm = MySQL Fabric
ssl_ca =
ssl_cert =
ssl_key =
```

- » password — его пароль (group).
- » backup\_user — имя пользователя, который будет использоваться для операций резервирования данных, например, утилитой *mysqldump* (в нашем примере — *fabric\_backup*).
- » backup\_password — его пароль (fabric).
- » restore\_password — имя пользователя, который будет отвечать за операции восстановления (в нашем примере — *fabric\_restore*).
- » restore\_password — его пароль (fabric).

В раздел [protocol.xmlrpc] внесем информацию о пользователе, который будет использоваться утилитой *mysqlfabric* для соединения с узлом *MySQL Fabric*. В разделе [protocol.xmlrpc] конфигурационного файла задаем его имя и пароль (например, *client*).

- » user — имя пользователя, которое используется при подключении к XML-RPC-серверу (в нашем примере — *client*).
- » password — его пароль (client).

Немного о других параметрах настройки XML-RPC сервера:

- » address — хост и порт XML-RPC сервера. Хост используется только клиентским приложением при подключении к узлу *MySQL Fabric*. Порт используется сервером при запуске протокола и клиентом при чтении информации и подключении к XML-RPC-серверу. Обычно номер порта — 32274, а имя хоста — localhost.
- » threads — количество паралельных запросов, которое принимает XML-RPC-сервер для обработки.
- » disable\_authentication — определяет, отключать ли аутентификацию или нет. Отключение бывает полезно при тестировании в закрытой среде, но *не* рекомендуется в промышленной эксплуатации. Значения параметра — *yes* или *no*, регистр роли не играет.

Для безопасного соединения клиентского приложения с XML-RPC сервером используйте следующие настройки:

- » realm — область, идентифицирующая XML-RPC-сервер при проверке подлинности.

► Рис. 4. Установка схемы данных *Fabric*.

```
admin@linux-63hp.site:/usr/local/mysql x
linux-63hp:/usr/local/mysql # mysqlfabric manage setup
[INFO] 1455117946.956825 - MainThread - Initializing persister; user (fabric), server (localhost:3312), database (Fabric).
Finishing initial setup
=====
Password for admin user is not yet set.
Password for admin/xmlrpc:
Repeat Password:
Password set.
Password set.
linux-63hp:/usr/local/mysql #
```

```
admin@linux-63hp.site:/usr/local/mysql x
linux-63hp:/usr/local/mysql # mysqlfabric manage start
[INFO] 1455117268.915536 - MainThread - Initializing persister; user (fabric), server (localhost:32274), database (fabric).
[WARNING] 1455117268.984234 - MainThread - Provider error: No module named novaclient.
[INFO] 1455117268.987944 - MainThread - Loading Services.
[INFO] 1455117269.019591 - MainThread - MySQL-RPC protocol server started, listening on localhost:32274
[INFO] 1455117269.048080 - MainThread - Fabric node version (1,5,0) started.
[INFO] 1455117269.132224 - MainThread - Starting Executor.
[INFO] 1455117269.132403 - MainThread - Setting 5 executor(s).
[INFO] 1455117269.133056 - Executor-0 - Started.
[INFO] 1455117269.135108 - Executor-1 - Started.
[INFO] 1455117269.141375 - Executor-2 - Started.
[INFO] 1455117269.151094 - Executor 3 Started.
[INFO] 1455117269.151191 - MainThread - Executor started.
[INFO] 1455117269.177209 - Executor-4 - Started.
[INFO] 1455117269.190084 - MainThread - Starting failure detector.
[INFO] 1455117269.206917 - XML-RPC-Server - XML-RPC protocol server ('127.0.0.1', 32274) started.
[INFO] 1455117269.212753 - XML-RPC-Server - Setting 5 XML-RPC session(s).
[INFO] 1455117269.214875 - XML-RPC-Session-0 - Started XML-RPC-Session.
[INFO] 1455117269.220712 - XML-RPC-Session-1 - Started XML-RPC-Session.
[INFO] 1455117269.228423 - XML-RPC-Session-2 - Started XML-RPC-Session.
[INFO] 1455117269.238448 - XML-RPC-Session-3 - Started XML-RPC-Session.
[INFO] 1455117269.246384 - XML-RPC-Session-4 - Started XML-RPC-Session.
```

► Рис. 5. Запуск узла *MySQL Fabric*.

- » ssl\_ca — путь к файлу, содержащему список доверенных центров сертификации SSL (CAs).
- » ssl\_cert — имя файла сертификата SSL для установления безопасного соединения.
- » ssl\_key — имя файла SSL ключей для установления безопасного соединения.

Выполнив изменения, сохраняем их и закрываем файл.

## 5 Устанавливаем схему данных *Fabric*

Схема данных *MySQL Fabric* создается с помощью команды

```
>> mysqlfabric manage setup
```

Система предложит ввести пароль для пользователя *admin*:

```
Password for admin user/xmlrpc:
```

Вводим свой пароль (например, такой же, как и имя пользователя — *admin*) и повторяем его в строке Repeat Password:

## 6 Запускаем узел *MySQL Fabric*

Для запуска узла *MySQL Fabric* выполняем команду

```
>> mysqlfabric manage start
```

Чтобы перевести узел *MySQL Fabric* в фоновый режим, используется команда

```
>> mysqlfabric manage start --daemonize
```

В этом режиме логи не выводятся на экран терминала, а пишутся в файл, указанный в файле *fabric.cfg* в разделе [logging]:

```
[logging]
```

```
level = INFO
```

```
url = file:///var/log/fabric.log
```

Параметр *level* устанавливает уровень журналирования. Допустимые значения — *CRITICAL* (критический), *ERROR* (ошибка), *WARNING* (предупреждение), *INFO* (информация) и *DEBUG* (отладка). Значение по умолчанию — *INFO*. Параметр *url* — это URL-адрес, куда пишутся логи. Поддерживаемые протоколы — *file* и *syslog*. Протокол *file* создает файловый обработчик, в то время как протокол *syslog* регистрирует сообщения, используя системный регистратор *syslogd*. Файловый обработчик принимает либо относительный, либо абсолютный путь. Обработчик *syslog* принимает либо путь (например, *syslog:///dev/log*), либо имя хоста, и в качестве необязательного параметра — порт (например, *syslog://localhost:555* или *syslog://my.example.com*). Если порт не задан, то по умолчанию ставится 541.

## 7 Создаем пользователя для соединения с XML-RPC-сервером

Для соединения с XML-RPC-сервером создадим пользователя с именем *client* (его мы уже внесли в конфигурационный файл). Этот пользователь создается под пользователем *Fabric* *admin*. Для этого выполним команду

```
>> mysqlfabric user add client --user=admin
```

Укажем протокол *xmlrpc* в приглашении *Protocol (default xmlrpc)*, пароль для пользователя *client* (например, *client*) в строке *Password*: и повторим его в *Repeat Password*. Далее выбираем из списка роль для нового пользователя. В строке *Enter comma separated list of role IDs or name*; введем 1. В случае успешного создания пользователя выведется сообщение "Fabric user added".

Вот и всё! Начинаем работать с *MySQL Fabric*!

## Хранилище состояния *MySQL Fabric*

А что находится в нем? Заглянем в хранилище состояния *Fabric*, выполнив команду

```
>> mysqlshow -u fabric -p fabric fabric
```

В результате мы получим на экране список таблиц, из которых состоит схема базы данных *Fabric*:

- » checkpoints Хранит информацию о выполнении процедур; используется для безопасного возобновления выполнения процедур в случае отказов и восстановлений.



- » **error\_log** Хранит информацию о серверных ошибках.
- » **group\_replication** Определяет репликацию между глобальными группами и группами в шардах; используется, в основном, для разделения шардов, их перемещения и глобальных обновлений.
- » **groups** Содержит информацию о ВД-группах.
- » **permissions** Содержит информацию о правах доступа к различным подсистемам *Fabric*.
- » **role\_permissions** Связывает роли и разрешения для ролей.
- » **roles** Содержит информацию о пользовательских ролях.
- » **servers** Хранит информацию о серверах, управляемых *Fabric*.
- » **shard\_maps** Содержит правила шардирования данных.
- » **shard\_ranges** Содержит индексы шардирования, которые используются при сопоставлении ключа шардирования с шардом.
- » **shard\_tables** Хранит список шардированных таблиц с привязкой к правилам шардирования для каждой таблицы.
- » **shards** Содержит информацию о группах, к которым принадлежат шарды.
- » **user\_roles** Содержит информацию о ролях пользователей.
- » **users** Хранит информацию о пользователях, которые имеют разрешение на доступ к различным подсистемам.

Хранилище состояния — очень важный элемент *MySQL Fabric*. Поэтому необходимо периодически выполнять резервное копирование, используя утилиты *mysqldump* или *mysqldbexport*, входящие в состав MySQL Utilities.

## Заглянем в конфигурационный файл

Конфигурационный файл *Fabric* состоит из разделов, в каждом из которых содержится набор переменных, отвечающих за настройку узла *MySQL Fabric*. Немного прогуляемся по разделам файла **fabric.cfg**:

- » **[default]** Содержит информацию о путях установки *MySQL Fabric*. Этот раздел создается в процессе установки утилиты и обычно не модифицируется.
- » **[storage]** Содержит параметры подключения к хранилищу состояния *MySQL Fabric*.
- » **[servers]** Содержит информацию, которую *MySQL Fabric* использует при подключении к управляемым серверам фермы.
- » **[logging]** Настройки журналирования при запуске узла *MySQL Fabric* в фоновом режиме.
- » **[protocol.xmlrpc]** В данном разделе содержится информация о том, как клиентское приложение подключается к *MySQL Fabric*, и параметры настройки XML-RPC-сервера.
- » **[protocol.mysql]** Раздел содержит информацию о том, как клиентское приложение подключается к узлу *MySQL Fabric*, используя протокол MySQL Client/Server.
- » **[executor]** Содержит параметры конфигурации исполнителя. Исполнитель выполняет процедуры в последовательном порядке и гарантирует, чтобы запросы не конфликтовали. Полученные запросы сопоставляются с процедурами, которые могут быть выполнены непосредственно или по расписанию через исполнителя. Процедуры, назначенные через исполнителя, обрабатываются в контексте потоков, порожденных исполнителем. Как правило, операции чтения немедленно выполняются в сессии XML-RPC, а операции записи ставятся в расписание и выполняются исполнителем.
- » **[sharding]** Для выполнения операций перемещения и разделения шардов *MySQL Fabric* использует клиентские программы *mysqldump* и *mysql*. Раздел содержит пути их расположения на сервере *MySQL Fabric*.
- » **[statistics]** Содержит настройки параметров мониторинга жизнеспособности серверов и журнала событий.
- » **[failure\_tracking]** Конфигурирует параметры для управления отказами в системе.
- » **[connector]** Содержит параметры настройки времени жизни (TTL) КЭШа, передаваемые от *MySQL Fabric* коннекторам.

```
admin@linux-63hp.site:/usr/local/mysql
linux-63hp:/usr/local/mysql # mysqlfabric user add client --user=admin
Password for admin:
Add a new Fabric user
-----
Username: client
Protocol (default xmlrpc): xmlrpc
Password:
Repeat Password:

Select role(s) for new user
ID  Role Name  Description and Permissions
--  -
1   superadmin  Role for Administrative users
      + Full access to all core Fabric functionality
2   useradmin   Role for users dealing with user administration
      + User administration
      + Role administration
3   connector   Role for MySQL Connectors
      + Access to dump commands
      + Reporting to Fabric

Enter comma separated list of role IDs or names: 1
Fabric user added.
linux-63hp:/usr/local/mysql #
```

» Рис. 6. Создание пользователя *MySQL Fabric*.

С разделами **[servers]**, **[storage]**, **[protocol.xmlrpc]** и **[logging]** мы уже познакомились при установке *MySQL Fabric*. Более подробную информацию об остальных параметрах конфигурации вы можете найти в разделе документации <http://dev.mysql.com/doc/mysql-utilities/1.5/en/fabric-cfgref.html>.

## Конфигурация на лету

В *MySQL Fabric* есть дополнительная опция **--param**, которая позволяет переопределять параметры конфигурации среды во время выполнения команд: **--param=section.option=value**. Например:

```
>> mysqlfabric manage setup --param=storage.user=fabric_store
--param=storage.password=fabric
>> mysqlfabric --param=storage.user=fabric_store
--param=storage.password=fabric manage setup
>> mysqlfabric manage setup --param=storage.
address=localhost:3315 \ --param=storage.user=root
--param=protocol.xmlrpc.password=admin
```

## Без подсказок — никуда: HELP!

Система помощи *MySQL Fabric* — это набор простых команд, предоставляющих всю необходимую информацию для работы с утилитой *mysqlfabric*:

- » **mysqlfabric help** Отображает синтаксическую информацию и основные команды помощи.
- » **mysqlfabric help commands** Отображает список всех доступных команд и их описание.
- » **mysqlfabric help groups** Отображает список доступных групп команд (не путать с ВД-группами!). Команды *Fabric* организованы в следующие группы: **dump**, **event**, **group**, **manage**, **provider**, **role**, **server**, **sharding**, **snapshot**, **statistics**, **threat** и **user**.
- » **mysqlfabric help [group] [command]** Детализированная информация о команде.

Теперь мы во всеоружии! **LXF**

## Справочник администратора

Для управления узлом *MySQL Fabric* предоставляются следующие команды:

- » **teardown** Удаляет систему хранения *Fabric*.
- » **stop** Останавливает сервер *MySQL Fabric*.
- » **setup** Устанавливает систему хранения *Fabric*.
- » **ping** Проверяет, запущен ли сервер *Fabric*.
- » **start** Запускает сервер *Fabric*.

» **logging\_level** Устанавливает уровень журналирования (используется, в основном, для отладки).

Синтаксис выполнения команды:  

```
>> mysqlfabric manage [command]
[--options]
```

Более детальную информацию по командам управления можно получить, используя систему помощи *Fabric*:

```
>> mysqlfabric help manage [command]
```

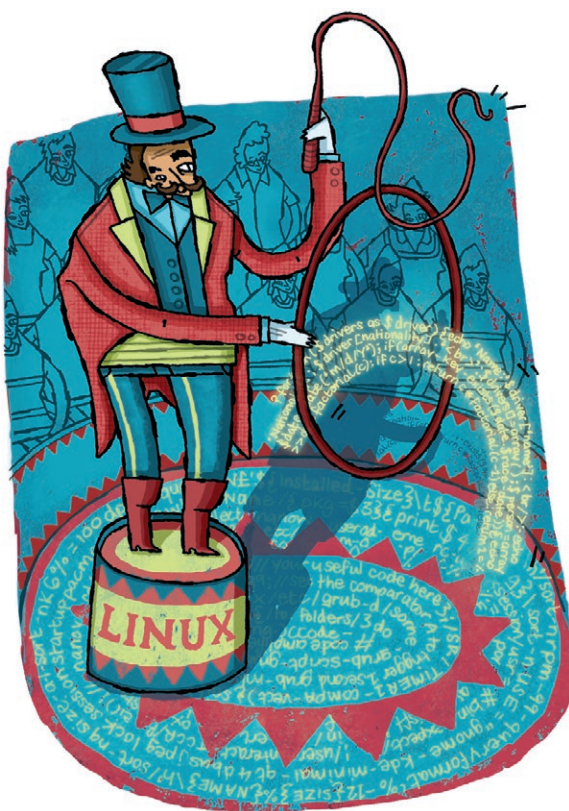
# Lua: Создаем скрипты Conky

Михалис Цукалос рассматривает основы программирования в Lua и создания скриптов, пригодных для применения в системном мониторе *Conky*.



**Наш эксперт**

**Михалис Цукалос** — администратор Unix, программист DBA и математик; он любит писать статьи и узнавать что-то новое.



Эта статья о Lua, легковесном и быстром встраиваемом скриптовом языке, и о том, как его скрипты могут использоваться в легковесном же системном мониторе *Conky*. Lua (что по-португальски означает «луна») был создан в 1993 г. Роберто Иерусалимски [Roberto Ierusalimsky] — и ныне это ведущий скриптовый язык для игр. Главное преимущество Lua в том, что он позволяет собирать всё самим, что одновременно является и его главным недостатком, поскольку иногда, в конечном итоге, вам и приходится собирать всё самим!

В Ubuntu вы можете установить Lua, выполнив `apt-get install lua5.2` от имени root. Вы найдете точную версию используемого вами Lua, запустив `lua -v`, что покажет вам нечто вроде `Lua 5.2.3 Copyright (C) 1994-2013 Lua.org, PUC-Rio`

Lua — небольшой язык с простым синтаксисом, но массой возможностей, а версия Lua программы “Hello World!” — это просто строка `print (“Hello World!”)`. Вы можете сохранить вышеупомянутую строку в файле с именем **hw.lua**, что выполнимо различными способами. Первый из них — использование компилятора Lua:

```
$ lua -o hw.luac hw.lua
```

```
1. bash
iMac:code mtsouk$ cat factorial.lua
function factorial(n)
  if n == 0 then
    return 1
  else
    return n * factorial(n-1)
  end
end

for i=3, 15 do
  io.write("The factorial of ", i, " is ", factorial(i), "\n")
end

iMac:code mtsouk$ vi factorial.lua
iMac:code mtsouk$ lua factorial.lua
lua: factorial.lua:5: stack overflow
stack traceback:
  factorial.lua:5: in function 'factorial'
  factorial.lua:5: in function 'factorial'
```

➤ Если вы попытаетесь вычислить в Lua факториал отрицательного целого числа, программа откажется это делать: она не умеет работать с такими числами.

```
$ ls -l hw.luac
-rw-r--r-- 1 mtsouk staff 157 Sep 30 10:31 hw.luac
$ lua hw.luac
Hello World!
$ file hw.luac
hw.luac: Lua bytecode, version 5.2
```

Если вы скомандуете `lua` без каких-либо аргументов, вас переправят в интерпретатор Lua, где можно вызвать **hw.lua** следующим образом:

```
> f = loadfile("hw.lua")
> f()
Hello World!
```

## Простой пример на Lua

Есть и другой способ запуска Lua, включающий создание автономного скрипта Lua. Чтобы создать скриптовую версию программы “Hello World!”, вам понадобится сделать

```
$ cat hw.script
#!/usr/bin/env lua
print("Hello World!")
$ chmod 755 hw.script
$ ./hw.script
Hello World!
```

Если у вас уже есть скомпилированный файл, можете включить его в скрипт и запустить его как скрипт с помощью следующей техники:

```
#!/usr/bin/env lua
dofile("./hw.luac")
```

Теперь вы знаете, как компилировать и выполнять код Lua; пора запрограммировать нечто поинтереснее.

## Скорая помощь

Документацию Lua и обширный список книг по Lua см. на [www.lua.org/docs.html](http://www.lua.org/docs.html). FAQ по Lua имеется на [www.lua.org/faq.html](http://www.lua.org/faq.html).

## Системное программирование на Lua

Хотя Lua не является языком системного программирования, он может взаимодействовать с ОС с помощью библиотеки Operating System. Однако для работы с файлами эта библиотека предлагает только две функции: `os.rename` и `os.remove`, используемые для переименования и удаления файла соответственно. Следующий код Lua показывает, как читать переменные среды:

```
> print(os.getenv("PATH"))
/usr/local/bin:/usr/bin:/bin:/usr/sbin:/sbin
> print(os.getenv("DOESNOTEXIST"))
nil
```

Как видите, если переменная, которую вы пытаетесь прочитать, не определена, вызов возвращает `nil`, то есть ноль или «ничего».

Функция `os.clock()` возвращает примерное количество времени CPU в секундах. Вы можете вызвать ее следующим образом:

```
> print(os.clock())
0.003559
```

Функция `os.difftime(t2, t1)` возвращает разницу в секундах между двумя временами, тогда как `os.date()` возвращает дату в виде строковой переменной в формате MM/DD/YY HH:MM:SS. Функция `os.execute([команда])` позволяет запускать внешние команды и аналогична системному вызову `system(3)` в C. Команда `os.exit([код])` использует функцию `C exit(3)`, чтобы завершить программу — внешние команды и аналогична системному вызову `system(3)` в C. Команда `os.exit([код])` использует функцию `C exit(3)`, чтобы завершить программу — внешние команды и аналогична системному вызову `system(3)` в C. Команда `os.tmpname()` создает временный файл,

но не открывает его, предоставляя это занятие разработчику. Следующий код Lua показывает, как извлечь различные компоненты из результата команды `os.date()`:

```
> t = os.date(" *t ")
> for key, value in pairs(t) do print(key,value) end
```

Первая команда создает таблицу, которая обрабатывается и печатается с помощью команды `pairs()`, позволяющей перебирать пары ключей и значений. Учтите, что `table.foreach()` — устаревшая форма, и ее следует заменять функцией `pairs()`.

Как вы, вероятно, поняли из нашего рассказа, занятие системным программированием в Lua — задача не из легких.

Следующий код Lua находит числа, принадлежащие последовательности Фибоначчи:

```
function Fibonacci(n)
  local function inner(m)
    if m < 2 then
      return m
    end
    return inner(m-1) + inner(m-2)
  end
  return inner(n)
end
for i = 1, 10 do
  print("For i = " .. i .. " fibonacci number is " .. Fibonacci(i))
end
```

Вначале определим функцию с именем `Fibonacci`. Затем определим локальную функцию с именем `inner()` — это функция, не действующая вне ограниченной области. По существу, вы сохраняете функцию, используемую рекурсивно, в локальной переменной (скриншот на стр. 80 показывает, как вычислить в Lua факториал целого числа без знака, не используя рекурсию). Функция `io.write()`, используемая для печати результатов, похожа на функцию `printf()` в C. Если вы попытаетесь вычислить факториал отрицательного целого числа, программа остановит исполнение, выдав сообщение об ошибке "stack overflow [переполнение стека]", поскольку при использовании с отрицательными числами текущий алгоритм создаст бесконечный цикл.

### Создание классов в Lua

В Lua «бесклассовая система»! Зато есть отличные средства метапрограммирования, облегчающие определение объектов класса. В объектно-ориентированных языках каждый объект — это экземпляр определенного класса. Lua не содержит концепции класса; каждый объект сам определяет свое поведение и имеет собственную форму. У каждого объекта может существовать прототип — эталонный объект, где первый объект отыскивает все неизвестные ему операции. Чтобы представить класс в подобных языках, вы просто реализуете некий объект для использования исключительно как прототип для других объектов. В Lua прототипы создаются следующим образом:

```
setmetatable(a, {__index = aaa})
```

По выполнении данной команды `a` будет искать каждую операцию, которой в нем нет, в `aaa`. Другими словами, `a` — это объект

класса `aaa`. Способов определить и реализовать классы имеется много, но следующий фрагмент кода покажет один частный метод, одновременно и гибкий, и вполне общепринятый:

```
Points = {}
Points.__index = Points
function Points.create(pnts)
  local newObject = {}
  setmetatable(newObject,Points)
  newObject.pnts = pnts
  return newObject
end
function Points:win()
  self.pnts = self.pnts + 3
end
function Points:loss()
  self.pnts = self.pnts + 0
end
function Points:draw()
  self.pnts = self.pnts + 1
end
function Points:punished(punishmentPoints)
  self.pnts = self.pnts - punishmentPoints
end
aTeam = Points.create(30) -- инициализация объекта
aTeam:win() -- pnts стал равен 30+3
aTeam:draw() -- pnts стал равен 33+1
aTeam:punished(10) -- pnts стал равен 34-10
print(aTeam.pnts)
```

В вредставленном методе используется команда `setmetatable`. Чтобы создать другие «компании», которые ведут себя так же, как `Points`, вы обеспечиваете наследование этими новыми объектами их операций из `Points` с помощью мета-метода `__index`. Lua пытается отыскать функцию `punished()` в `aTeam` и не может ее найти. Поскольку у `aTeam` есть мета-табличка, которая определяет `__index`, позднее он будет искать функцию `punished()` в этой мета-табличке. Следовательно, `aTeam:punished(10)` на самом деле является вызовом `Points`.

```
aTeam:punished(aTeam,10).
```

Как вы, видимо, догадались, строки, которые начинаются с "—", являются в Lua комментариями. Подобным образом можно также приставить комментарий в конце строки после записанного в строке кода Lua.

### Скорая помощь

Установить *Conky* в Ubuntu можно с помощью `sudo apt-get install conky`. В Arch Linux понадобится использовать `распан -S conky` от имени `root`. Текущую конфигурацию *Conky* покажет команда `conky -C`, выдав результат приблизительно в 80 строк.

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

### Скорая помощь



Следующие ключевые слова зарезервированы и не могут быть использованы в качестве имен в Lua: and, break, do, else, elseif, end, false, for, function, if, in, local, nil, not, or, repeat, return, then, true, until, while. Пожалуйста, учтите, что для Lua также важен регистр.

Настало время узнать больше о других функциях Lua, прежде чем использовать их для расширения *Conky*. Функции в Lua — это граждане первой категории, следовательно, они могут быть переданы в качестве параметров функции и присвоены другим переменным. Lua — язык программирования с динамической типизацией, то есть переменные во время выполнения обладают типом, несмотря на тот факт, что в программе тип переменных обычно не указан. Пока вы не узнаете больше о Lua, можете трактовать термин «мета-таблица» как «таблица позади таблицы». Другой синоним для этого — «настраиваемое поведение».

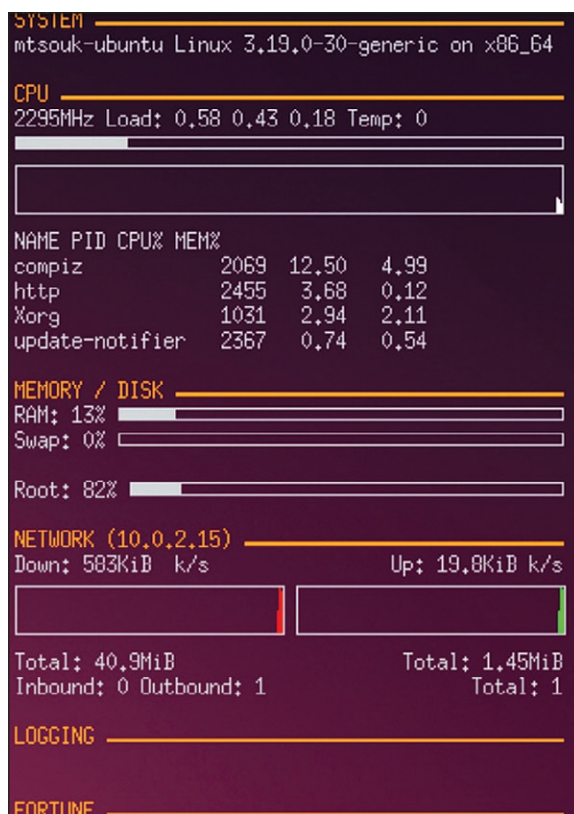
Последнее, что вам стоит знать — *Conky* работает со скриптами Lua, так что при разработке для *Conky* вашим продуктом на выходе должен быть скрипт Lua. Следовательно, чем лучше вы знаете Lua, тем проще будет ваша работа с *Conky*. Однако скрипты, используемые в *Conky*, не надо начинать со строки `#!/usr/bin/env lua`.

## Настройка Conky

*Conky* — это легковесный инструмент системного мониторинга для системы *X Windows*, который способен отобразить любую информацию на вашем экране. Хорошим свойством *Conky* является расширяемость с помощью встроенной поддержки Lua. *Conky* поддерживает различные параметры командной строки. Наиболее полезный из них — `-d`: он заставляет *Conky* работать в фоновом режиме без привязки к терминалу. Поскольку при работе *Conky* обычно выдает сообщения, лучше совместить `-d` с `-q` [quiet — англ. тихий], чтобы *Conky* помалкивал.

Для использования *Conky* с Lua понадобится установить дополнительный пакет. В Ubuntu Linux требуемые для этого шаги выглядят следующим образом:

```
$ wget http://gnome-look.org/CONTENT/content-files/139024-Conky-lua1.tar.gz
$ tar xzvf 139024-Conky-lua1.tar.gz
$ cd Conky-lua/
```



В директории *Conky-lua* есть множество файлов *tar.gz*, названных в честь различных дистрибутивов Linux. Для системы с Ubuntu надо распаковать файл *Conky ubuntu-lua.tar.gz* — создастся директория с именем *Conky ubuntu-lua*, где окажутся следующие файлы: *clock\_rings.lua*, *conkyrc* и *new-ubuntu-logo.png*. Файл *conkyrc*, который вы найдете в директории *Conky ubuntu-lua*, надо будет употребить для настройки *Conky*, всего лишь выполнив `sr conkyrc ~/.conkyrc`. О файле *.conkyrc* вы скоро узнаете больше, но сейчас просто взгляните на следующий вывод, показывающий, как *Conky* использует скрипты Lua:

```
$ grep lua conkyrc
lua_load ~/.lua/scripts/clock_rings.lua
lua_draw_hook_pre clock_rings
```

Файл *clock\_rings.lua*, упомянутый в выводе команды `grep`, можно обнаружить в директории *Conky ubuntu-lua*. Файл *clock\_rings.lua* должен содержать функцию, упомянутую после команды `lua_draw_hook_pre`, со строкой `conky_` перед именем. Если функция `conky_clock_rings()` отсутствует, вам будут выдаваться различные сообщения об ошибках при запуске *Conky*, наподобие, например, такого:

```
Conky: lua_do_call: function conky_clock_rings execution failed:
attempt to call a nil value
[Выполнить не удалось из-за попытки обратиться
к величине nil]
```

Запускаться *Conky*, тем не менее, будет, но область, зарезервированная для часов, будет пустой. Следует прояснить, что файл *clock\_rings.lua* содержит только код Lua, подобный тому, что вы видели в первой части нашего урока. Если вы открываете существующий скрипт Lua, работающий в *Conky*, и не уверены, какую функцию следует задать в *.conkyrc*, ищите имя функции, которое начинается с `conky_`.

Как и с большинством приложений Linux, вы можете изменять большинство опций *Conky* и хранить их в текстовом файле под названием *.conkyrc*, который должен быть в вашей директории *home*. Это обычный текстовый файл, который можно изменять с помощью вашего любимого текстового редактора (пример *Conky*, запущенного с измененным файлом *.conkyrc*, см. на рис. внизу слева). Пожалуйста, помните, что *Conky* требует немало системных ресурсов, так что не отображайте слишком много полей и информации на изрядно загруженном клиентском или серверном компьютере Linux.

Чтобы поддерживать порядок, лучше поместить файл *clock\_rings.lua* в отдельную директорию, так что сделайте такую директорию под названием *.conky* в вашей директории *home* и скопируйте файл *clock\_rings.lua* туда. Чтобы изменения вступили в силу, понадобится изменить файл *.conkyrc* в директории *Conky ubuntu-lua*. Следует заменить строку

```
lua_load ~/.lua/scripts/clock_rings.lua
```

на

```
lua_load ~/.conky/clock_rings.lua
```

Эта строка сообщит *Conky* местоположение скрипта для использования, а строка, начинающаяся с `lua_draw_hook_pre`, сообщит *Conky*, какую функцию запустить из загруженного скрипта Lua. Прежде чем приступить к оставшейся части урока, пожалуйста, убедитесь, что в вашем файле *.conkyrc* нет ошибок синтаксиса, запустив *Conky* и осмотрев полученный вывод. Файл *.conkyrc*, поставляемый с *Conky* в *ubuntu-lua.tar.gz*, как раз и придает *Conky* тот вид, который показан на стр. 83 (вверху справа) и который столь впечатляет и столь отличается от внешнего вида *Conky* по умолчанию или от нашего индивидуального *.conkyrc* (см. слева). Как нетрудно догадаться, сравнив эти экранные снимки, внешний вид *Conky* можно радикально менять.

➤ **Запуск Conky с индивидуальным файлом .conkyrc** позволит включить нужную вам информацию в вывод. Но будьте осторожнее при редактировании существующих файлов *.conkyrc*: они бывают довольно-таки сложны.

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

## Простые скрипты Lua

Представленный код Lua заставит *Conky* отображать сообщение "Hello World!". Как вы можете представить, понадобится создать функцию Lua в отдельном файле с именем **hwConky.lua** и заставить *Conky* загрузить эту функцию. Код Lua для реализации функции `conky_hello_world` выглядит следующим образом:

```
require 'cairo'
function conky_hello_world()
-- Стандартный код, который должен быть представлен только
в conky_function
if conky_window == nil then
return
end
local cs = cairo_xlib_surface_create(conky_window.display,
conky_window.drawable,
conky_window.visual,
conky_window.width,
conky_window.height)
cr = cairo_create(cs)
print ("Hello world!")
cairo_destroy(cr)
cairo_surface_destroy(cs)
cr=nil
end
```

Код прост и легок для понимания. Первая строка показывает, что скрипт собирается использовать `cairo` — это имя графической библиотеки, которую Lua будет применять для отрисовки графики, отображаемой в *Conky*. Затем начинается реализация `conky_hello_world()`. Первые строки — это стандартный код, и они должны присутствовать только в функции `conky_`. Они используются для описания фона, на котором Lua будет рисовать текст и графику; каждый скрипт Lua должен содержать эти строки. Далее дается команда `print()`. Интересный момент заключается в том, что ее вывод может наблюдаться в терминале, который вы использовали для выполнения бинарника **conky**, а не в графической области *Conky* на вашем экране. А плохо то, что сообщение будет отображаться при каждом обновлении *Conky*.

Последние три команды немного почистят память.

Стоит также внести следующие два изменения в файл **.conkyrc**, чтобы загрузить желаемую функцию:

```
lua_load ~/.lua/scripts/hwConky.lua
lua_draw_hook_pre hello_world
```

Слегка измененная версия `conky_hello_world` пригодится для того, чтобы и в самом деле отобразить ваш текст на экране. Пожалуйста, включите следующие строки кода после вызова функции `print()` и перезапустите *Conky*:

```
font="Mono"
fontsize=14
x = 100
y = 200
cairo_set_source_rgba (cr,1,1,1,1)
cairo_select_font_face (cr, font, CAIRO_FONT_
SLANT_NORMAL, CAIRO_FONT_WEIGHT_BOLD);
cairo_set_font_size (cr, fontsize);
cairo_move_to (cr,x,y)
cairo_show_text (cr,"Hello LinuxFormat!")
cairo_stroke(cr)
```

Теперь давайте немного попрактикуемся в написании кода Lua для разработки чего-нибудь более полезного.

## Больше скриптов Lua

Имя файла Lua, который вы ищете на **LXFDVD** — **LXF.lua**, а имя главной функции — `conky_uptime()`. Код в **LXF.lua** считывает три значения времени работы из оболочки и печатает их значения с помощью подхода, описанного нами ранее.

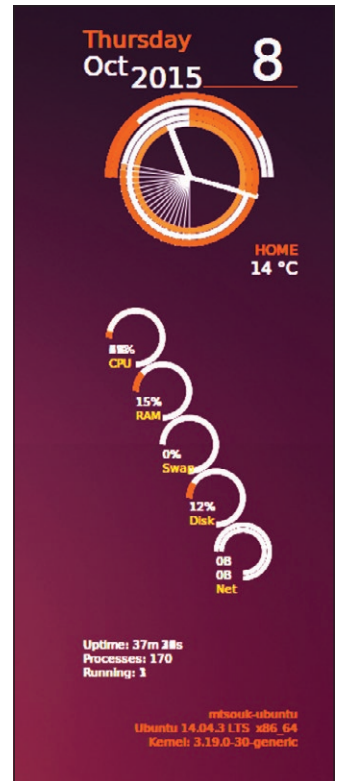
Помимо общего, большое отличие на сей раз заключается в том, что код вызывает внешнюю команду `uptime` с помощью функции `io.popen()` и получает желаемый результат с некоторой помощью `Awk`. Команда `os.execute()` не в состоянии сделать то же самое, поскольку она не позволяет получать результаты из выполненной команды. Так что хитрость по получению результата кроется в использовании следующих строк кода Lua:

```
local command = ("/usr/bin/uptime | /usr/bin/awk '{print $(NF-2)}'
\"$(NF-1)}' \"$(NF)}'")
local aHandle = assert( io.popen(command , 'r' ))
local text = aHandle:read("*l")
```

Первая строка определяет команду для выполнения, вторая выполняет команду как отдельный процесс и возвращает дескриптор файла, который вы можете использовать для считывания данных из программы. Наконец, последняя строка читает настоящие данные из дескриптора файла и хранит их в новой переменной.

Возможно, существуют более простые способы сделать то же самое, но этот подход более общий и может применяться во множестве различных ситуаций. Пожалуйста, не забудьте внести соответствующие изменения в **.conkyrc**, чтобы его использовать.

Помните, что раз вы умеете получать и отображать информацию с помощью Lua и *Conky*, ваши возможности ограничиваются только вашим воображением! **LXF**



» Вот как изменится вид *Conky* по умолчанию, если использовать файл **.conkyrc**, идущий с *Conky* в **ubuntulua.tar.gz**.

## О таблицах в Lua

Плохая новость в том, что таблицы в Lua являются единственным «контейнерным» типом. А хорошая новость — таблицы Lua очень гибкие и мощные, поскольку они являются ассоциативными массивами, то есть могут хранить любые пары символов и значений. Следующий код Lua показывает, как создать и использовать таблицу:

```
> aTable = {} -- define an empty table
> print(aTable)
table: 0x7f187235002b0
> aTable[0] = "zero"
> aTable[1] = "one"
> =>aTable[0]
zero
```

```
> return aTable[2]
nil
```

Требуется сначала объявить пустой массив, а потом уж добавлять к нему элементы. Как видно из последней строки предыдущего вывода, если нет значения, ассоциируемого с символом, возвращаемое значение будет "nil" — присвоение "nil" существующему ключу стирает из таблицы пару ключа и значения. Данный пример показывает, как использовать таблицу Lua в качестве традиционного массива, но вы также можете определять элементы как `aTable[«key»] = «value»`

Как вы поняли, массив Lua может содержать несколько типов ключей одновременно.

Таблицу легко создать (добавлять порядковые номера вручную не надо) следующим образом:

```
> anotherTable = {"one", "two", "three", "four"}
Заметьте, что индекс первого элемента 1.
Если вы намерены использовать в качестве ключей строковые константы, можете использовать специальный синтаксис клавиш быстрого запуска, предлагаемый Lua: > aTable.stringConstant = "anotherValue".
Следующий код показывает, как можно напечатать все элементы ассоциативного массива — весьма похоже на то, что делают другие языки программирования:
> for key,value in pairs(aTable) do print(key,value) end
```

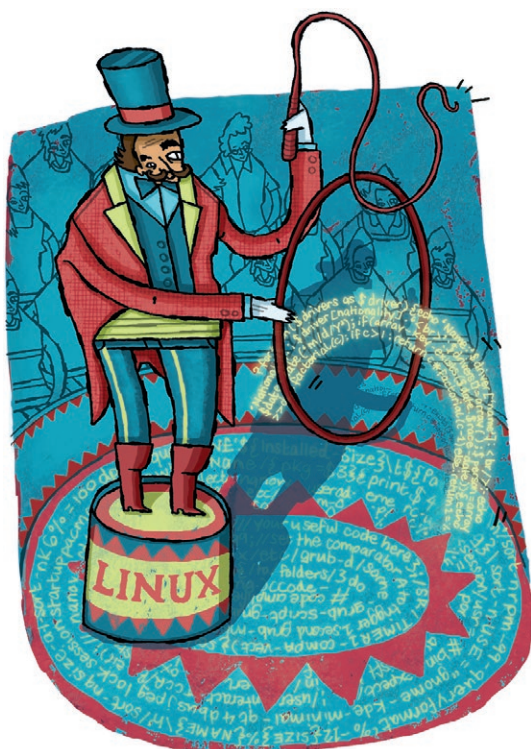
# Perl 6: Откроем новые черты

Михалис Цукалос рассказывает о вещах, которые необходимо знать, чтобы извлечь преимущество из уникальных функций Perl 6.



Наш эксперт

**Михалис Цукалос** — администратор Unix, программист, администратор баз данных и математик; он любит писать статьи и узнавать что-то новое.



Perl 6 — свежая версия языка — поддерживает объектно-ориентированное программирование, включая общие средства программирования, роли и множественную диспетчеризацию, а также простейшее функциональное программирование, включая оценку списка, соединения, автоматическую организацию поточной обработки и гипероператоры. Одна из главных новых функций — поддержка нескольких ядер, а также определяемой грамматики, что увеличивает возможности Perl по сопоставлению шаблонов и позволяет делать обобщенную обработку строк.

Мы воспользуемся компилятором Rakudo для кода Perl 6. В дистрибутиве Ubuntu Perl 6 можно установить командой `sudo apt-get install rakudo`. (Полный процесс установки изображен вверху.) Несмотря на имя пакета, исполняемый файл компилятора — **perl6**. Команда `$ perl6 -v` покажет точную версию, выведя нечто вроде `This is perl6 version 2013.12 built on parrot 5.9.0 revision 0`. Файл Perl 6 можно выполнить с помощью `$ perl6 file.pl`. Либо создайте скрипт:

```
$ perl6 file.pl
Другой способ создать скрипт —
$ cat hw.pl
#!/usr/bin/env perl6
use v6;
print "Hello World!\n";
```

```
mtsouk@mtsouk-ubuntu:~$ apt-cache search perl6
libmethod-signatures-perl - method and function declarations with signatures and no source filter
libparse-method-signatures-perl - Perl module to parse Perl6-like method signatures
libperl6-caller-perl - Perl6-like OO caller() interface for Perl 5
libperl6-export-attrs-perl - Perl 6 'is export(...)' trait as a Perl 5 attribute
libperl6-export-perl - Implements the Perl 6 'is export(...)' trait
libperl6-form-perl - perl - Perl6::Form - Implements the Perl 6 'form' built-in
libperl6-junction-perl - module providing Perl6-style Junction operators in Perl 5
libperl6-say-perl - print -- but no newline needed
libperl6-slurp-perl - module implementing the Perl 6 'slurp' built-in
libsyntax-keyword-junction-perl - Perl6 style Junction operators in Perl 5
nqp - Not Quite Perl compiler
rakudo - Perl 6 implementation on top of Parrot virtual machine
mtsouk@mtsouk-ubuntu:~$ sudo apt-get install rakudo
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  libparrots.9.0 nqp parrot parrot-minimal
Suggested packages:
  parrot-doc
The following NEW packages will be installed:
  libparrots.9.0 nqp parrot parrot-minimal rakudo
0 upgraded, 5 newly installed, 0 to remove and 0 not upgraded.
Need to get 4,359 kB of archives.
After this operation, 60.1 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://gr.archive.ubuntu.com/ubuntu/ trusty/universe libparrots.9.0 amd64 5.9.0-1build1 [521 kB]
Get:2 http://gr.archive.ubuntu.com/ubuntu/ trusty/universe parrot-minimal amd64 5.9.0-1build1 [22.5 kB]
Get:3 http://gr.archive.ubuntu.com/ubuntu/ trusty/universe parrot amd64 5.9.0-1build1 [45.9 kB]
Get:4 http://gr.archive.ubuntu.com/ubuntu/ trusty/universe nqp amd64 5.9.0-1build1 [22.5 kB]
Get:5 http://gr.archive.ubuntu.com/ubuntu/ trusty/universe rakudo amd64 5.9.0-1build1 [22.5 kB]
```

➤ Процесс установки компилятора Perl 6 в системе с Ubuntu. У вашего дистрибутива Linux, возможно, есть похожий пакет, который можно установить.

Выполнив `perl6` без аргументов или опций, вы войдете в REPL (цикл `read-eval-print` — прочитать-оценить-напечатать) — это новая функция. Кроме того, REPL — это оболочка, т.е. незамысловатая и интерактивная среда программирования, которая принимает вводимые данные от одиночного пользователя, обрабатывает их и сразу выдает результат. Это также удобно для освоения новых функций Perl 6.

## Новые изменения

Если файл программы содержит главную подпрограмму с именем `MAIN`, он автоматически выполнится при запуске программы. Это может пригодиться для получения аргументов и опций командного строки, поскольку дает бесплатную программу синтаксического анализа. Код ниже (`readWords.pl`) демонстрирует это:

```
use v6;
my $count = 0;
sub MAIN($file) {
    print "Файл: $file\n";
    for $file.IO.words -> $word {
        $count++;
    }
    print("В файле $file содержится $count слов\n");
}
```

Код также показывает новый способ чтения слов из файла. Как видите, теперь больше не требуется открывать текстовый файл для чтения: прочитайте его построчно и закройте. Подпрограмма

## Скорая помощь

От Perl 6 нигде не деться, и чем скорее вы освоите его новые функции, тем лучше будет для вас. Вдобавок, Perl 6, определенно, сделает вас лучшим и более продуктивным программистом, так что держаться его, несомненно, стоит!

MAIN требует одного параметра командой строки — если передать ей два или более аргументов, она автоматически выдаст следующее сообщение об ошибке:

```
$ perl6 readWords.pl readWords.pl readWords.pl
```

Usage:

```
readWords.pl <file>
```

Вдобавок можно настроить сообщения об ошибках, определив подпрограмму с именем USAGE, которая будет автоматически вызываться при неверном количестве аргументов командной строки согласно подпрограмме MAIN. Пожалуйста, учтите, что доступ к файлу с помощью open() и close() все еще осуществим. Желая пойти еще дальше, можете объявить MAIN множественной, что позволит объявлять различные другие синтаксисы. Вот как это выглядит:

```
use v6;
multi MAIN() { print "Не передан аргумент командной строки.\n"; }
multi MAIN($x) { print "Передан один аргумент командной строки.\n"; }
multi MAIN($x, $y) { print "Передано два аргумента командной строки.\n"; }
multi MAIN($x, $y, $z) { print "Передано три аргумента командной строки.\n"; }
sub USAGE {
    print "Передано слишком много аргументов командной строки!\n";
}
```

Все аргументы читаются как строковые переменные; позднее их следует конвертировать в другой формат. Также учтите, что сообщение об ошибке будет использоваться так, как определено в подпрограмме USAGE.

Следующая строка кода показывает, как прочитать весь файл и поместить его в массив данных, где каждый элемент массива — одиночная строка файла:

```
my @lines = "myFile".IO.lines;
```

Максимальное значение среди данных любого типа, поддерживающего упорядочение, легко найти с помощью встроенной функции max:

```
say max -10, -10, -15, -2, -12;
say max ["a", "2", "aa", "aaa"];
```

Если вы используете команду say, помещать символ перевода строки в конце команды не надо. А вот если вы используете print, в конце строковой переменной стоит поместить этот символ.

## Новые структуры и циклы

Первая управляющая структура, которую надо изучить, это конструкция given-when, способная изящно заменить выражения if-elsif-if. Вы можете увидеть ее в действии ниже (файл givenWhen.pl):

```
my $continue = 1;
while ( $continue )
```

```
ntsouk@ntsouk-ubuntu:~$ perl6 --help
perl6 [switches] [--] [programfile] [arguments]

With no arguments, enters a REPL. With a "[programfile]" or the
"-e" option, compiles the given program and by default also
executes the compiled code.

-c          check syntax only (runs BEGIN and CHECK blocks)
--doc      extract documentation and print it as text
-e program one line of program
-h, --help display this help text
-n        run program once for each line of input
-p path   same as -n, but also print $_ at the end of lines
-I path   adds the path to the module search path
-M module loads the module prior to running the program
--target=[stage] specify compilation stage to emit
--optimize=[level] use the given level of optimization (0..3)
-t, --trace=[flags] enable trace flags, see 'parrot --help-debug'
--encoding=[mode] specify string encoding mode
-o, --output=[name] specify name of output file
-v, --version display version information
--stagesstats display time spent in the compilation stages
--ll-exception display a low level backtrace on errors
--profile  print profile information to standard error

Note that only boolean single-letter options may be bundled.

Output from --profile can be visualized by kcachegrind.

To modify the include path, you can set the PERL6LIB environment variable:
PERL6LIB="lib" perl6 example.pl

For more information, see the perl6(1) man page.
```

```
{
# Считываем значение
my $input = prompt "Выбор между 0 (Выход), 1 и 2: <";
# Разбираем его
given $input {
when "0" {
print("Выход.\n");
$continue = 0;
}
when "1" { print("Передано 1!\n"); }
when "2" { print("Передано 2!\n"); }
default { print(Ошибка. Пожалуйста, повторите ввод!\n>); }
}
}
```

Другое интересное изменение кроется в цикле for — теперь он называется не for, а loop:

```
loop (my $i = -5; $i <= 5; $i++) {
print $i ~ " ";
}
```


Как показывает приведенный код, конкатенация строк теперь использует тильду (~) вместо точки. Применив в Perl 6 знаменитый цикл for, вы получите такое сообщение об ошибке:

```
===SORRY [ИЗВИНИТЕ]!===
Unsupported use of C-style "for (;;)" loop; in Perl 6 please use "loop (;;)"
Неподдерживаемый цикл for в стиле C; пожалуйста, в Perl 6 пользуйтесь "loop (;;)"
```

Как вы уже знаете, цикл for теперь является итератором, позволяющим получить доступ к элементам массива или списка.

» Это вывод команды perl --help, показывающий все доступные опции командной строки perl6.

**Скорая помощь**



Главный вопрос — лучше ли Perl 6, чем Perl 5, или нет. Хотя об этом еще рано говорить, Perl 6 кажется намного лучшей, усовершенствованной версией. Perl 5 умеет делать то же самое, но Perl 6 делает всё изящнее и с более понятным кодом.

## Касательно строк

В новой версии языка становится намного проще конвертировать соответствующие строки в их числовое значение. Поскольку и числа, и строковые переменные являются объектами, перекодирование совершается с помощью встроенного объектного метода.

Метод base() принимает два аргумента: первый — целая часть числа, а второй, опциональный, определяет количество знаков, используемое для дробной части. Если второй параметр опущен, принимается значение по умолчанию — 0 для целых

чисел и большее число для других типов чисел. Метод chr превращает целое число в одиночный символ Юникода.

Объекты типа Str (встроенный класс) являются константами. Вы можете определить строковую константу следующим образом:

```
> my Str $str := "123";
123
> $str = "1234"; # Cannot be changed!
Cannot assign to an immutable value [Константу нельзя изменить]
```

Пожалуйста, обратите внимание на использование оператора :=. С его помощью Perl 6 поддерживает привязку; это означает, что \$str напрямую указывает на Str "123", и, следовательно, эту переменную больше нельзя менять. Как вы понимаете, оператор := работает и с другими типами переменных:

```
> my Int $nInt := 123;
123
> $nInt = 32;
Cannot assign to an immutable value [Константу нельзя изменить]
```



Однако есть хитрая проблема, описанная ниже:

```
my @values = ["1", "2", "3", "4", "5", "6"];
for @values <-> $value {
    $value = $value~" euros";
    print $value~" ";
}
for @values -> $value {
    $value = $value~" euros";
    print $value~" <";
}
```

Вы заметите, что если вы захотите изменить итерацию переменной в итерации for, понадобится использовать символ <->. При использовании символа -> переменная итерации будет доступна только на чтение. Это отнюдь не плохое нововведение, поскольку оно может оградить вас от многих проблем! Сообщение об ошибке, которое вы получите после запуска **for.pl**, выглядит следующим образом:

```
Cannot assign to a read only variable or a value [Нельзя присвоить значение переменной только для чтения]
```

in block <unit> at for.pl:7  
Напоследок, Perl 6 больше не поддерживает блок continue: внутри цикла вместо него надо использовать блок NEXT.

В Perl 5 вы бы написали так:

```
next if $line =~ /match/ ;
next if $line !~ /match/ ;
$line =~ s/xyz/123/;
```

А в Perl 6 надо будет, соответственно, написать так:

```
next if $line =~ /match/ ;
next if $line !~ /match/ ;
$line =~ s/xyz/123/;
```

Альтернатива — использовать в Perl 6 новые методы .match и .subst (о которых мы здесь не рассказали).

## Регулярные выражения

Perl 6 поддерживает именованные регулярные выражения и грамматики. Главным преимуществом этого нового подхода является не функциональность, остающаяся прежней, а улучшенная читаемость и уменьшение количества ошибок, поскольку сложные регулярные выражения Perl 5 были трудночитаемы и тяжелы для понимания. Приведем пример грамматики (**regExp.pl**), которая соответствует целым числам со знаком и без, а также числам с десятичной точкой:

```
#!/usr/bin/env perl6
use v6;
# Определим грамматику
my grammar checkInteger {
    rule TOP {
```

```
<integer>
}
token sign { <[-+]> }
token decimal { \d+ }
regex integer {
    <sign>?
    <decimal>+
}
regex isNumber {
    <sign>?
    <decimal>+
    "."
    <decimal>?
}
rule number {
    <isNumber>
}
}
# Используем грамматику
my $input = "123.3";
if checkInteger.parse($input) {
    say "$input является целым числом";
}
else {
    say "$input является целым числом!"
}
if checkInteger.parse($input, :rule<number> ) {
    say "$input является числом!";
}
}
```

Как видите, грамматика теперь представляет собой набор правил. Вначале вы определяете его, а затем используете. Поначалу это может показаться немного сложным, но в дальнейшем поможет писать улучшенный код, с меньшим количеством ошибок. При вызове .parse() грамматика попытается найти соответствие введенной строковой переменной регулярному выражению с именем TOP внутри грамматики. Если такого выражения не найдено, возникает ошибка. Как вы понимаете, TOP рассматривается как точка входа грамматики. Желая использовать другую точку входа, вы можете сделать это следующим образом:

```
checkInteger.parse($input, :rule<number> )
```

Предыдущая команда подвергает введенную информацию синтаксическому анализу, используя вместо TOP определенное правило с именем number.

Вдобавок, как и классы, грамматика может наследовать и отменять правила, и т.д. Она также позволяет выполняться другим командам, пока вы анализируете вводимую информацию. Грамматика Perl 6 настолько мощная, что способна анализировать даже целый язык программирования, включая сам Perl 6!

## Отличия Perl 6

Режим strict [взыскательный] теперь стоит по умолчанию. По умолчанию также отображаются предупреждения. Функции, которые были изменены autodie, чтобы отбрасывать исключения в ответ на ошибки и в тех случаях, когда возвращаемое значение не проверяется явно. use base и use parent в объявлении класса заменены в Perl 6 на ключевое слово is, как в следующем примере:

```
package aPackage; # Perl 5
use base qw(anotherName); # Perl 5
class aPackage is anotherName; # Perl 6
```

Константы в Perl 6 теперь объявляются так:

```
constant $VARIABLENAME = 0;
В Perl 5 вы бы объявили константу немного иначе:
use constant VARIABLENAME => 0;
```

Вдобавок, pi, e, i встроены в Perl 6 как константы, и больше незначим их явно определять. Теперь вы можете определять тип

### Скорая помощь

Более подробную информацию о Perl 6 можно найти на <http://perl6.org>, но ничто не заменит практики. Если вы не знаете, с чего начать, начните с реализации на Perl 6 упрощенных версий существующих утилит Unix.

➤ REPL в Perl 6 — отличное поле для экспериментов и пробы чего-то нового!

```
mtsouk@mtsouk-ubuntu: ~
mtsouk@mtsouk-ubuntu:~$ perl6 --version
This is perl6 version 2013.12 built on parrot 5.9.0 revision 0
mtsouk@mtsouk-ubuntu:~$ perl6
> ?
===SORRY!=== Error while compiling <unknown file>
Bogus statement
at <unknown file>:2
-----> <BOL> EOL>
    expecting any of:
      statement list
      prefix or term
      prefix or meta-prefix
> help
===SORRY!=== Error while compiling <unknown file>
Undeclared routine:
  help used at line 1

> my $line = "Linux Format Magazine!";
Linux Format Magazine!
> next if $line =~ m/ <[abc]> / ;
> next if $line =~ m:P5/[abc]/ ;
```



## Отложенные списки и ряды

Недоделанные (отложенные) списки — уникальная характеристика Perl 6, которая поначалу может показаться странной. Но прежде чем узнавать больше об отложенных бесконечных списках, рассмотрим код, определяющий такой список, а также выполняющий некоторые вычисления:

```
my @fib = 0, 1, "*" ... *;
say "Fibonacci number #5 is @fib[4]";
```

Отложенные списки похожи на массивы, но с несколькими основными отличиями.

Во-первых, у них не обязательно предопределен размер; они даже могут быть бесконечными. Во-вторых, они вычисляют свои значения не заранее, а только при необходимости. И наконец, вычисленное значение может сохраняться для быстрой проверки. Следует прояснить, что бесконечные списки поддерживаются из-за их свойства отложенности. Противоположностью бесконечным спискам являются неотложенные списки [eager list], подобные массивам C. Perl 6 поддерживает и неотложенные,

и бесконечные списки. Но поскольку отложенные списки более эффективны с точки зрения памяти, Perl 6 пытается по возможности использовать их.

В Perl 6 также поддерживаются диапазоны, отложенные по умолчанию. Следующий код определяет конечный и бесконечный диапазоны:

```
# Конечный список
my @iList = 1..20000;
# Бесконечный список
my @iList = 1..Inf;
```

значений, которые переменная может хранить. Это делается посредством добавления имени типа к объявлению переменной, как показывает следующий пример:

```
my Int $i = 3;
my Numeric $a = 2.3;
```

Роли Numeric определяют число или объект, способный действовать как число, что включает целые (Int), рациональные (Rat) числа и числа с плавающей запятой (Num). Попытка поместить неверный тип значения в переменную вызовет ошибку (как показано на рис. справа). Тип переменной можно узнать, применив метод .WHAT. Похожим образом можно проверить, принадлежит ли что-то к определенному типу, вот так:

```
> $x = 123;
> if $x.WHAT === Int {say "Это целое число!";}
> say "321".WHAT
(Str)
> if $x.isa(Int) {say "Это целое число!";}
Это целое число!
> if !$x.isa(Str) {say "Это не строка!";}
Это не строка!
>
```

Заметьте, что сравнивая возвращенное значение метода WHAT, вы должны использовать оператор ===. Методы WHAT() и isa() очень удобны для проверки, тот ли у вас вид объекта или как обрабатывать объект согласно его типу.

## Обратная совместимость

Из того, что вы уже видели, должно быть ясно, что для работы с компилятором Perl 6 готовый код Perl 5 потребует изменений. Что до регулярных выражений — если у вас есть сложное регулярное выражение Perl 5, которое вы хотите без изменений использовать в Perl 6, воспользуйтесь модификатором P5, как в следующем примере:

```
# Perl 5 code
next if $line =~ m/[abc]/;
# Код Perl 6, использующий модификатор P5
next if $line =~ m:P5/[abc]/;
# Новый код Perl 6
next if $line =~ m/<[abc]>/;
```

Как вы уже видели, цикл for теперь используется только для итераций по спискам, так что для использования в Perl 6 понадобится изменить свои циклы for. Подпрограммы теперь также определяются с помощью ключевого слова sub, а параметры в подпрограммах по умолчанию являются неизменяемыми [read-only]. Единственный способ изменить их — использовать следующий метод:

```
$ cat subs.pl
#!/usr/bin/env perl6
use v6;
sub changeMe($var is rw) { $var++; return $var; }
```

```
2. mtsouk@mail: ~ (moar)
rMacBook:perl mtsouk$ perl6
> my Numeric $a = 2.3;
2.3
> my Int $i = 3;
3
> $i = 2.1;
Type check failed in assignment to '$i'; expected 'Int' but got 'Rat'
in block <unit> at <unknown file>:1

> $a = "123";
Type check failed in assignment to '$a'; expected 'Numeric' but got 'Str'
in block <unit> at <unknown file>:1

> say $i.WHAT
(Int)
> say $a.WHAT
(Rat)
> my Rat $rational = 123.3
```

```
sub cannotBeChanged( $var ) { $var = 2; }
my $myVar = 12;
$myVar = changeMe($myVar);
say $myVar;
cannotBeChanged($myVar);
$ ./subs.pl
13
Cannot assign to a readonly variable or a value
in sub cannotBeChanged at ./subs.pl:4
in block <unit> at ./subs.pl:8
```

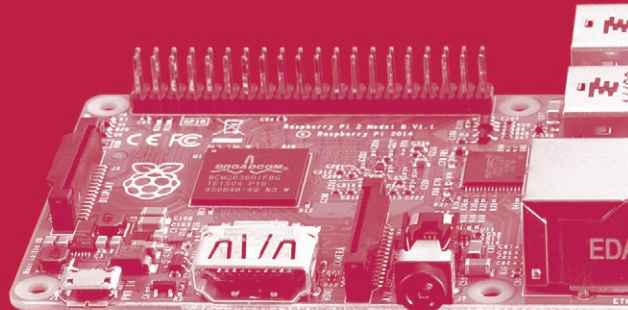
Подпрограммы Perl 6 также поддерживают «заглатывание» параметров, применяемое в тех случаях, когда точное количество параметров, которые получит подпрограмма, заранее неизвестно:

```
#!/usr/bin/env perl6
use v6;
sub unknow($first, $second, *@remaining)
{
print "First = $first, Second = $second\n";
say "Remaining parameters: @remaining[]";
}
unknow(1, 2, 3, 4.1, 5, 6, 7, "eight", ["a", "l", "i", "s", "t"]);
```

Если вы не уверены насчет действия команды или функции, всегда можно опробовать ее в REPL и посмотреть, работает ли она (см. пример на рис. стр. 86 внизу). Как вы также видите из последней данной команды, «недоделанный» список не делает возврата в REPL, и вы должны прервать его вручную.

Надеемся, что теперь вы убедились: Perl 6 поддерживает больше парадигм программирования, и у него полезные и более информативные предупреждения и сообщения об ошибках, чем в Perl 5. Perl 6 очень скоро будет основной версией, так что необходимость узнать больше о языке и начать использовать его в ваших новых проектах весьма очевидна. Однако мы бы советовали не использовать его в первом же крупном проекте, а начать с проектов поменьше. **LXF**

➤ **Определив тип величины, впоследствии вы не сможете изменить его значение. Это отличный способ снижения количества дурацких ошибок в программах.**



Ваша порция смачных новостей, обзоров и учебников от Raspberry Pi



дэйв джонс  
Создатель библиотеки Picraft

Привет...

**Р**icraft начался с того, что придумали способ передавать в *Minecraft* снимок, сделанный камерой Pi. Я хотел сделать то же для видео, но увидел, что в *Minecraft Pi Edition (MCPi)* не хватает скорости. И забросил эту идею, но наткнулся на схожее решение для Sonic Pi: спектроанализатор в реальном времени для *Minecraft*. Он обновлялся гораздо быстрее, и я понял, что надо объединить запросы в одну сетевую транзакцию — и стал проверять, нельзя ли применить нечто подобное в *MCPi*. Увы, чем глубже я копал, тем больше обнаруживал всяких неприятностей (Горбатые Регистры, методы чтения, механизмы настройки и т.д.). И я решил все переделать.

Начал я с азов. Создал приличный класс Vector с поддержкой всех известных мне операций, причем неизменяемый, чтобы применять его как ключ в словарях и наборах (dict и set). Потом класс Block с конструктором переменных, чтобы можно было затребовать блок, соответствующий цветовой спецификации HTML, и класс Connection, с возможностью групповой обработки. Ну и дошел до банальных классов World [Мир], Player [Игрок] и Events [События]. Затем всякими способами расширил API, внедрил кое-какие элементы из прекрасной библиотеки *Minecraft* Мартина О'Хэнлона [Martin O'Hanlon].

Напоследок я поэкспериментировал с новыми обработчиками событий. Теперь можно пометить функцию как относящуюся к конкретному блоку по его местоположению, или назначить функцию как ответ игроку, перемещающемуся по конкретной территории. Подробнее о *Picraft* — на <http://picraft.readthedocs.org>.

## Pi + Code Club

Raspberry Pi Foundation и Code Club объединяют усилия для невероятного обучающего потенциала.

**С** 2012 г. было продано 7 миллионов Pi и создано более 3800 клубов программирования, и сегодня по всей Британии более 40000 детей, 40% из которых — девочки, регулярно посещают Code Club'ы. Но этого, безусловно, мало. И Foundation, и Code Club хотят добиться большего и стремятся сделать Code Club полноправной дочерней компанией Raspberry Pi Foundation.

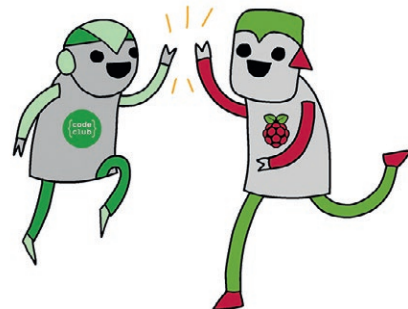
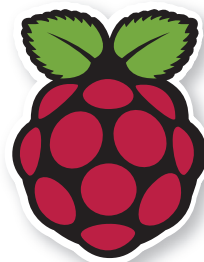
Как нас заверили в Pi Foundation, сам бренд и его подход останутся прежними, поскольку это уже проверенная модель, прекрасно себя зарекомендовавшая. Для тех, кто возглавляет клубы, ничего не изменится, Code Club HQ будет по-прежнему создавать невероятные проекты на любом оборудовании, какое им нравится.

Pi Foundation усматривают в этом отличную возможность внести разнообразие в школьную

программу и еще больше расширить свою аудиторию за счет нейтрального Code Club.

Нам кажется, что этот союз заключен на небесах, и сами члены Code Club встретили его с присутствием их юному возрасту радостным воодушевлением. Более подробную информацию ищите на [www.codeclub.org.uk](http://www.codeclub.org.uk).

➤ Союз, заключенный на цифровых небесах: Code Club и Raspberry Pi Foundation.



## Pi на заказ

Находя для Raspberry Pi коммерческое применение, Element14 предлагает сервис индивидуальной сборки.

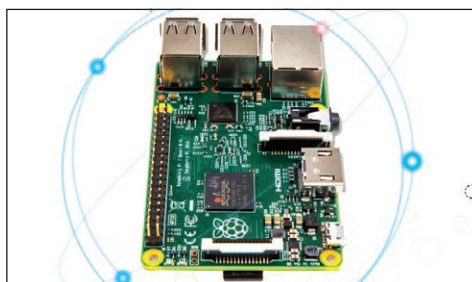
**И**нтернет вещей (Internet of Things, IoT) укрепляет свои позиции на рынке, и ключевую роль в этом играет Raspberry Pi. Никто не ожидал, что Pi сам по себе будет столь успешен, тем более станет реальной фигурой на рынке IoT. Но благодаря крошечным размерам, малому

энергопотреблению и низкой цене, Pi может стать сердцем практически любого IoT-проекта.

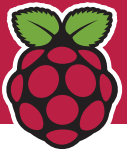
Единственным минусом была невозможность менять базовый дизайн Pi, чтобы он идеально встраивался в ваши IoT-проекты; однако новое партнерство между Raspberry Pi Trading (коммерческим подразделением Foundation) и Element14 исправляет это, предлагая эксклюзивные, доступные по всему миру услуги консультирования, проектирования и производства.

Любые платы Raspberry Pi и дополнительное оборудование к ним можно будет подобрать по индивидуальному заказу, неизменно на базе чипа Broadcom. Заказы от 3000 до 5000 образцов принимаются через Design Centre, стоимость и время выполнения варьируются в зависимости от проекта.

Новая услуга обещает распространить Pi во все уголки мира и в еще более удивительные проекты. За подробностями обращайтесь на [www.element14.com/custompi](http://www.element14.com/custompi).



➤ Мечтали добавить, убрать или настроить почти любой элемент структуры Pi? Теперь это возможно.



# Raspbian Jessie

Будучи знатоком всего связанного с Raspberry Pi, Лес Паундер обнюхивает последний релиз Raspbian от Pi Foundation.

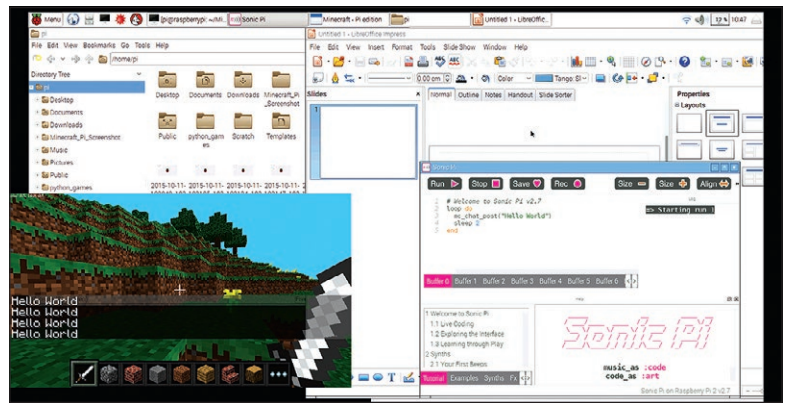
## Вкратце

» Последний релиз официально дистрибутива, поддерживаемого Raspberry Pi Foundation. Дистрибутив основан на Debian Jessie и внес множество улучшений в области оборудования и ПО, превратившись не только в созревшую всеохватывающую платформу для хакеров и разработчиков, но и в выгодный по цене второй компьютер.

В качестве официальной операционной системы Raspberry Pi с момента своего дебюта в 2012 г. употреблял Raspbian. Оглядываясь на ранние версии Raspbian, мы видим среднюю ОС, всё еще находящуюся во младенчестве; но последний релиз Raspbian вывел на арену совершенно нового зверя.

Последний Raspbian основан на Debian Jessie и идет с ядром 4.1.7 по умолчанию и улучшенным рабочим столом, который перешел на более новый инструментарию GTK 3 по инициативе Саймона Лонга [Simon Long] в 2014 г. При первом запуске заметно основное отличие процесса загрузки — теперь по умолчанию вы попадаете в рабочий стол Raspbian; но это легко изменить. Raspbian Jessie также новым способом вносит изменения в настройки системы, и этот способ — приложение *Raspberry Pi Configuration*, GUI для доблестного *raspi-config*, применяемого, среди прочего, для разгона Pi. Добавление приложений в главное меню теперь упростилось благодаря *Main Menu Editor*, похожему на *Alacarte*, написанный на языке Python. Впечатляет также добавление *scrot*, приложения для изготовления скриншотов, что мы частенько проделываем, меняя дистрибутивы.

Чтобы прочнее зацементировать идею использования Raspberry Pi как обычно стационарного ПК, Raspbian Jessie поставляется с комплектом *LibreOffice* и почтовым клиентом *Claws Mail*. Преимущества оборудования Pi 2 позволили ввести в меню Programming две новых IDE Java: BlueJ и Greenfoot.



» Рабочий стол Raspbian благодаря трудам Саймона Лонга сильно улучшил интерфейс пользователя и стал просто блестящим.

Возможно, главное изменение кроется внутри. Обычно только пользователю root или пользователю с доступом sudo разрешалось использовать выходы GPIO. Обходным путем для этого было обращение к *IDLE*, редактору Python, где надо было открыть терминал и запустить *IDLE* через *sudo*. Теперь это больше не проблема: любой пользователь получает доступ к выводам GPIO с помощью *IDLE* в меню Programming. Это главное изменение, которое намного упростит переход для тех, кто учится работать с кодом через GPIO.

## Мощная платформа

Кстати о Python: в Raspbian Jessie есть новая версия популярной библиотеки *Pygame* под названием *Pygame Zero*. *Pygame* часто используется для создания игр с помощью Python, но изучить ее непросто. *Pygame Zero* — упрощенная версия *Pygame* с акцентом на помощь преподавателям, которые желают обогатить уроки по работе с кодом. Любой код, написанный для *Pygame Zero*, использует текстовый редактор или *IDLE*, но чтобы запустить его, надо открыть терминал и набрать `python <название вашего проекта>`.

Благодаря этому релизу понятно, почему Raspbian считается дистрибутивом де-факто. Смесь из продуманных доработок для устоявшихся приложений, *raspi-config* и доступ к GPIO без *sudo*, плюс новое ПО, включая *LibreOffice*, позволяет Pi удовлетворять потребности самых разных групп пользователей, а не только программистов.

На установку Raspbian Jessie придется отвести более 4 ГБ, оставив меньше 3 ГБ

на 8-ГБ SD-карте. Это не столь высокие требования, поскольку Raspberry Pi поставляется с 8-ГБ картами micro-SD. Но владельцам более старых версий Pi, идущих с 4-ГБ SD-картой, или Raspberry Pi Compute придется немного подождать до выхода запланированной «облегченной» версии.

Мы протестировали этот новый релиз на Raspberry Pi B+ с 512 МБ ОЗУ и можем отпарировать, что он превосходно работает даже при меньшем объеме ОЗУ. Однако при 256 МБ ОЗУ, которые предусмотрены на A и A+, Pi работал медленнее, чего и следовало ожидать.

Raspbian Jessie продолжает лидировать в сообществе одноплатных компьютеров. А последний релиз Raspbian упрочивает свое лидерство, несмотря на возрастающую конкуренцию со стороны дистрибутива Ubuntu Mate. **LXF**

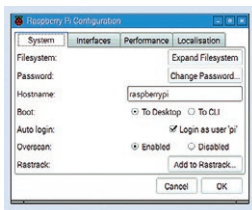


## Свойства навскидку



### Доступ к GPIO

Первый релиз, давший доступ к GPIO всем пользователям. Раньше только root или sudoer могли использовать разъемы.



### Настройки

Новый комплект программ позволяет каждому вдоволь повозиться с настройками, от раскладки клавиатуры до разгона Pi.

## LINUX FORMAT Вердикт

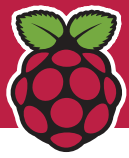
### Raspbian Jessie

Разработчик: Raspberry Pi Foundation  
Сайт: [www.raspberrypi.org](http://www.raspberrypi.org)  
Лицензия: Разные

Функциональность	9/10
Производительность	9/10
Удобство в работе	7/10
Документация	7/10

» С улучшением взаимодействия с пользователем благодаря доработанным инструментам стал мощной платформой.

## Рейтинг 9/10



# AstroCam: Фото, снятые Pi

Лес Паундер берет свой верный Pi, Умную Шляпу и официальную камеру Pi и создает камеру для юных искателей космических приключений.



**Наш эксперт**

Лес Паундер путешествует по Великобритании, работая с Raspberry Pi Foundation, и проводит для учителей свой популярный тренинг Picademy. Он также пишет блог на <http://bigl.es>.

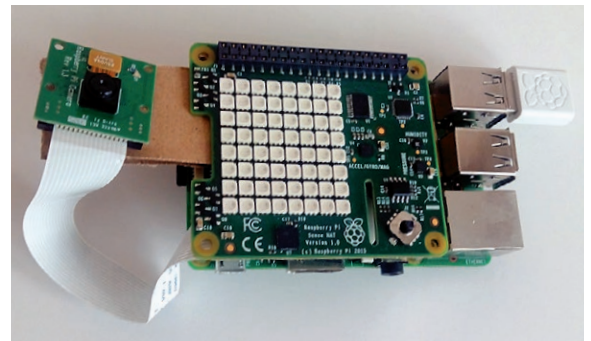
В прошлом выпуске мы представили вам Sense HAT [Hardware Attached on Top — плата расширения, hat — англ. шляпа], с которой будет работать миссия Astro Pi на борту Международной космической станции. На этом уроке мы употребим ее для управления камерой Raspberry Pi и задействуем джойстик для управления настройками, такими как таймер и вспышка и пуск затвора для создания фото. Для проекта в целом понадобятся Raspberry Pi 2, A+ или B+ и самый свежий релиз Raspbian, а также HAT и камера. Весь код см. на <http://bit.ly/LXF205AstroCam>.

Установка камеры и Sense Hat достаточно хитроумна и включает протяжку ленточного кабеля камеры через Sense Hat сквозь слот слева от джойстика. Позаботьтесь, чтобы голубой край соединения был обращен к порту Ethernet на вашем Pi, затем снимите пластиковую затычку на порте камеры, вставьте ленточный кабель и затычку, чтобы зафиксировать кабель, и присоедините Hat ко всем 40 выводам GPIO.

Для данного проекта мы настоятельно рекомендуем установить самую свежую версию Raspbian Jessie, поскольку в нее включена библиотека Sense Hat. Jessie также дает любому пользователю возможность получить доступ к выводам GPIO через Python и больше не требует запускать *IDLE* через *sudo*. Последнюю версию можно скачать с [www.raspberrypi.org/downloads](http://www.raspberrypi.org/downloads).

Загрузите свой Pi; запустится среда рабочего стола. Откройте главное меню, перейдите в Programming и выберите IDLE 3 (Python 3). IDLE 3 запустит интерактивную оболочку, которая нам сейчас не нужна — перейдите в File > New Window, чтобы открыть пустой документ, и сохраните пустой документ, File > Save as, с именем **astro-cam.py**. Код начнем с импорта библиотек, требуемых нашему проекту:

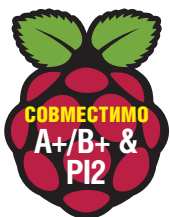
```
import pygame, time
from pygame.locals import *
```



► Наш завершённый проект прост в сборке, и его относительно легко интегрировать в фотокабину.

```
from picamera import PiCamera
from sense_hat import SenseHat
sense = SenseHat()
import datetime
pygame требуется для включения ввода с джойстика Sense Hat, который дублирует клавиши со стрелками и клавишу Enter клавиатуры. PiCamera даст нам доступ к камере. Мы импортировали библиотеку Sense Hat и ввели переменную, чтобы облегчить работу с синтаксисом. И, наконец, импортировали datetime, для создания временной отметки для имен файлов.
pygame.init()
screen = pygame.display.set_mode((640,480))
pygame.display.set_caption('Pygame Test')
```

Чтобы задействовать *pygame*, мы инициализировали ее; затем создали дисплей с разрешением 640×480 пикселей и назвали его окно *screen*.



**Скорая помощь**

Камера Raspberry Pi должна быть защищена от контактов с Raspberry Pi, особенно с выводами GPIO. Чехол для камеры можно купить на CPC (<http://cpc.farnell.com>) и Pimoroni (<https://shop.pimoroni.com>), дешевле — £5.

## PiCamera

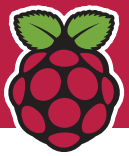
Официальная камера Raspberry Pi — одна из первых приставок, созданных для Raspberry Pi. Она использует 5-MP сенсор для создания видео 1080p и фотографий высокого качества с разрешением 2592×1944 пикселей, и работает со стабильной, богатой и грамотно написанной библиотекой Python.

При уменьшении разрешения до 640×480 пикселей возможна высокоскоростная съемка на 90 fps [frames per second, кадров в секунду], и это позволяет менять камеру для научных экспериментов

и в спорте (например, для фотофиниша). Камеру также можно рассматривать как новый вид устройства ввода информации, благодаря распознаванию цветов, встроенному в библиотеку *PiCamera* на Python.

Другой формой ввода является OpenCV, позволяющий компьютеру «видеть». Для этого пользователь должен запрограммировать компьютер на поиск и интерпретацию объектов/лиц и соответствующие действия. Эта форма ввода довольно накладна в плане компьютерных ресурсов и снижает частоту кадров камеры примерно

до 10 fps, однако ею можно успешно пользоваться. Фокус камеры не поддается изменению программными средствами. Он фиксирован, но это преодолели с помощью удаления клея вокруг линз ножом с тонким острием; однако мы бы не советовали этого делать. В последнюю версию ОС Raspbian, Jessie, включена преднастроенная и готовая к употреблению камера, но если у вас более старая версия Raspbian, понадобится добавить камеру через *raspi-config*, а затем через *pip3-2* установить на Raspberry Pi библиотеку *PiCamera*.



Далее создадим две функции — первая будет производить фотографирование:

```
def takepic(timer,toggle):
    for seconds in range(timer):
        sense.show_message(str(seconds), text_colour=[255,0,0],
        scroll_speed=0.05)
        time.sleep(1)
```

Сначала мы определили имя функции — `takepic`; мы также создали два аргумента, передаваемые функции. Эти аргументы — значение таймера и переключатель, управляющий вспышкой. В цикле `for` перебирается количество секунд, на которое установлен таймер. Цикл прокрутит значение таймера по светодиодной матрице Sense Hat, затем на секунду уснет и повторит процесс.

```
a = str(datetime.datetime.now())
a = a[0:19]
flash(toggle)
```

Мы вышли из цикла `for`, но остались в функции, и создали строковую переменную с именем `a`, записав в нее текущую дату и время, конвертированные в строку. Строку мы укоротили, используя только символы из позиций строки от 0 до 19 (`[0:19]`), т.е. настоящие дату и время. И, наконец, вызвали функцию `flash` со значением переключателя.

```
with PiCamera() as camera:
    temp = round(sense.get_temperature(),2)
    camera.resolution = (800, 600)
    camera.framerate = 24
    camera.start_preview()
    time.sleep(5)
    camera.stop_preview()
    camera.annotate_text = 'На этой картинке температура %s C'
    % temp
    time.sleep(0.1)
    camera.capture('/home/pi/'+a+'.jpg')
```

## Съемка

Начнем с создания переменной `temp`, для хранения текущей температуры. Снимем показания датчика температуры Sense Hat — его точность для нас сейчас чрезмерна, поэтому округлим его значение до двух десятичных знаков. Зададим разрешение камеры как  $800 \times 600$  пикселей, а частоту кадров — как 24 кадра в секунду, и на 5 секунд откроем окно предпросмотра, давая время кадрировать снимок до закрытия окна. Затем аннотируем изображение на включение текущей температуры перед тем, как перейти в спящий режим на 0,1 секунды, делаем снимок и сохраняем изображение в `home` с текущим временем и датой в качестве имени файла.

Наша последняя функция работает со вспышкой.

```
def flash(toggle):
    print(toggle)
    if toggle == 'on':
        sense.clear(255,255,255)
    elif toggle == 'off':
        sense.clear()
```

Мы уже ссылались на эту функцию в функции `takepic()`, и будем вызывать ее из `takepic()`. Функция `flash` имеет один аргумент, контролирующий включение/выключение вспышки. Если значение переключателя — `'on'`, все светодиоды устанавливаются на максимальную яркость; если значение переключателя — `'off'`, то светодиодная матрица выключается. Теперь перейдем к основному коду (см. <http://bit.ly/LXF205AstroCam>):

```
try:
    timer = 0
    while True:
        for event in pygame.event.get():
            if event.type == KEYDOWN:
```

## Обновление Scratch

В новейшей версии Scratch, выпущенной с новым образом Jessie, камеру Raspberry Pi можно применять для создания спрайтов, управляемых через палитру команд Scratch.

Новая версия Scratch также поддерживает платы расширения типа Sense Hat,

позволяющие детям воссоздавать такие проекты, как AstroCam. Работа над этим проектом еще ведется, но Raspberry Pi Foundation выделил значительные ресурсы на создание версии Scratch, отвечающей потребностям самых разных пользователей.

Мы начали с использования теста `try...except`, который в случае необходимости позволит изящно выйти из нашего кода. Здесь мы установили значение таймера в 0 (`timer = 0`) и используем цикл для проверки любого пользовательского ввода с джойстика. При нажатии джойстика обнаруживается событие «нажатие клавиши».

```
if event.key == pygame.K_UP:
    print('Adding time')
    timer = timer + 5
    sense.show_message(str(timer), text_colour=[255,0,0])
```

Если событие возвращает, что джойстик потянули вверх, то к таймеру добавляется 5 секунд и об этом сообщается пользователю через светодиодную матрицу. Подобным же образом, если джойстик потянули вниз, то из таймера вычитается пять секунд.

```
elif event.key == pygame.K_LEFT:
    print('Add flash')
    sense.show_message('Flash ready', text_
    colour=[255,0,0], scroll_speed=0.05)
    toggle = 'on'
```

Здесь мы используем сдвиг джойстика влево, чтобы включить вспышку, сменив значение, хранящееся в переменной переключателя, на `'on'`. Сдвиг джойстика вправо выключит вспышку.

Наше финальное условие запускает начало съемки:

```
elif event.key == pygame.K_RETURN:
    print('Takepic')
    takepic(timer,toggle)
    flash('off')
```

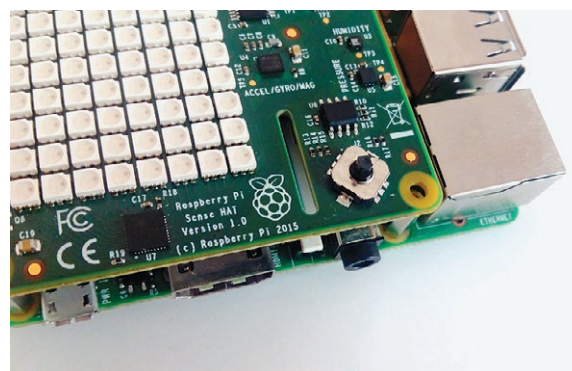
Нажатие на джойстик — это то же самое, что и нажатие на Enter (он же Return) — запускает функцию `takepic()` и передает значения таймера и переменные выключателя в качестве аргументов функции. Снимок сделан, и вспышка выключается.

Последний раздел кода — это часть `except` теста `try...except`.

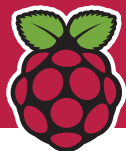
`except KeyboardInterrupt:`

```
pygame.display.quit()
pygame.quit()
```

Если пользователь нажмет `Ctrl+c`, проект закроет все открытые окна `pygame` и завершится. Создав весь этот код, сохраните свою работу и нажмите на `Run > Run Module`, чтобы приступить к съемке с AstroCam! **LXF**



► Sense Hat — замечательная платформа для проведения научных экспериментов, и благодаря надежной библиотеке Python интегрируется в массу всяких проектов.



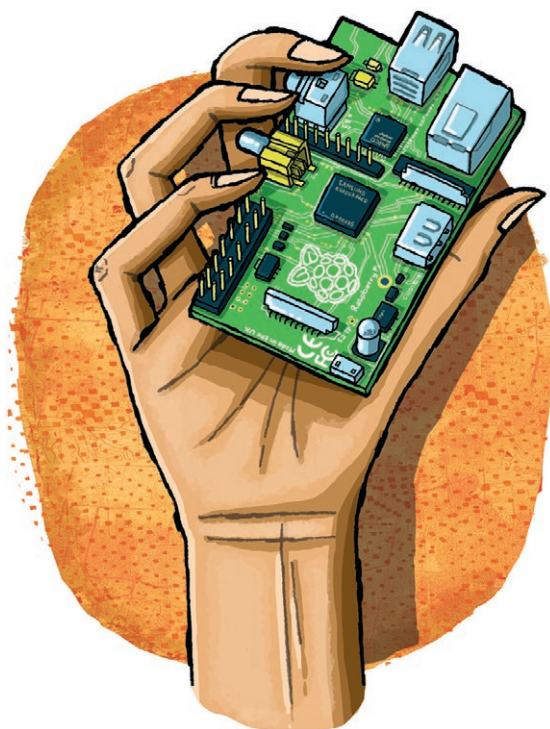
# RetroPi: Беремся за геймпад Xbox

Маянк Шарма возрождает золотую эпоху игр, присоединяя свой Raspberry Pi к винтажному игровому контроллеру.



Наш эксперт

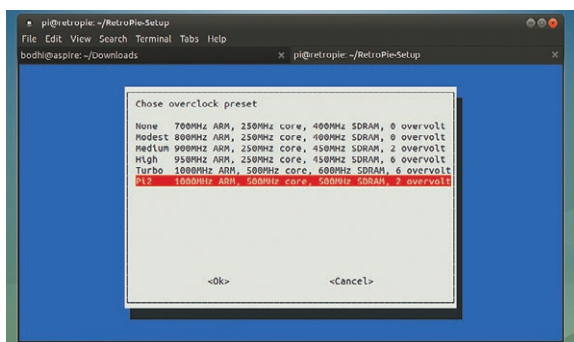
Маянк Шарма был пишущим редактором [www.linux.com](http://www.linux.com) и делал статьи для Linux Today, Digg и PC Plus.



Видеоигры в 1980-х сильно отличались от нынешнего набора игр «убивай-пока-не-убьют» — «наводи-и-пали». Это были со вкусом сделанные шедевры 8-битной графики, с напряженным сюжетом и игровым процессом, затягивавшим на долгие часы. Если, читая это, вы испытали ностальгию, можете эмулировать золотую эру игровых консолей на вашем современном оборудовании и совершить побег в прошлое.

Новому четырехъядерному Raspberry Pi 2 вполне по зубам виртуально воссоздать консольные видеоигры прошлого. Большинство программ, воссоздающих угасшие платформы, доступно

➤ Перед попыткой разогнать свой Pi на всякий случай попробуйте поиграть в игры на тактовой частоте по умолчанию.



в виде ПО с открытым кодом, которое можно установить поверх дистрибутива Raspbian. Однако простейшим способом начать играть в винтажные игры на Raspberry Pi является установка предназначенного для этих целей дистрибутива RetroPie, поставляемого с комплектом эмуляторов.

RetroPie можно установить вручную поверх существующего дистрибутива Raspbian, но целесообразнее использовать заранее подготовленный образ. Дистрибутив работает не только с Raspberry Pi 2, но и с более старыми моделями, так что выбирайте правильный образ. Вам нужно перенести этот образ на как минимум 4-ГБ карту, либо использовать команду `dd` в Linux, например,

```
$ dd if=retropie-rpi2.img of=/dev/sdd
```

Понадобятся также USB-клавиатура и мышь — для некоторой первоначальной настройки, которую нельзя выполнить удаленно через SSH. Мы также подключили к Raspberry Pi совместимый Wi-Fi-адаптер, который с ходу не заработает, но мы вернемся к этому позже. Главное во всем этом — нахватать игровых контроллеров, чтобы от души насладиться играми, а RetroPie умеет работать с разными контроллерами, от дешевых и безымянных до контроллеров PS3 и Xbox.

Подготовив карту памяти с образом RetroPie, вставьте ее в Raspberry Pi, подключите контроллер, Wi-Fi-адаптер, динамики и USB-устройства ввода, подключите все это к вашему HDMI-монитору и запустите. Pi загружается прямо в *Emulation Station* — это графический интерфейс, используемый для переключения между эмуляторами. Интерфейс попросит вас сконфигурировать контроллер. Однако перед этим несколько настроек надо изменить. Нажмите клавишу F4 на клавиатуре, чтобы выйти из *Emulation Station*, затем перейдите в *XTerminal*.

## Основная настройка

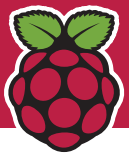
Первоочередная задача — расширить образ, чтобы охватить всю карту. Для этого выведите утилиту настройки Raspbian с помощью

```
$ sudo raspi-config
```

и выберите первую опцию, чтобы расширить файловую систему. После этого перейдите ко второй опции, чтобы изменить пароль по умолчанию для пользователя pi.

Затем перейдите в *Advanced Options* и выберите опцию SSH, она включит удаленный доступ. Чтобы обеспечить максимум памяти для игры, перейдите к опции *Memory Split*. Если у вас Raspberry Pi 2, выделите 512 GPU. Пользователи старой модели B+ должны выделить 256. И наконец, прокрутите вниз до опции *Overclock*, где пользователи Raspberry Pi 2 должны выбрать опцию Pi2. Внеся все изменения, вернитесь в главное меню и выберите *Finish*, чтобы перезапустить Raspberry Pi, сохранив изменения.

Перезапустившись, еще раз нажмите F4, чтобы выйти из *Emulation Station*. Теперь займемся Wi-Fi-адаптером — заставим его работать. Открывайте файл настройки —



## Обновите RetroPie

Скрипт RetroPie — это чудесный инструмент, способный превратить исходный дистрибутив Raspbian в потрясающую машину для аркад. А если вы уже запустили версию RetroPie, можете применить скрипт для обновления до более новой версии без скачивания и очередной переустановки всего дистрибутива.

Чтобы обновить свою установку, выйдите из *Emulation Station* и введите в командной строке

```
$ sudo apt-get update
```

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

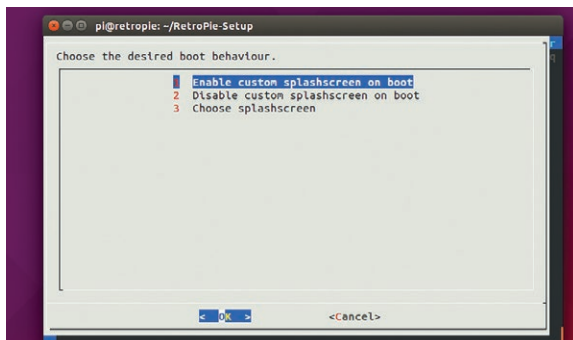
Эти команды обновляют репозитории дистрибутива, переустанавливая устаревшие пакеты. Теперь, когда основной дистрибутив обновлен, настало время обновить различные игровые эмуляторы. Опять же в командной строке, введите

```
$ cd RetroPie-Setup
```

```
$ sudo ./retropie_setup.sh
```

Поскольку вы находитесь внутри скрипта, первая задача — обновить сам скрипт RetroPie-Setup.

Вверху скрипта приводятся на выбор две разные опции обновления. Первая скачает предустановленные бинарники эмуляторов, а вторая скомпилирует их из источника. Первая опция гораздо быстрее, а вторая, хотя и до ужаса медленно работает на Raspberry Pi, скачивает новейшие версии эмуляторов. Вы можете спокойно проигнорировать вторую опцию и просто выбрать первую, которая скачивает и настраивает новые версии всех эмуляторов.



► При желании через скрипт RetroPie-Setup можно поменять заставку.

```
$ sudo nano /etc/network/interfaces
```

и измените его содержимое, чтобы оно стало похожим на такое:

```
auto lo
iface lo inet loopback
iface eth0 inet dhcp
allow-hotplug wlan0
auto wlan0
iface wlan0 inet dhcp
wpa-ssid "Имя вашей беспроводной сети"
wpa-psk "Пароль вашей беспроводной сети"
```

Обязательно замените текст в строке `wpa-ssid` на SSID и пароль для вашей сети Wi-Fi. Нажмите `Ctrl+x`, чтобы сохранить файл, и выйдите из текстового редактора. Теперь перезагрузите Pi с помощью `sudo reboot`. Когда Pi загрузится снова, Wi-Fi-адаптер подключит вас к вашему роутеру.

В очередной раз выйдите из *Emulation Station* и сделайте заметку об IP-адресе, присвоенном RetroPie вашим роутером. Предположим, это `192.168.3.111`; теперь вы можете войти в него с другого компьютера по команде `sudo ssh pi@192.168.3.111`.

Независимо от способа получения доступа к Pi, следующей задачей является изменение некоторых настроек RetroPie. Перейдите в директорию **RetroPie-Setup** с помощью

```
$ cd ~/RetroPie-Setup
```

и выполните скрипт настройки:

```
$ sudo ./retropie_setup.sh
```

Этот скрипт скачает все требуемые пакеты, которых нет в вашей системе, и отобразит меню на базе *Ncurses*, когда будет готов. Вначале прокрутите вниз до предпоследней опции, которая обновляет сам скрипт RetroPie-Setup. Затем перезапустите скрипт и прокрутите вниз до третьей опции, означенной как `Setup/Configuration`.

Здесь прокрутите вниз и выберите '323', чтобы внеслись необходимые изменения для отображения меню настроек RetroPie

в *Emulation Station*; это поможет модифицировать дистрибутив, не обращаясь к командной строке. Теперь, в зависимости от вашего звукового оборудования и способа его подключения к Raspberry Pi, может понадобиться подправить RetroPie вручную, прежде чем он сможет правильно отправлять звуковой вывод. Выберите опцию '301', чтобы настроить параметры аудио. Если автоматическая опция по умолчанию звука не воспроизводит, прокрутите вниз и выберите вывод, к которому подключены ваши динамики. В меню также предлагается опция вызова микшера для настройки громкости.

## Настройте контроллеры

Перезагрузите дистрибутив — в последний раз! — и продолжите работу с *Emulation Station*. Если ваш контроллер подключен, дистрибутив найдет его. Нажмите и придержите любую клавишу на контроллере, чтобы помочь дистрибутиву правильно распознать контроллер. Затем вас попросят обозначить клавиши на контроллере. Имейте в виду, что это первичное обозначение нужно только для навигации по графическому интерфейсу и помогает переключаться между системами эмуляции и выбирать игры. Когда вы настроите контроллер, вас перенесет в главное меню интерфейса *Emulation Station*. Теперь, чтобы настроить контроллер для игры, перейдите в меню RetroPie в *Emulation Station* и выберите опцию `Configure RetroArch Keyboard/Joystick`. С помощью клавиатуры выберите первую опцию под названием `Configure Joystick/Controller` и следуйте подсказкам на экране, чтобы настроить свой контроллер. Если у вашего контроллера нет клавиш, о которых упоминается в подсказках, просто подождите несколько секунд, и настройка перейдет к следующей клавише.

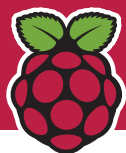
## Скорая помощь

Чтобы обозначить опцию выхода на игровом контроллере, отредактируйте `retroarch.cfg` — добавьте `input_enable_hotkey_btn = "X"` и `input_exit_emulator_btn = "Y"`. Замените X и Y на кнопки, обозначенные как `Start` и `Menu` в выбранном вами контроллере.



► *Emulation Station* отображает число игр в конкретном эмуляторе.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)



## Скорая помощь



При использовании нескольких контроллеров одновременно лучше всего брать одинаковые контроллеры, чтобы избежать проблем с настройками или геймплеем.

## Драйверы контроллера

Если вы используете контроллер Xbox 360 или PS3, надо сперва установить их драйвера, чтобы RetroPie мог их обнаружить. В более ранних версиях тут требовалась работа с командной строкой. Однако в более поздних версиях дистрибутива это дело очень простое и не вызывающее затруднений. Перейдите к опции RetroPie-Setup в меню RetroPie внутри *Emulation Station*. Это приведет вас к меню *Ncurses* скрипта RetroPie-Setup, где мы уже побывали. При помощи клавиатуры выберите третью опцию для настройки дистрибутива. Прокрутите список и выберите соответствующую опцию установки драйвера для вашего контроллера — '318', чтобы установить драйвер PS3, и '332', чтобы установить драйвер для Xbox 360.

Скрипт Xbox360 загружает драйвер `xboxdrv` и редактирует файл `/etc/rc.local`, чтобы драйвер запустился, а также добавляет данные для проводных контроллеров 360. Если же вы используете

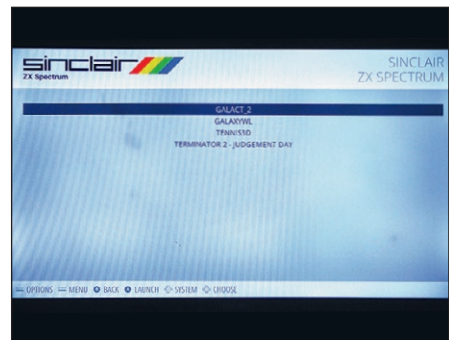
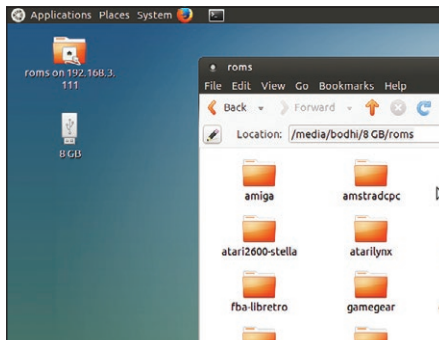
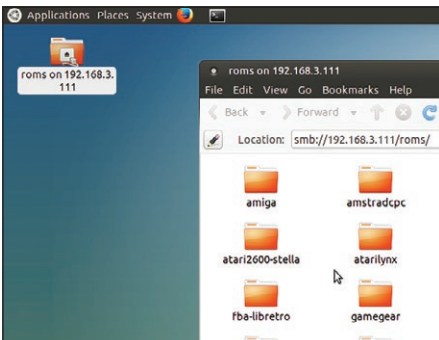
беспроводные контроллеры, откройте файл `/etc/rc.local` в текстовом редакторе, найдите строки, начинающиеся с `xboxdrv`, и замените опцию `--id` на `--wid`.

Если вы используете контроллеры PS3, после установки драйверов с помощью скрипта, как рассказано ранее, вам будет дана подсказка подключить адаптер Bluetooth для контроллеров. Даже после того, как вы сделаете это, RetroPie не сможет обнаружить ваши контроллеры. По мнению разработчиков, это вполне ожидаемо. Выйдите из скрипта и из *Emulation Station*. После того, как вы вернетесь в командную строку, переключитесь на директорию `/opt/retroPie/supplementary/ps3controller/` и наберите

```
$ sudo ./sixpair
```

Эта изящная маленькая утилита должна обнаружить адаптер Bluetooth и известить о нем RetroPie. Затем перезагрузите Raspberry Pi, и как только он загрузится, перейдите в директорию `/dev/input`

## Перенос ROM



### 1 Сетевой перенос

Если RetroPie подключен к роутеру, можно перенести в него игровые ROM (ПЗУ) с любого компьютера из той же сети. Дистрибутив поставляется с преднастроенным сервером *Samba* и ведет себя как разделяемые ресурсы Windows. Скопируйте ROM в директорию для соответствующего ей эмулятора.

### 2 Через USB

Простейший способ перенести ROM — использовать флешку. При обнаружении флешки RetroPie создает структуру каталогов для файлов ROM, отражающий эмуляторы, установленные в дистрибутиве. Подождите немного, пока он создаст директории, и затем извлеките флешку.

### 3 Plug and play

Теперь вставьте флешку в свой ПК и скопируйте ROM на нее, позаботившись поместить их в правильную папку. Когда вы снова подключите эту флешку к своему Pi, RetroPie автоматически поместит ROM в соответствующую директорию для связанного с ней эмулятора.

## Используйте виртуальный геймпад

Не стоит переживать, если у вас нет игрового контроллера — вместо этого можно создать и использовать виртуальный контроллер с вашего телефона или планшета.

Чтобы создать виртуальный геймпад, перейдите в *XTerminal* и установите требуемые компоненты — следующими командами:

```
$ sudo apt-get update
$ sudo apt-get upgrade
$ wget http://node-arm.herokuapp.com/node_latest_armhf.deb
$ sudo dpkg -i node_latest_armhf.deb
```

Теперь, когда у вас имеются нужные компоненты, переключитесь на пользователя `root` с помощью команды `su`. Вам будет предложено ввести пароль пользователя `root` (это `raspberrypi`). После аутентификации введите

```
# git clone https://github.com/miroof/
```

```
nodevirtual-Gamepads
```

```
# cd node-virtual-gamepads
```

```
# npm install
```

```
# npm install pm2 -g
```

Вышеуказанные шаги требуют некоторого времени на выполнение. Когда все закончится, можете запустить контроллер и позволить ему загружаться автоматически при загрузке системы:

```
# pm2 start main.js
```

```
# pm2 startup
```

```
# sudo pm2 save
```

Теперь возьмите свой телефон или планшет, откройте web-браузер (для лучших результатов разработчики рекомендуют *Google Chrome*) и введите в панели адреса IP-адрес Pi. Вы должны увидеть на странице виртуальный контроллер. Учтите, что надо настроить ваш контроллер с помощью RetroArch так же, как вы делали бы это с физическим

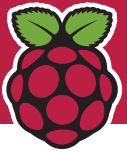
контроллером. Web-приложение игрового контроллера также обеспечивает тактильную отдачу — если это будет раздражать вас, можете просто убрать ее, выключив вибрацию на своем устройстве.



» К RetroPie можно подключить целых четыре виртуальных контроллера.

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





и просмотрите его содержимое с помощью `ls`. Если ваш контроллер обнаружен, он появится в списке как `js0`. Контроллер можно проверить командой

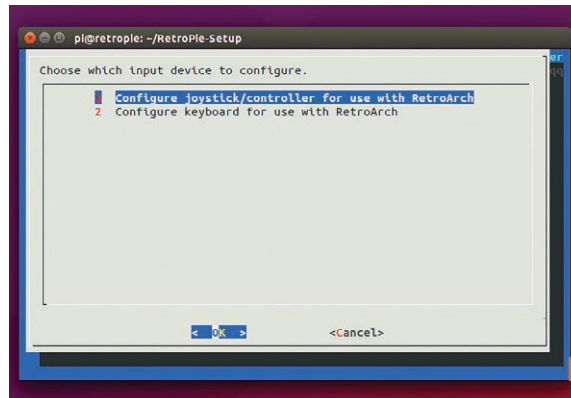
```
$ jstest /dev/input/js0
```

— она вызовет программу *jstest*, разработанную для проверки функций контроллера. Теперь вернитесь к меню RetroPie в *Emulation Station* и используйте опцию Configure RetroArch Keyboard/Joystick, чтобы настроить контроллер. И — ура! — ваш контроллер настроен и готов к использованию. Вы можете проделать это со всеми вашими контроллерами, и RetroPie сохранит настройки и будет автоматически их загружать, когда бы вы ни подключили контроллер.

Теперь можете прокручивать *Emulation Station* и играть в предустановленные игры со своими контроллерами. Когда это вам надоест, следуйте пошаговому руководству по переносу в RetroPie ваших личных игровых ROM.

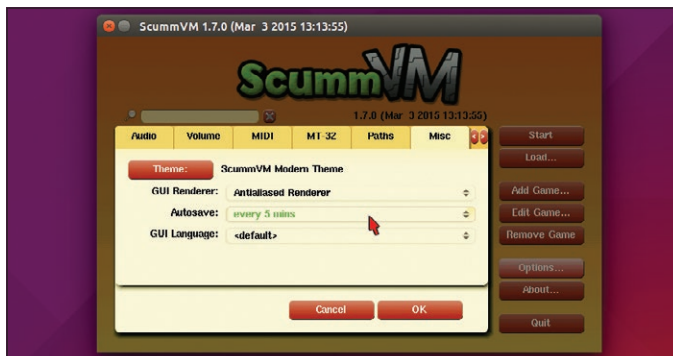
Есть несколько сайтов, таких как World of Spectrum ([www.worldofspectrum.org](http://www.worldofspectrum.org)), которые хранят доступные для легального и бесплатного скачивания ROM, пожертвованные или заброшенные разработчиками. Подлинные фанаты ретро-игр создают

свои собственные ROM из старых картриджей, что не столь уж и хитро проделывается благодаря существующим адаптерам наподобие Retrode. **LXF**



➤ Мультисистемный эмулятор *RetroArch* делает за дистрибутив черную работу.

## Играйте в игры со ScummVM



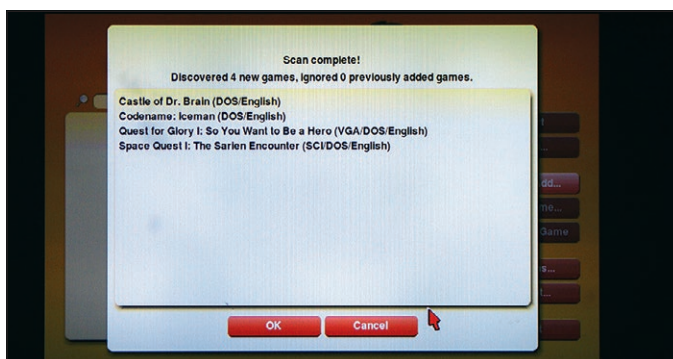
### 1 Настройте ScummVM

Откройте программу для запуска и щелкните по кнопке Options. Перейдите в последнюю вкладку, где размещаются дополнительные настройки. Воспользуйтесь кнопкой Theme, чтобы изменить внешний вид программы запуска, переключаясь на другие темы оформления. Настройки GUI Renderer определяют способ запуска, а опция Autosave задает в *ScummVM* интервал между сохранениями текущего состояния игры.



### 2 Пути по умолчанию

Переключитесь на вкладку Paths, чтобы настроить местоположение, где *ScummVM* будет искать определенные файлы. Опция Save Path указывает на папку по умолчанию, куда *ScummVM* будет заносить сохраненные игры. Если опция не установлена, игры сохраняются в текущей директории. Есть еще опция Theme Path, она указывает директорию, где будут храниться дополнительные темы оформления для программы запуска.



### 3 Добавьте игры

Чтобы загрузить в *ScummVM* поддерживаемую игру, скопируйте ее массивы данных из первоисточника. Если вы скачали файлы с сайта *ScummVM*, то перед копированием в RetroPie их понадобится распаковать. Затем запустите *ScummVM*, нажмите кнопку Mass Add и укажите *ScummVM* на извлеченную папку. Все игры в папке автоматически обнаружатся и появятся в списке игр.



### 4 Глобальное меню

Выберите игру, в которую хотите поиграть, и нажмите Start. Во время игры можно нажать комбинацию клавиш Ctrl+F5, чтобы сделать паузу и вывести глобальное меню. Оно дает возможность получить подсказку и повлиять на процесс. Кнопка Help дает доступ ко всей внутриигровой справочной документации, а кнопка Options позволяет менять некоторые настройки — например, громкость.

# ОТВЕТЫ

Есть вопрос по открытому ПО? Пишите нам по адресу [answers@linuxformat.ru](mailto:answers@linuxformat.ru), и мы найдем ответ.

В этом месяце мы ответим на вопросы про...

- 1 Слишком чувствительный тачпад
- 2 Общение с более старыми серверами SSH
- 3 Мониторинг устройств на Android
- 4 Проблемы с беспроводным подключением
- 5 Медленный запуск рабочего стола
- 6 Надежное уничтожение данных

## 1 Чувствительный тачпад

Вам кажется, что тачпад моего ноутбука не в меру чувствителен; когда я набираю текст, курсор иногда случайно перепрыгивает в другое место, и текст попадает совсем не туда, куда я хотел. Я не хочу отменять постукивания по тачпаду как щелчки, это очень неудобно; а нельзя ли понизить чувствительность или временно отключить тачпад во время набора? У меня Kubuntu 15.04.

Крис Коллинз [Chris Collins]

Если Ваш трекпад использует драйвер `synaptics` (а это относится к большинству из них), тогда можно воспользоваться вкладкой `Touchpad` [Тачпад] в разделе `Input Devices` [Устройства ввода] системных настроек KDE. Есть возможность отключить тачпад и во время набора, но результат получится неоднозначный. Проблема кроется в таймауте: если установить его слишком коротким (например, 250 мс — значение по умолчанию), тачпад будет «просыпаться» от малейшей паузы при наборе; если же установить его слишком длинным, придется долго дожидаться, чтобы проделать что-то на тачпаде после прекращения набора. Поэкспериментируйте с этим параметром и посмотрите, подойдет ли он Вам. Если нет, то имеется другой вариант — настроить горячие клавиши для включения и отключения тачпада вручную, и они будут работать прекрасно, если Вы не забудете ими воспользоваться.

Если у Вас другой рабочий стол или если Ваш трекпад не использует `synaptics`, то можно включить и отключить его вручную командой `xinput`. Сначала запустите `xinput` в терминале без аргументов, и она выведет информацию об устройствах ввода:

```
Virtual core pointer
> Virtual core XTEST pointer
> SynPS/2 Synaptics TouchPad
```

Получив имя трекпада, Вы сможете включить и отключить его с помощью `xinput`:

```
$ xinput --disable "SynPS/2 Synaptics TouchPad"
```

```
$ xinput --enable "SynPS/2 Synaptics TouchPad"
```

Назначив для каждой из этих команд комбинацию клавиш в настройках рабочего стола (`System Settings > Shortcuts > Custom Shortcuts in KDE 5` [Параметры системы > Горячие клавиши > Пользовательские горячие клавиши в KDE 5]), Вы сможете включать и выключать тачпад по желанию. Если Вы хотите делать это одной клавишей, сохраните где-нибудь следующий скрипт, заменив `TP` на имя тачпада, которое Вы получили с помощью команды `xinput` выше, сделайте скрипт исполняемым и привяжите к сочетанию клавиш (сначала проверьте скрипт в терминале).

```
#!/bin/sh
```

```
TP="SynPS/2 Synaptics TouchPad"
```

```
if xinput --list "$TP" | grep -q 'This device is disabled'; then
```

```
  xinput --enable "$TP"
```

```
else
```

```
  xinput --disable "$TP"
```

```
fi
```

Этот несложный скрипт проверяет вывод `xinput --list`, чтобы узнать, отключен ли тачпад. Если да, скрипт включает тачпад, а в противном случае — отключает. Скрипт должен работать в любом дистрибутиве, любом окружении рабочего стола и для любого тачпада — или любого другого входного устройства, если Вы по какой-то причине захотите отключить клавиатуру или мышь (хотя в этом случае Вы не сможете повторно включить их с помощью горячих клавиш!).

## 2 Проблемы с SSH

Вы пытаетесь подключиться к удаленному серверу по SSH с помощью ключей, а не паролей, но при каждой попытке подключения получаю следующее сообщение:

```
Unable to negotiate with 82.917.814.129: no matching host key type found. Their offer: ssh-dss
```

```
Невозможен диалог с хостом 82.917.814.129:
```

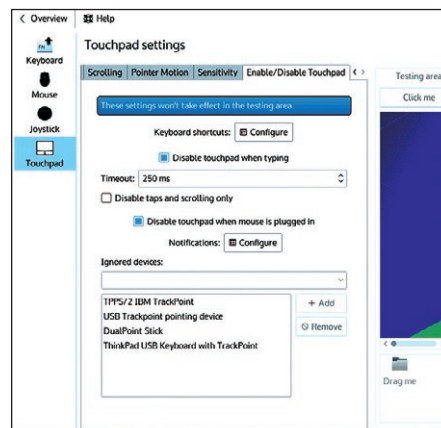
```
не найден подходящий тип ключа. Предлагается: ssh-dss
```

Раньше я подключался к ним нормально; что же изменилось? Дело во мне или в них?

Я не управляю сервером; я просто пользователь и ничего у себя не менял.

Брайан Эдвардс [Bryan Edwards]

Вы утверждаете, что ничего не меняли, но я подозреваю, что заодно с обычными обновлениями своей системы Вы обновили и `OpenSSH`. В `OpenSSH 7.0` изменился способ работы с некоторыми устаревающими и более слабыми криптографическими протоколами. Они все еще поддерживаются `OpenSSH`, но по умолчанию



Настройте отключение тачпада (автоматически или вручную) в системных настройках KDE.

отключены. В Вашем случае Вы пытаетесь подключиться к серверу, который поддерживает только алгоритм `ssh-dss` (DSA) с открытым ключом, а его разработчики `OpenSSH` считают слабым.

Решение состоит в том, чтобы снова включить алгоритм `ssh-dss`, поскольку алгоритм слабый, делать это только для таких сайтов, которым это абсолютно необходимо. Если у Вас уже есть запись хоста для этого сайта в файлах `/etc/ssh/ssh_config` или `~/.ssh/config`, припишите к ней следующую директиву:

```
HostKeyAlgorithms +ssh-dss.
```

Если ни в одном из этих файлов записи хоста нет, добавьте ее следующим образом:

```
Host 82.917.814.129
```

```
HostKeyAlgorithms +ssh-dss
```

Если для подключения обычно используется имя хоста, а не IP-адрес, это имя следует поместить в строку `Host`. Если в файле есть раздел, начинающийся с `Host *`, надо поместить свою запись выше него. Это связано с тем, что `SSH` использует первую подвернувшуюся подходящую запись, и если раздел `Host *` встретится раньше Вашего, то Ваши изменения никакой роли не сыграют. Если это одноразовое подключение и Вы не хотите добавлять хост к своему конфигурационному файлу, можно также включить ключевой алгоритм с командной строки для одного вызова `SSH`:

```
ssh -oHostKeyAlgorithms+=ssh-dss user@hostname
```

Плюс перед именем алгоритма означает, что он добавляется к списку доступных алгоритмов. (Если плюс не указать, то весь список заменится на указанный Вами алгоритм.) Это очень важно, потому что когда на сайте окажется доступным более надежный метод, `OpenSSH` автоматически применит его без какого-либо участия с Вашей стороны. Не указав `+`, Вы будете прикованы к более слабому

## 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`.

методу навсегда. Существуют похожие ограничения на алгоритмы обмена ключами, которые обрабатываются аналогичным образом при средствах параметра `KexAlgorithms`.

### 3 Общаемся с Android

В Я пытаюсь скопировать файлы на новые устройства с Android. Они убрали возможность монтирования хранилища телефона как USB-устройства, и вместо этого приходится пользоваться MTP. Я пробовал MTP, но это работает медленно и ненадежно. Поэтому мне интересно, нет ли удобного способа скопировать

файлы через Wi-Fi? «Удобный» для меня означает отсутствие трудностей и заклинаний, необходимых для Samba и ее аналогов.

Фил Уилер (Phil Wheeler)

Я разделяю Ваше недовольство MTP: иногда он работает вполне нормально, а иногда вообще перестает работать в самый неподходящий момент. Монтирование как USB-накопитель изначально было плохой идеей, потому что для подключения устройства его приходилось размонтировать, и программы переставали работать. Две системы не могут обращаться к одной и той же смонтированной файловой системе без риска что-нибудь испортить.

При копировании данных по беспроводному подключению имеется несколько вариантов. Если Вам требуется сохранять каталоги на рабочем столе и на телефоне (например, коллекции музыки или фотографий) синхронизированными, хороший вариант — *Syncthing* (<https://syncthing.net>) [см. Учебники, стр. 72 LXF203]. Для копирования файлов «вручную» чрезвычайно удобна SSH, особенно если она уже настроена на Вашем компьютере. Некоторые файловые менеджеры Android поддерживают подключения по SSH или SFTP (SFTP отвечает за передачу файлов в SSH). Я пользуюсь ES File Explorer и просто передаю ему имя компьютера с именем пользователя и паролем, и он отображает содержимое жесткого диска на мобильном устройстве. Разумеется, тогда на Вашем компьютере должен быть установлен сервер SSH, а во многих дистрибутивах установлен только клиент, поэтому установите пакет с *openssh-server* (или с похожим именем).

Другой вариант — поступить наоборот и установить из Play Store *SSH Helper*. Это сервер SSH, который поддерживает команды `scp`, `sftp` и `rsync`. Он работает на порту 2222, а не на более стандартном порту 22, потому что для открытия портов с номерами меньше 1024 необходимы права `root`. Это означает, что нужно указать параметр `-p` с `ssh` и `-P`

с `scp` или `sftp`. Теперь Вы можете копировать файлы на устройство и с него командами `scp` или `rsync` или просматривать содержимое устройства в большинстве графических файловых менеджеров, открыв в них адрес `sftp://IP-ADDRESS:2222`.

Пароль можно задать в окне настроек *SSH Helper* (после чего надо перезапустить сервер), но имя пользователя не имеет значения, поэтому оставьте его незаполненным, и SSH по умолчанию будет использовать Ваше текущее имя пользователя. *SSH Helper* припас еще одного туза в рукаве — он может смонтировать Ваше устройство утилитой *sshfs*. Сначала может потребоваться установить ее на свой компьютер, затем выполните команду

```
$ sshfs -p 2222 IP-ADDRESS:/storage/emulated/0 ~/android
```

Может потребоваться изменить путь ко вложенному хранилищу. Приведенный выше пример работает для последних устройств Nexus, но нужно убедиться, что используемая точка монтирования существует, и она пуста. Теперь Ваше устройство смонтировано, как бывало с USB-устройствами в прежние времена. Когда закончите, размонтируйте его одной из следующих команд:

```
$ sudo umount ~/android
$ fusermount -u ~/android
```

### 4 Шаткий Wi-Fi

В Я пытаюсь (безуспешно) подключиться к незащищенной беспроводной сети с помощью *wpa\_supplicant* в Ubuntu 15.04 на ноутбуке Dell. Я также попробовал подключиться к сети с шифрованием WEP, и все работает. Вот содержимое моего файла *wpa\_supplicant.conf*:

```
ctrl_interface=/var/run/wpa_supplicant
network={
    ssid="XXXX"
    key_mgmt=NONE
}
network={
    ssid="YYYY"
    key_mgmt=WEP
```



## Коротко про...

# Точки монтирования

Буквы дисков и точки монтирования могут запутать, особенно тех, кто пришел из Windows и ожидает увидеть нечто вроде известной нотации C:, D:, E:. В Linux все доступно в корневой файловой системе `/`. При монтировании устройства вы указываете как монтируемое устройство, так и место (точку) его монтирования. После этого содержимое устройства становится видимым в данной точке монтирования. В точке монтирования нет ничего магического — это обычный каталог. Самый распространенный пример — использование отдельного раздела для каталога `home`. В данном случае каталог `home` в корневом разделе

является пустым, в чем вы сможете убедиться, загрузившись с Live CD и смонтировав корневой раздел. Когда в `/home` монтируется другой раздел, в этом каталоге отображается его содержимое. Например, если раздел `home` содержит каталоги для пользователей `alice` и `bob`, они появятся как `/home/alice` и `/home/bob` соответственно.

Права на точки монтирования могут вызывать неясности. Права каталога, используемого для точки монтирования, не имеют значения — каталог принимает права доступа и владельцев всех файлов и каталогов, которые в нем монтируются, поэтому пытаться изменять права

и владельца для неиспользуемой точки монтирования — пустое занятие. Если вы монтируете такую файловую систему, у которой нет прав доступа Linux, например, флешку, отформатированную в FAT, точке монтирования назначаются фиктивные права доступа, которые позволяют вам считывать и записывать данные с устройства.

Точки монтирования способствуют огромной гибкости; например, если у вас есть большой диск с фильмами, его можно смонтировать в каталог `~/Videos`, чтобы казалось, что он находится в вашем домашнем каталоге, тогда как на самом деле он на совершенно другом диске.

```
wep_key0 = "abcde"
```

При попытках подключиться к открытой сети лог-файл заполняется записями вида

```
new0: CTRL-EVENT-SCAN-STARTED
```

```
new0: SME: Trying to authenticate with
aa:bb:cc:dd:ee:ff (SSID='XXXX' freq=2432 MHz)
```

```
new0: SME: Authentication request to the driver
failed
```

```
[Неудачный запрос к драйверу
на аутентификацию.]
```

Генри Александер (Henry Alexander)

Похоже, что проблема кроется в драйвере, но Вы не написали, какой беспроводной картой или драйвером пользуетесь. Попробуйте добавить `auth_alg=OPEN` после строки `key_mgmt`. Этот шаг не является строго обязательным, но некоторым драйверам нужно оказывать всю необходимую помощь. Если это не поможет, я бы взглянул на драйвер, которым Вы пользуетесь, особенно если у Вас карта Broadcom, которая встречается во многих ноутбуках Dell.

Карты Broadcom обычно используют драйвер ядра `b43`, который работает корректно не со всеми чипсетами Broadcom 43XX. Решение — использовать собственный драйвер Broadcom, установив пакет `broadcom-stacommon`. После установки надо отключить драйвер `b43`, добавив следующие строки в файл `/etc/modprobe.d/blacklist.conf` — если файла не существует, создайте его:

```
blacklist ssb
blacklist bcma
blacklist b43
blacklist brcm-smac
```

Затем либо удалите старые модули командой `rmmod` и загрузите новый модуль, набрав команду `modprobe` (модуль называется `wl`, независимо от имени пакета), либо просто перезагрузите компьютер, чтобы изменения вступили в силу.

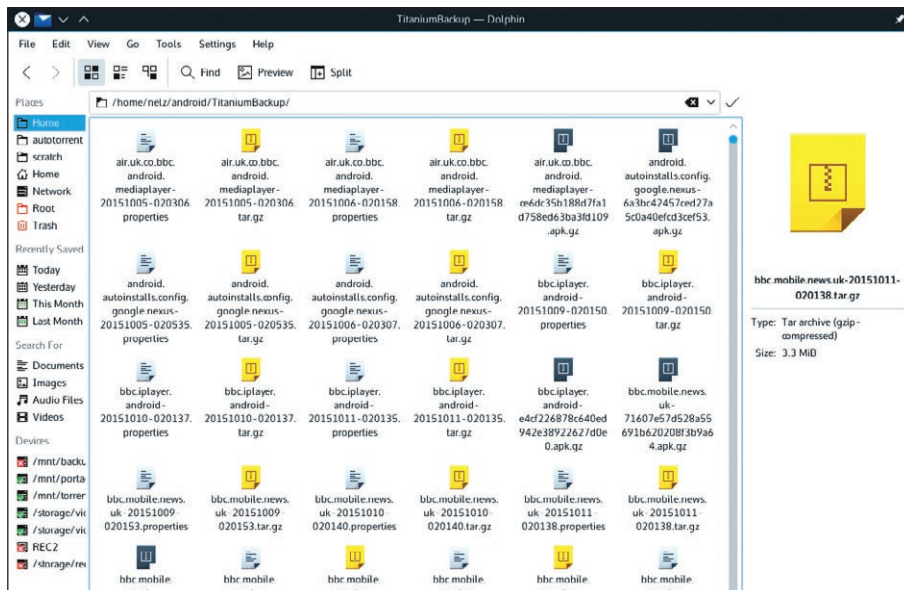
Если у Вас карта не от Broadcom (судя по всему, именно они вызывают такие проблемы), Вы можете получить дополнительную информацию по `dmesg` или из системного лог-файла. Так как интерфейс, который Вы пытаетесь настроить, называется `new0` (Вы пытаетесь сделать что-то хитрое с виртуальными интерфейсами?), то перед попыткой подключения стоит попробовать запустить такую команду и затем наблюдать за ее выводом:

```
$ journalctl --follow | grep new0
```

## 5 Медленный запуск

У меня в Ubuntu 14.04 конфигурация с двумя дисками: твердотельный диск для / и жесткий диск объемом 1 ТБ для /home. Загрузка происходит быстро, но первые несколько минут после нее компьютер работает очень медленно. Каждый раз, когда я запускаю панель Unity и набираю в поиске название приложения, на появление результатов может потребоваться минута. В следующий раз это работает более-менее быстро. Nautilus при первом запуске также загружается медленно и медленно отображает файлы.

Артур Хэйл (Arthur Hale)



## Просмотр содержимого телефона Android на настольном ПК с помощью SSH Helper.

Похоже, проблема с жестким (а не с твердотельным) диском — если он почти полон, то может быть фрагментированным. Система быстро загружается с SSD и быстро загружает рабочий стол с SSD. Потом ей приходится загружать файлы с жесткого диска, и все начинает работать медленно. После первого запуска необходимая информация кэшируется, и производительность восстанавливается. Основной виновник ситуации — по всей вероятности, панель Unity, и если она обращается к жесткому диску, то это повлияет и на другие утилиты, которым нужен диск: например, *Nautilus*, сканирующий файлы. В этот момент Вы должны увидеть большую активность жесткого диска и, возможно, даже услышите, как неистово дергают головки шаговые двигатели.

Простое решение — уменьшить объем поисковой информации, который собирает панель. Перейдите в раздел Security & Privacy [Безопасность и приватность] системных настроек [System Settings] и на вкладке Search [Поиск] снимите флажок Include online search results [Включить поисковые результаты из Интернета]. Затем перейдите на вкладку Files & Applications [Файлы и приложения], нажмите Clear Usage Data [Удалить данные об использовании] и затем снимите флажок Record file and application [Запись данных о файлах и приложениях].

Использование твердотельного диска для системы обуславливает быструю загрузку программ, но при обращении ко всем своим данным Вы вернетесь к скоростям жесткого диска; и у программ, которые кэшируют настройки и данные в нескольких файлах, Вы увидите резкое падение производительности. Производительность можно было бы значительно улучшить, поместив на твердотельный диск также и /home, но тогда пространство обычного жесткого диска останется неиспользованным. Одно из решений — смонтировать жесткий диск в домашний каталог (см. врезку «Точки монтирования», стр. 97) и сохранить большие файлы именно в этом каталоге. Вполне достаточно

будет переместить в этот каталог фильмы, фотографии и музыку, если Вы хотите, чтобы они оставались на старом месте. Например, если смонтировать жесткий диск в `~/bigstuff`, можно переместить в него каталог `Pictures` и затем выполнить команду `$ ln -s bigstuff/Pictures Pictures` после которой Ваши фотографии появятся в `~/Pictures`. На такую настройку системы требуется чуть больше усилий, но различия в скорости может быть существенным. Например, почтовые программы с кэшами на SSD работают гораздо быстрее.

## 6 Уничтожение данных

Мой жесткий диск показывает ошибки SMART. Он все еще на гарантии, и это означает, что я могу его вернуть, но что делать с данными, которые на нем хранятся? Я совершенно не хочу, чтобы моя личная информация стала доступна случайным людям. Достаточно ли удалить мой домашний каталог? Я слышал, что удаленные файлы все равно можно прочитать. Джордж Боуэн (George Bowen)

Вы правы — при удалении файлов они на самом деле не удаляются. Удаляются только записи о них из таблицы каталога, при этом содержимое файлов остается именно там, где они были созданы. Перезаписать файлы недостаточно, так как во многих файловых системах при перезаписи файла сначала записывается новая копия файла, а затем просто обновляется таблица каталогов — с тем, чтобы она указывала на новую версию; данные же остаются на диске. Это позволяет избежать негативных последствий при отключении питания и сбоях — тогда, по крайней мере, старая версия у Вас остается; но в Вашей ситуации от этого мало прока. Проще всего будет удалить все, что Вам не нужно, а затем перезаписать всё свободное пространство файловой системы нулями с помощью следующих команд:

```
$ sudo rm -fr /home/*
$ sudo dd if=/dev/zero of=/home/nothing bs=4k
```

## Помогите нам помочь вам

Ежемесячно к нам поступает несколько писем, на которые мы не в состоянии ответить, поскольку проблема описана в них с недостаточной полнотой. Чтобы дать вам наилучший ответ, нам необходимо получить как можно больше информации о проблеме.

Если у вас появляется сообщение об ошибке, приведите его текст в точности и опишите конкретные условия, когда оно появляется. При возникновении проблемы с устройствами перечислите нам все устройства, которые у вас установлены.

Если Linux в вашей системе запущен, вы сможете применить для этого превосходную 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 — *Ishw* (<http://ezix.org/project/wiki/HardwareLiSter>). Одна из указанных программ непременно должна быть включена в ваш дистрибутив (а иногда и обе).

Если у вас нет желания или возможности их установить, выполните в терминале от имени root приведенные ниже команды и приложите сгенерированный ими файл `system.txt` к письму. Это окажет неоценимую помощь в диагностике вашей проблемы.

```
uname -a > system.txt
lspci >> system.txt
lspci -vv >> system.txt
```

Желая надежно удалить данные, воспользуйтесь DBAN, но не перепутайте диск!

Эта команда примется записывать нули в файл `home/nothing` — до тех пор, пока файловая система не заполнится; и перезапишет все пространство, используемое другими данными.

Нередко ведутся дискуссии о том, достаточно этого или нет. Рассказываются всяческие истории о сложных устройствах и способах, с помощью которых можно восстановить данные даже после полного обнуления. Однако на самом деле сделать это стало гораздо сложнее, чем раньше, поскольку данные на дисках размещаются всё с большей плотностью; но если Вы всё равно беспокоитесь, замените `/dev/zero` на `/dev/urandom` в приведенной выше команде, чтобы заполнить диск псевдослучайными данными. Не используйте `/dev/random`, это заставит Вас принять участие в создании энтропии для случайных данных, и Вы устанете двигать мышью, заполняя данными диск приличного размера. Затем удалите созданный файл и повторите этот процесс еще несколько раз.

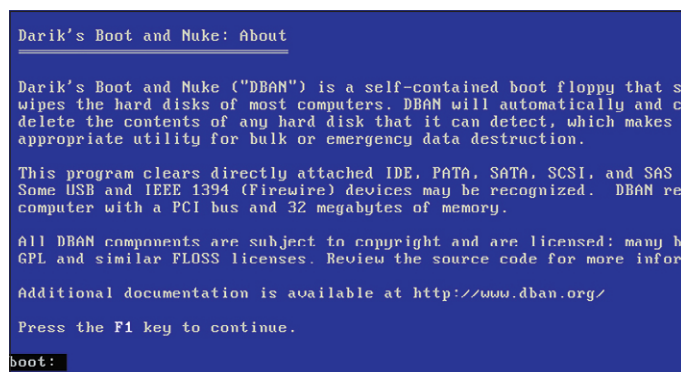
Тогда будут уничтожены все данные из раздела `home`, но как насчет файлов паролей в таких каталогах, как `/etc`? Для полной надежности сделайте

копию всех важных данных, которые включают такие каталоги, как `/etc`, наряду с `home` (если у Вас есть место, сделайте полный образ диска программой *Clonezilla*). После этого можно очистить весь диск командой

```
$ sudo dd if=/dev/zero of=/dev/sdX bs=4k
```

где `/dev/sdX` — это Ваш диск. Предварительно проверьте правильность команды, так как `dd` не спросит Вас ни о чем; после нажатия клавиши Enter обратной дороги уже не будет.

Если это системный диск, запустите команду с live CD и используйте `/dev/urandom` как указано выше, если Вам требуется дополнительное спокойствие. Помните, что Вы не получите свой диск обратно — скорее всего, его обменяют; но если его отремонтируют, то смогут отослать на замену неисправного диска кому-нибудь другому, поэтому будьте осторожнее с Вашими данными. Если Вам нужно полное спокойствие, всегда есть Live CD Darik's Boot and Nuke (DBAN) с сайта [www.dban.org](http://www.dban.org), который сделает для уничтожения данных всё возможное, кроме разве что проделывания дырок в дисках. **LXF**



## Коротко про...

# Использование дискового пространства

Каким бы большим ни был ваш жесткий диск, с течением времени он всё равно заполнится. Хотя с появлением дисков большой емкости это менее актуально, чем раньше, размеры медиа-файлов становятся все больше и больше.

Также пространство на диске занимают файлы, которые вам больше не нужны, временные файлы и файлы кэша, сброшенные в каталог `home` различными программами, а также, например, файлы пакетов, оставшиеся от обновлений ПО. Хотите ли вы восстанавливать это свободное место или нет, зависит от того, чего у вас больше — свободного времени или свободного

места, но сначала надо обнаружить «пожирателей пространства».

Существует несколько программ, которые покажут используемое место для каждого каталога. Пользователи KDE могут установить *Filelight*, а у пользователей Gnome, скорее всего, уже установлен *Disk Analyzer*. В терминале есть почтенная `du` или более новая `ncdu`. Вывод `du` нужно перенаправить через `sort`, чтобы увидеть тех, кто поглощает больше всего пространства:

```
$ du -sh * | sort -h
```

Примечание: эта команда не будет учитывать скрытые каталоги, так как большинство оболочек не сопоставляют их с `*`. Чтобы увидеть всё, воспользуйтесь командой

```
$ du -sh * .* | sort -h
```

Одна из вещей обо всех этих программах, которые нужно знать, состоит в том, что они также учитывают файлы из различных файловых систем, смонтированных в каталогах, которые вы проверяете. Даже если вы указали с `du` параметр `-x`, команда все равно зайдет во все файловые системы, точка монтирования которых покрывается `*`.

Простейшее решение — воспользоваться `ncdu` (ее, скорее всего, придется установить).

Запустите эту утилиту в формате `ncdu -x`, чтобы увидеть список каталогов в текущем каталоге, отсортированный по размеру, но не содержащий файлов

из других файловых систем. Для просмотра другого каталога укажите путь к нему в командной строке.

С `ncdu` можно не только вывести размеры каталогов. Выделите каталог и нажмите Enter или стрелку вправо, чтобы перейти в него, и вы увидите отсортированный список его содержимого; чтобы подняться на уровень выше, нажмите стрелку влево. Имеются и другие параметры; нажмите `?`, чтобы увидеть их список. Одним из полезных дополнений является возможность удалять и искать файлы, занимающие слишком много места. Как всегда при удалении файлов, соблюдайте осторожность.



## LXF HotPicks



**Александр Толстой**

предлагает горячий и острый соус (свободный) на гарнир к тщательно подобранным открытым приложениям, вам на радость.

SweetHome3D » Liri » PulseAudio » eXeLearning » Bomi » QMMP » SANE  
» Solar Wolf » TORCS » FLIF » Blender

Планировщик домашнего интерьера

# SweetHome3D

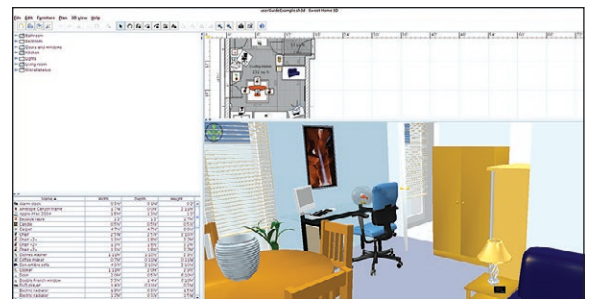
Версия: 5.1 Сайт: [www.sweethome3d.com](http://www.sweethome3d.com)

Тем, кто продумывает, как обставить свой дом или квартиру, прежде чем отправиться за покупками, пригодится отличная утилита с открытым кодом *SweetHome3D*. Аналоги существуют — например, инструмент планирования IKEA; но кода его не открыт, и он ограничен исключительно ассортиментом IKEA, чего и следовало ожидать.

Помимо этого, набор функций *SweetHome3D* гораздо богаче и включает расширяемый каталог мебели и библиотеку текстур плюс поддержку нескольких уровней плана, а главное — редактируемость параметров каждого пункта, включая размеры и пропорции. Можно также создавать свои

собственные изделия с нуля или импортировать их из разных поддерживаемых форматов 3D, таких как OBJ, DAE, KMZ и 3DS.

Визуальная раскладка *SweetHome3D* содержит слева дерево категорий мебели; его вы можете расширять и перетаскивать отсюда требуемые предметы в основную часть окна, где и рисуется 2D-план. Список поддерживаемых предметов и их габариты размещены в нижнем левом углу.



» Простой способ спланировать расстановку мебели, чтобы не спотыкаться о торчащее напольное покрытие.

## SweetHome3D имеет впечатляющий набор инструментов.

Чтобы настроить расширенные опции объекта, дважды щелкните по нему. В нижней части экрана, прямо под планом, размещен вид 3D, соответствующий плану. Вид можно перетаскивать и даже употреблять для этого клавиши **AWSD**, как вы бы поступили в стрелялке от первого лица. Набор инструментов для индивидуальной настройки плана в *SweetHome3D* впечатляет: можно строить стены, комнаты, уровни, поиграть с фоном и текстурами и смоделировать идеальное жилое пространство любого стиля и размера. Приложение весьма продуманное; оно написано на Java (GUI буквально кричит об этом) и выглядит одинаково на всех платформах. Онлайн-документация *SweetHome3D* очень хороша, но, вероятно, наилучшим стартом будет скачать разные образцы планов из официальной галереи. Это простейший способ овладеть основами создания собственного интерьера.

*SweetHome3D* может быть доступен в некоторых дистрибутивах Linux через менеджеры пакетов, но самый свежий релиз очень легко скачать с сайта проекта. *SweetHome3D* поставляется с собственным экземпляром Java Runtime Environment для обеспечения совместимости и стабильности, и должен хорошо работать без всяких дополнительных зависимостей.

## Исследуем интерфейс SweetHome3D

### Магазин мебели

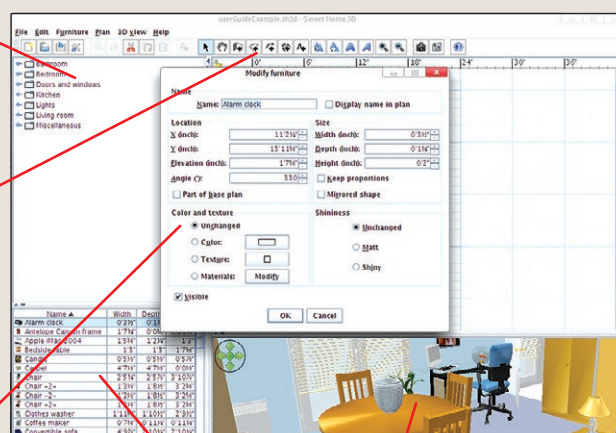
Расширьте категории и наслаждайтесь огромным списком мебели и прочих прелестей для дома.

### Панель инструментов планирования

Стены, комнаты, ломаные линии и подписи добавляются через эту удобную панель инструментов.

### Свойства объектов

Дважды щелкните по объекту из предыдущего списка, чтобы открыть его *Advanced Properties* [Расширенные свойства].



### Сейчас на плане

Все, что вы добавляете к плану дома, появляется здесь со своими текущими размерами.

### Вид 3D

3D-отображение любого вашего интерьера обновляется автоматически.

## Web-браузер

## Liri

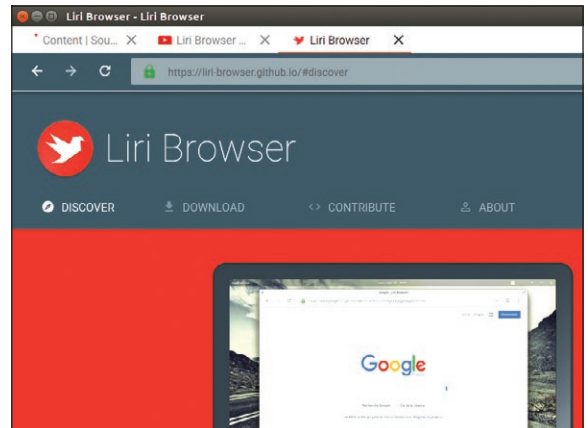
Версия: 0.3 Сайт: <http://bit.ly/LiriBrowser>

**У** web-браузеров на базе *Chrome* — прибавление семейства. Однако нужен ли нам еще один клон супер-успешного продукта Google? А вот посмотрим. *Liri* — студенческий проект нескольких молодых разработчиков из Германии, Франции и США. Идея, вдохновившая *Liri*, заключалась в создании быстрого, минималистского браузера с материальным исполнением UI и мощью проекта *Chromium* под капотом. Предельной точности ради — *Liri* использует самый свежий код *QtWeb-Engine*, основанный на движке *Chromium*. GUI создан с помощью *QML*, технологии, применяемой рабочим столом *Plasma*, и *Liri* отлично совместим с *Plasma 5* как в плане используемого графического набора инструментов (*Qt5* и *QML*), так и по внешнему виду и по ощущениям, поскольку соответствует теме *Breeze*.

В остальном *Liri* является упрощенной версией *Chromium 40*, из которого удалили львиную долю «избыточных» функций. Менеджер загрузок, закладки, вкладки

и прочие основные функции по-прежнему сохраняются, но это пока и все. Команда *Liri* продвигает свой просмотрщик исходного кода, работающий удаленно, как облачный сервис. При выборе пункта *View source* в главном меню *Liri* браузер показывает код, используя сервис API на <http://bit.ly/LiriSourceCodeViewer> с очень удобной раскраской и форматированием. Код этого просмотрщика на базе *Web* тоже открыт и постоянно улучшается.

Будучи сильно урезанным *Chromium*, *Liri* относительно мало нагружает память: всего 85 МБ после холодного запуска. Браузер пока что на ранней стадии разработки, и в нем отсутствует ряд важных функций, например, сохранение паролей, поддержка плагинов и переводы (желающие помочь



► Цветовая тема *Liri* автоматически адаптируется к сайту, на который вы зашли.

будут весьма кстати), но его стоит попробовать ради его поразительной скорости и простоты. Лучше всего скачать *Liri* с его официальной страницы *Download*. Браузер кросс-платформенный и для Linux предлагает репозиторий *Debian*, совместимые с *Ubuntu Deb*-пакеты и универсальную портируемую сборку в *ZIP*-формате.

**Liri использует свежий код QtWebEngine, на движке Chromium.**

## Аудиосервер

## PulseAudio

Версия: 7.0 Сайт: <http://bit.ly/PulseAudio7>

**Д**олгие годы *PulseAudio* был для многих консервативных пользователей проектом спорным, но он все больше обогащается функциями, становясь лучшим выбором, чем классический *ALSA*. Сейчас, спустя 11 лет разработки, *PulseAudio* достиг версии 7.0, и может похвастаться новыми замечательными функциями. Самая смелая из них — синтез канала *LFE* с низкочастотной фильтрацией. *LFE* означает *Low-Frequency Effects* [Низкочастотные эффекты] (термин *Dolby*), и сейчас они включены в *PulseAudio* по умолчанию. А значит, если у вас есть отдельный сабвуфер и вы намерены воспроизвести простой стереофайл всего с двумя каналами, сервер *PulseAudio* отфильтрует для него высокие частоты. Раньше *PulseAudio* подавал на сабвуфер усредненную информацию со всех каналов независимо от частоты.

Еще одна немаловажная вещь — поддержка распознавания соединительных гнезд при использовании *UCM* (*Use Case*

*Manager*) в *ALSA*. Когда подключается соединитель для одного устройства, *PulseAudio* теперь лучше понимает, надо ли ему заглушить другое устройство, чтобы не было наложения звука.

Есть и другие расширения, например, улучшенные устройства передискретизации для высококачественного аудиоконтента, и *PulseAudio* также отлично справляется с воспроизведением идеального звука с разрядностью до 32 бит; а тем, кто умеет различать разные психоакустические модели в слепом тесте, это определенно важно. Для остальной части человечества *PulseAudio* остается умной технологией, способной перенаправлять звук на разные приемники (в т. ч. удаленные), управляя

**PulseAudio отфильтрует высокие частоты для сабвуфера.**

```
atolstoy@hp-laptop: /etc/pulse
; dl-search-path = (depends on architecture)

; load-default-script-file = yes
; default-script-file = /etc/pulse/default.pa

; log-target = auto
; log-level = notice
; log-meta = no
; log-time = no
; log-backtrace = 0

; resample-method = speex-float-1
; enable-remixing = yes
```

► В *daemon.conf* масса возможностей настройки.

устройствами ввода и даже улучшать качество звука, устраняя эхо и прочие дефекты. Мы уже рассказывали о некоторых крутых командах для *Pulseaudio 6.0* [см. *HotPicks*, стр. 103 *LXF197*] — они применимы и в этой последней версии.

Обновление *PulseAudio 7.0* планируется в большинстве дистрибутивов Linux на ближайшее будущее, и если его нет в вашем репозитории обновлений прямо сейчас, лучше набраться терпения и немножко подождать. Самый свежий код можно будет протестировать, клонировав его на жесткий диск (`$ git clone git://anongit.freedesktop.org/pulseaudio/pulseaudio`) и скомпилировав. Загляните в документацию, поставляемую вместе с кодом.

Средство разработки

# eXeLearning

Версия: 2.0.4 Сайт: <http://exelearning.net>

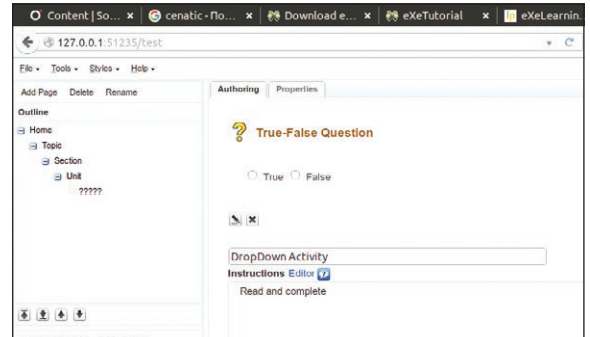
Э то развитие отличной программы *eXeLearning* с открытым кодом, разрабатываемой CENATIC, испанским государственным центром технологий, для сферы образования, чтобы помочь преподавателям в подготовке материалов для студентов. Суть ее в предоставлении не слишком технически подкованным преподавателям простого в работе инструмента, который позволит им публиковать свои тексты и приложения, не обладая углубленным знанием XHTML, HTML5, CSS, Markdown и прочих технологий.

После установки *eXeLearning* запускается командой \$ *exe*, которая создает локальный web-сервер на <http://127.0.0.1:51235> и открывает web-страницу в web-браузере по умолчанию. Язык, формат разметки и web-браузер пользователь должен выбрать сам.

Сценарий продуктивности для *eXe* начинается с создания правильной структуры страницы. Обратите внимание, что кнопки Add page, Delete и Rename размещаются

на верхней левой панели. С их помощью вы можете подготовить дерево категорий (Topic > Section > Unit > ...), а затем заполнить его контентом. На нижней левой панели расположен список разных шаблонов, именуемых iDevices. Требуемую категорию можно раскрыть и дважды щелкнуть по пункту с типом информации, которую надо добавить на страницу; это может быть голосование, викторина, текстовый ввод, изображение и видео YouTube или Vimeo, и т. д. И, наконец, можно будет сохранить всю структуру как «пакет» с названием.

Для каждого пакета *eXe* может сгенерировать интерактивный контент в XHTML или HTML5 и создать web-страницы с простой навигацией, включающие текст, изображения, интерактивную деятельность,



► **Постройте web-страницу с опросами и заданиями, не думая о языке разметки.**

галереи изображений или медиа-клипы. Все образовательные материалы, созданные в *eXeLearning*, можно экспортировать в разные цифровые форматы и использовать независимо или интегрировать в LMS (Learning Management System — Систему Управления Обучением), например, *Moodle*.

Инструмент по большей части основан на Python, и легко скомпилируется в любом дистрибутиве Linux. Есть также PPA Ubuntu на [ppa:exelearning/exelearning](http://ppa:exelearning/exelearning), пакет RPM для Fedora и независимая от дистрибутива статическая сборка в виде TAR.GZ.

**Помочь преподавателям в подготовке материалов для студентов.**

Плеер мультимедиа

# Bomi

Версия: 0.9.11 Сайт: <https://bomi-player.github.io>

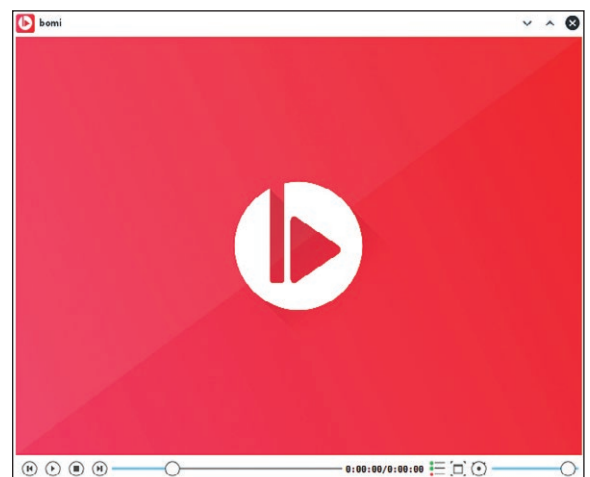
Э легантный мультимедиа-плеер *Bomi* создавали как современный GUI для движка MPV. Он выглядит очень стильно — по всему окну всплывает логотип *Bomi*, а всякие кнопки управления невидимы. Для доступа к меню приложений надо щелкнуть правой кнопкой мыши где-нибудь на окне *Bomi* и проследовать по требуемым пунктам меню.

В этом плеере — колоссальное число функций и опций. Прежде всего, преимущества ему дают функции MPV. MPV — это ответвление *Mplayer*, имеющее лучшую поддержку удаленных URL, стриминга, вывода OpenGL и функций кодирования и современный оптимизированный код (удалено множество обветшавших элементов *Mplayer*). И поверх всего этого *Bomi* добавляет свои функции, такие, как интерполяция видео, точный поиск, настройка субтитров, отключение скринсейвера, и т. д.

Обработка видео в *Bomi* позволяет улучшить качество видео и для создания более

приемлемой картинки использует передовые алгоритмы, такие, как Lanczos и маска Unsharp.

При воспроизведении видео *Bomi* не путается у вас под ногами, пока вы не наведете мышью на окно. Появится полупрозрачная панель OSD с кнопками управления воспроизведением и еще кое-чем — кнопкой Playlist, кнопками выбора разных субтитров и аудиотреков. Есть еще «скрытая» панель, она выпадает из левой части окна *Bomi* и отображает историю воспроизведения. Следует признать, что *Bomi* — очень сильный претендент на звание самого передового медиа-плеера из всех, виденных нами, со множеством современных функций, как то: ускорение оборудования,



► **UI *Bomi* не возникает, пока вы не наведете на него мышью.**

поддержка потокового вещания, создание плей-листов, и т. д. *Bomi* — относительно недавний проект, и многие дистрибутивы Linux еще не включили его в свои репозитории. Однако официальная страница Downloads *Bomi* предлагает ряд пакетов, которые удовлетворяют многих пользователей, включая пользователей Ubuntu, Arch, KaOS, openSUSE, Chakra и Fedora.

**Улучшает качество видео и использует передовые алгоритмы.**



## Музыкальный плеер

## QMMP

Версия: 1.0 Сайт: <http://qmmp.ylsoftware.com>

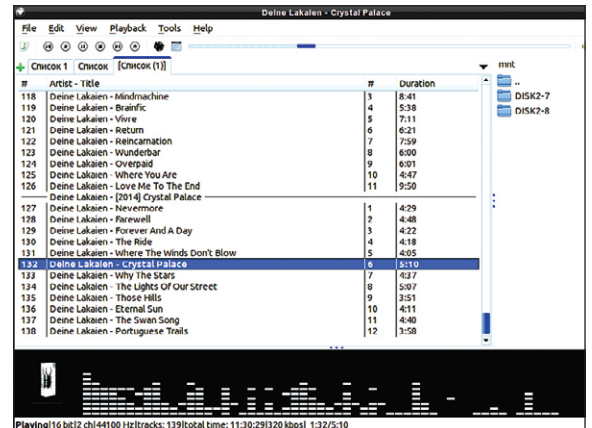
**Q**MMP означает *Qt-based Multimedia Player*, но, в отличие от *Bomi*, он в первую очередь предназначен для воспроизведения музыки и работы с плей-листами и медиа-библиотеками. Выделяется *QMMP* очень разумным разделением на базовую часть и GUI, поверх которых размещены десятки дополнительных модулей. Столь сильный акцент на дизайн проекта обусловил неплохую гибкость — например, возникновение нескольких интерфейсов, не требующих настройки части ядра плеера. *QMMP* начинался как клон *Winamp* с полной поддержкой всех скинов *Winamp 2.x*, но с более широким набором функций. *QMMP* поддерживает форматы файлов через модули вроде *qmmp-fmpeg*, *qmmp-aac* и т. д., позволяя в любой момент добавить и другие модули.

В *QMMP 0.9* можно переключить внешний вид плеера в режим 'simple', и тогда *QMMP* будет выглядеть, как простое приложение *Qt4* со стандартной панелью инструментов и кнопками управления. Но для

любителей плееров чуть более современного вида команда *QMMP* усердно потрудились над подготовкой ветви 1.0, которая предлагает стиль в духе *Clementine*.

В ближайшее время разработчики планируют одновременно поддерживать как ветвь 0.9.x (*Qt4*), так и самую последнюю ветвь 1.x; но если вы не являетесь убежденным фанатом *Winamp*, предпочтительнее будет версия поновее. Код у *QMMP 1.0* лучше и не имеет зависимостей от более старых версий *Taglib* и *Jack*. Также предлагаются новый эквалайзер, ровная и стабильная поддержка FLAC и Wavpack и низкое потребление памяти. Кроме того, у *QMMP* впечатляющее количество встроенных плагинов; некоторые из них по умолчанию не включены, но тем не менее стоят

## У QMMP впечатляющее количество встроенных плагинов.



► *QMMP* — основанный на *Qt* плеер с элегантно современным видом и отлично сделанным кодом под капотом.

ознакомления. В настройках *QMMP* в разделе *Plugins* отметьте окошко для плагинов, которые вы хотите включить, и перезапустите плеер, чтобы ваш выбор вступил в силу. Есть также удобные сторонние плагины — например, для поддержки YouTube и ряда экзотических форматов ввода, типа *YM* и *OptimFROG*.

Большинство дистрибутивов всё еще предлагают классическую версию *QMMP 0.9*, так что получение релиза 1.0 может потребовать компиляции из исходника. Скачайте tar-архив с сайта *QMMP* и скомпилируйте его как обычное приложение с *Cmake*.

## Система сканирования

## SANE

Версия: 1.0.25 Сайт: <http://sane-project.org>

**В**сякий раз, когда вы подключаете свой сканер или МФУ к компьютеру с Linux, с ним работают компоненты *SANE* (что означает *Scanner Access is Now Easy* — Доступ к Сканеру Теперь Прост). *SANE* существует с 1996 г. и включает пользовательские драйверы сканера (движки) для определенных моделей, утилиту *scanimage* для тестирования, демон *saned* для сетевого сканирования и ряд библиотек для работы со сканерами с помощью API *SANE*.

Поддерживается ли ваш сканер и какие функции реализуются, в большой мере зависит от пакета *Sane-backends*. Пакет был недавно обновлен, спустя почти два года разработки, и вышел новый *SANE 1.0.25*. Довольно долго утилита *scanimage* умела только сохранять растровые изображения в TIFF, но наконец-то обзавелась поддержкой JPEG и PNG. Теперь поддерживаются в общей сложности 1515 сканеров, из которых около 300 — новые или ранее не поддерживаемые модели.

С *SANE 1.0.25* большинство новых сканеров Canon отлично работают в Linux, как и некоторые модели Epson, PIE и Reflecta. У большинства существующих движков почищен код и исправлены ошибки, так что теперь всем, кто раньше жаловался на постоянные ошибки в работе сканеров в Linux, настоятельно рекомендуем попробовать новый релиз *SANE*. Остальная часть *SANE* долгие годы пребывала стабильно.

Приложения, подобные *XScan*, которое включено в пакет *sane-front-ends*, прочны, как скала, и остаются популярными. Проект *SANE* официально рекомендует использовать пакеты вашего дистрибутива Linux, однако, вероятно, придется некоторое время дожидаться подготовки обновлений.

## Поддерживаются 1515 сканеров, из которых около 300 — новые.

**SANE: Backends (Drivers)**

The following table summarizes the backends/drivers distributed with sane-backends-1.0.25, and the hardware software they support.

This is only a summary! Please consult the manpages and the author-supplied webpages for more detailed (and important!) information concerning each backend.

If you have new information or corrections, please file a [bug report](#) with as many details as possible. Also, if your scanner isn't mentioned in this list at all.

For an explanation of the tables, see the [legend](#).

**Summary**

Device type	Total	Number of devices						
		Sum	Complete	Good	Basic	Minimal	Untested	Unsupported
Scanners	2366	1515	681	699	110	25	369	482
Still cameras	5	5	1	2	2	0	0	0
Video Cameras	8	3	0	1	0	2	3	2
Meta backends	2					n/a		
APIs	5					n/a		

► Как видно из списков совместимости сайта *SANE*, поддерживается много новых сканеров.

В качестве обходного пути — можно скомпилировать *sane-backends* из исходника (<http://bit.ly/SaneBackendsSource>), используя традиционную команду `$. /configure && make && make install`, и она должна справиться с этой задачей, поскольку исходники *SANE* имеют очень мало зависимостей. Альтернатива — запустить тестовую виртуальную машину с самым современным Linux, например, Arch или Tumbleweed.

## HotGames Развлекательные приложения

Игра-аркада

# Solar Wolf

Версия: 1.5 Сайт: <http://bit.ly/SolarWolf>

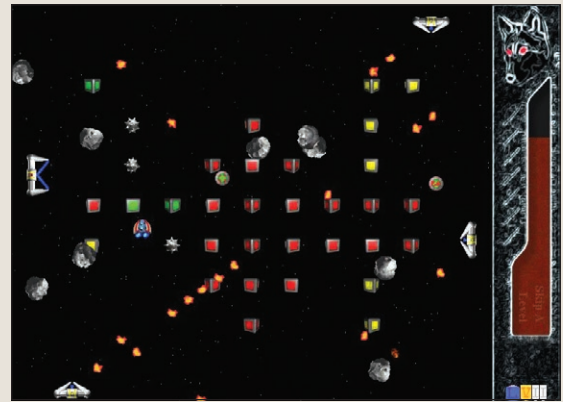
**М**ы продолжаем вновь открывать для себя классические игры, незаслуженно забытые и покрывшиеся паутиной в кладовке Интернета. И так, страхнем пыль с *Solar Wolf*, относительно современной реинкарнации *SolarFox*, вышедшей в 1981 г. и затем портированной на Atari 2600. *Solar Wolf* — аккуратный и надежный порт *SolarFox* для современных ОС, а среди них и Linux. Игра написана на Python и использует вывод SDL, хотя мы подозреваем, что она могла бы работать весьма ровно и без разгона OpenGL. Так или иначе, нам эта игра показалась захватывающей.

Вы должны провести свой небольшой космический корабль через частокороль прямоугольных матриц солнечных батарей внутри корпуса. Захватывая батареи, вы должны уклоняться от огненных шаров, которые в вас швыряют охранники,

постоянно гоняющиеся вдоль периметра. От уровня к уровню задача усложняется, поскольку добавляется все больше шаров и препятствий, таких, как мины и астероиды. Ваш корабль может заодно собирать бонусы, которые или замедляют темп, или обеспечивают дополнительную защиту. В каждой игре у вас три жизни, и если все они потрачены, можно начать играть с места вашей кончины. Это позволяет продвигаться на более сложные уровни, а не просто плюнуть на игру.

Игра сопровождается бодрой музыкой, а иногда — волчьим воем, и несмотря на простую графику и ограниченное

**Порт SolarFox для современных ОС, включая Linux.**



► *Solar Wolf* заставит вас собирать солнечные батареи, уворачиваясь от огненных шаров и астероидов!

число препятствий, *SolarWolf* является отличным способом убить время и улучшить реакцию.

Установить игру элементарно: у многочисленных дистрибутивов Linux она имеется в стандартных репозиториях, или скачайте ее прямо с сайта проекта. Игра адаптирована для работы на современных широкоэкранных дисплеях, хотя не так давно это было проблемой.

Имитатор гонок

# TORCS

Версия: 1.3.6 Сайт: <http://torcs.sourceforge.net>

**С**ейчас в Linux полным-полно имитаторов гонок, хотя код открыт у немногих из них. *TORCS* (The Open Racing Car Simulator) — отличная игра, восполняющая этот недостаток. Это гоночный симулятор 3D для многих платформ (включая Linux), в котором вы состязаетесь с созданными компьютером соперниками.

*TORCS* интересна точностью имитации динамики автомобиля и вращающегося момента разных двигателей. Не ограничиваясь простыми ускорением и торможением, машины в *TORCS* ведут себя почти как в реальной жизни, с переключением передач, реалистичным заносом при поворотах на высокой скорости и точным входом в поворот, в зависимости от веса и габаритов машины. Фоновая музыка и рев двигателей воссоздают атмосферу гонок! Здесь десятки автомобилей и всего 41 трасса, в т. ч. 24 дорожных, 9 грунтовых

и 9 овальных гоночных трасс, которые собраны в пакет весом 195 МБ. Игра поддерживает различные режимы, так что вы можете погонять с другом на том же компьютере или играть против нескольких роботов ИИ; однако в *TORCS* нет многопользовательского сетевого режима.

При запуске игры появляется основное меню для смены настроек. Стандартный процесс, кроме опции 'Quick Race', требует выбора трека, числа участников, уровня ИИ и марки автомобиля, а также ввода вашего ника.

Поиграть в *TORCS* — истинное удовольствие, поскольку это одна из самых зрелищных и реалистичных гонок

**Одна из самых зрелищных и реалистичных гонок в Linux.**



► Ботов ИИ довольно трудно обогнать, играя на уровне высокой сложности.

в Linux с качественной графикой. Причем *TORCS* способна работать не только на дорогостоящих видеокартах: если малость снизить настройки видео, интегрированная графика тоже сгодится.

## Конвертор формата файлов изображений

## FLIF

Версия: GIT Сайт: <http://bit.ly/FLIFEncoder>

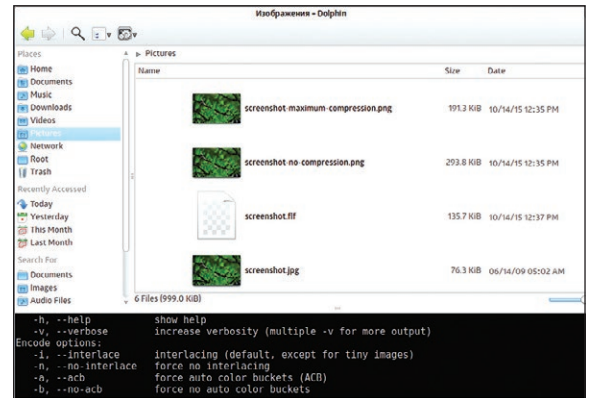
**К**оэффициент сжатия файлов растровой графики — извечная проблема. Все предпочитают файлы малого объема, более быструю загрузку страниц, больше свободного места на жестком диске и отсутствие необходимости тратить лишние деньги на дополнительные мобильные данные. Большинство картинок, которые вы находите в Сети — или JPG, или PNG, и оба формата неплохо соблюдают равновесие между сжатием и качеством изображения. Однако нельзя ли добиться еще лучшего?

FLIF — новый формат изображений без потерь, основанный на методе сжатия MAnIAC (что означает Meta-Adaptive Near-zero Integer Arithmetic Coding), который по коэффициенту сжатия способен превзойти PNG, FFV1 и не имеющие потерь WebP, BPG и JPEG2000.

Чтобы попробовать FLIF, надо скачать моментальный снимок кода с сайта проекта и проверить, есть ли у вас пакет *libpng-dev* (или с похожим именем), и затем ввести

`$ make`. Через несколько секунд вы получите скомпилированный бинарник *flif*, который работает как преобразователь и декодер изображений FLIF. Синтаксис очень прост: `./flif input.png output.flif`, а традиционная `./flif --help` покажет список опций, например, чередование, палитра и качество, и т.д. Помните, что FLIF пока что умеет конвертировать только форматы файлов PNG, PAM и PNM, поэтому есть смысл сравнить результаты FLIF с PNG, как с наиболее распространенным форматом в списке.

В наших тестах результат FLIF без потерь был меньше, чем PNG при наивысшем коэффициенте сжатия, и по размеру файла может сравниться с JPG. Единственная проблема — файлы FLIF пока нельзя просматривать: можно только применить бинарник



➤ Единственное, чего на самом деле не хватает FLIF — это просмотрщика.

FLIF для архивирования изображений и их последующего восстановления командой `./flif -d input.flif output.png`. Но можно посмотреть кодированные во FLIF изображения при помощи отдельного инструмента под названием UGUI\_FLIF (<http://bit.ly/UGUI-FLIF>), основанного на HTML и JSON. Как только формат файлов FLIF будет окончательно завершен, появится более адекватная программа просмотра, но файлы, преобразованные с помощью более старого бинарника FLIF, не будут совместимы с новым. И всё же этот активно разрабатываемый преобразователь — весьма обнадесивающая новинка.

**Результат меньше, чем PNG с наивысшим коэффициентом сжатия.**

## Инструмент 3D-моделирования

## Blender

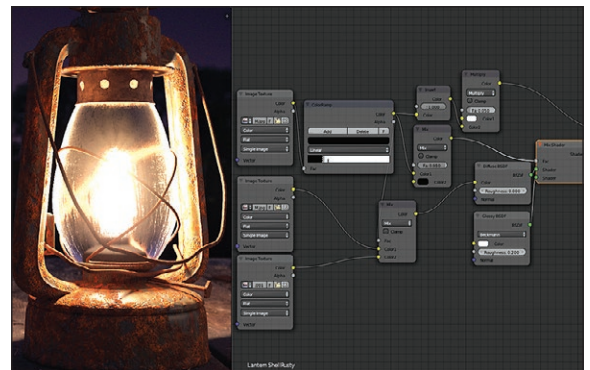
Версия: 2.76 Сайт: [www.blender.org](http://www.blender.org)

**B**lender — превосходный проект, которым по праву может гордиться сообщество Linux с самого момента его появления в 1998 г. Созданный как любительский проект группы голландских энтузиастов, Blender стал полностью интегрированным пакетом создания 3D-контента, который предлагает широчайший выбор основных инструментов: моделирование; UV-картирование; текстуры; сборка и создание поверхностей; анимация; имитация разных видов (текучие и твердые объекты); скрипты; рендеринг; композиция; создание VFX [видеоэффектов] и игр.

Интерфейс оптимизирован для одновременного использования мыши и клавиатуры, и Blender снабжен экранным ориентированным курсором, который позволяет размещать объекты в 3D-пространстве. Наведя курсор с помощью мыши, можно использовать клавиши быстрого запуска для размещения объекта (освещение и геометрия, и т.д.). Рендеринг запускается нажатием

F12. Разумеется, работа с Blender предполагает сложный процесс обучения для новичков, так что вам придется поистратить изрядное количество времени, прежде чем вы сможете создать нечто посложнее, чем, скажем, шар на плоскости.

Однако опытным пользователям, знакомым с 3D Max и Maya, Blender предлагает очень удобную предустановку Interaction. Недавний релиз Blender, 2.76, позволяет выделять потрясающие вещи, например, фотореалистичный HDR-рендеринг с использованием GPU, высококачественные текстуры из множества материалов (стекло, металл, и т.д.), и имеет отличный набор инструментов анимации. Новая версия стала еще одним шагом вперед: улучшилось



➤ Blender — умелая программа. Ржавые поверхности можно создать наложением разных слоев и эффектов.

использование GPU от AMD в Linux; добавлена начальная реализация «напыла камеры», новые инструменты моделирования 'Flatten faces' и 'Edge offset', поддержка библиотеки Pixar OpenSubdiv, и т.д.

Официальный сайт Blender предлагает разные опции скачивания, но не пакеты для определенных дистрибутивов Linux. Лучше найти его в ваших репозиториях. Например, пользователи Ubuntu могут скачать последнюю версию Blender с [ppa:thomas-schiex/blender](http://ppa:thomas-schiex/blender). LXF

**Blender позволяет фотореалистичный HDR-рендеринг с GPU.**

# На диске

Дистрибутивы, приложения, игры, книги и всякое-разное...

Лучшее из Интернета, упакованное в 9 ГБ качественного DVD.



## Дистрибутивы

Старая поговорка гласит: «Есть много способов ободрать кошку». В наши дни, наверное, лучше будет сказать: «Есть миллионы способов сделать видео с котиком и выложить его на YouTube». В любом случае, это применимо к DVD этого месяца.

Месяц назад я возмущался новыми правилами Ubuntu насчет распространения модифицированных версий дистрибутива, а сейчас предпринял более конструктивный подход. В итоге у нас всё равно вышел ремикс рабочих столов, благодаря свойству устойчивости, встроенной в систему загрузки Ubuntu. Она разработана так, чтобы вы могли использовать USB-диск или файл на вашем жестком диске как постоянное хранилище для системы live и сохранять файлы и настройки. Мы применили ее для хранения добавочных рабочих столов вместе с их файлами и настройками в отдельном файле на DVD. Пусть я и не восторге от их юристов, но отдаю должное тем, кто очень внятно документировал систему загрузки Ubuntu, casper, сделав эту задачу намного понятнее. И так, юристы счастливы — ведь мы предоставляем беспримесный ISO-образ Ubuntu; и вы тоже будете счастливы, поскольку добавочные рабочие столы там всё же есть. И все в выигрыше!

*Neil*

## Важно ВНИМАНИЕ!

### Порченные диски

В маловероятном случае какого-то дефекта вашего LXF DVD обращайтесь, пожалуйста, по адресу [disks@linuxformat.ru](mailto:disks@linuxformat.ru) или телефону +7 (812) 309-0686.

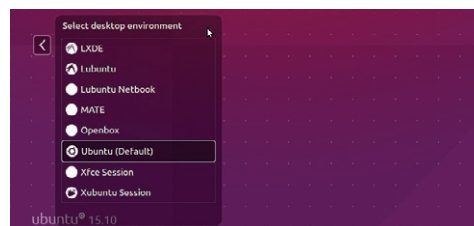
## Ремикс

# Ubuntu 15.10

В этом месяце на DVD *Linux Format* — праздник нового релиза Ubuntu, с несколькими вариациями. Первым в меню загрузки стоит один из наших ремиксов с разными рабочими столами, в котором вы можете переключаться между Unity, LXDE, Mate и Xfce.

LXDE и Xfce являются легковесными рабочими столами, что хорошо для старого оборудования или для желающих употребить большую часть своих ресурсов для работы с приложениями, а не с рабочим столом. Mate вдохновлялся рабочим столом Gnome 2 и предлагает более традиционный интерфейс, но с современными усовершенствованиями. По умолчанию DVD загружает Unity; если хотите попробовать другие рабочие столы, выйдите из него, нажмите на маленький логотип Ubuntu справа от окна приглашения, чтобы открыть меню с разными опциями, и войдите как пользователь ubuntu с пустым паролем. Для LXDE и Xfce выберите соответственно Lubuntu и Xubuntu, и получите все прелести Ubuntu на соответствующем рабочем столе.

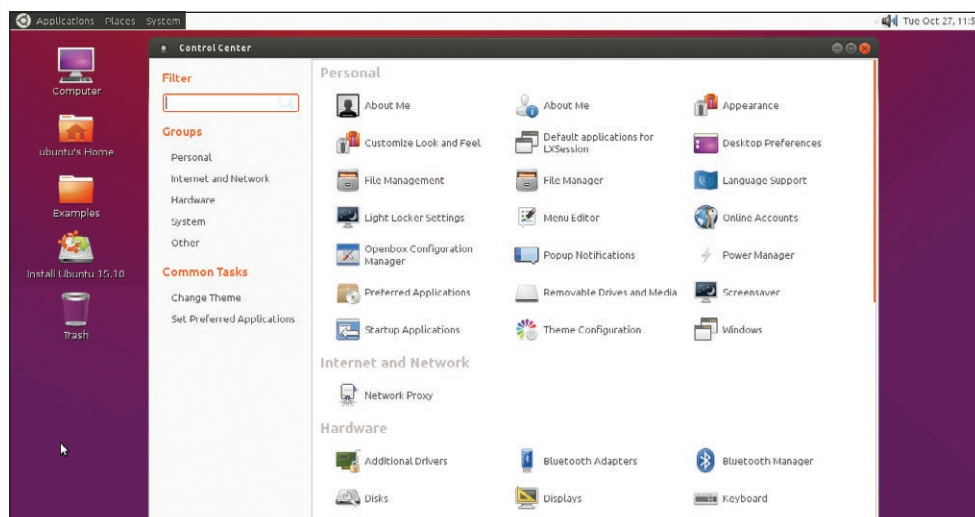
Из-за ограничений, которые Ubuntu налагает на ремастеринг образов своего диска, нам пришлось изменить способ создания ремикса, и вы не сможете выполнить установку прямо отсюда: процесс-то запустится, но потом выдаст ошибку. Выберите опцию установки из меню загрузки или загрузите стандартную альтернативу рабочего стола и установитесь оттуда. Чтобы добавить к своей установке рабочие столы, откройте менеджер приложений и установите себе один или более из *lubuntu-desktop*, *mate-desktop-environment* или



Выйдя из рабочего стола Ubuntu Unity, вы сможете выбрать совершенно другой рабочий стол.

*xubuntu-desktop*. Каждый из этих пакетов установит все нужные для вашего рабочего стола пакеты. Возможно, вам надо будет включить дополнительные репозитории, чтобы они стали доступными — и это очень просто. Запустите менеджер пакетов *Synaptic*, выберите пункт меню *Settings > Repositories* и отметьте окошко 'universe'. Вероятно, вы захотите включить там заодно и multiverse. Закройте это окно, нажмите *Reload*, и все дополнительные пакеты станут доступны для вас.

Одно из преимуществ этого метода — то, что исходный ISO-образ остается нетронутым. Это означает, что любой желающий увидеть работу чистого Ubuntu 15.10 ее получит: вторая опция в меню загрузки стандартный рабочий стол Ubuntu Unity, который вы можете использовать или установить. Один из мелких недостатков данного подхода — то, что загрузка смиксованной системы кажется дольше, чем это было раньше, но это заметно только при медленном DVD-приводе.





# Новичок в Linux? Начните отсюда!

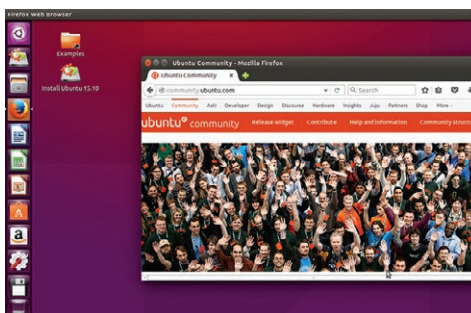
- » Что такое Linux? Как его установить?
- » Есть ли в нем эквивалент *MS Office*?
- » Зачем нужна командная строка?
- » Как устанавливать программы?

Ответы приводятся в [Index.html](#) на диске.

Редакция с меньшим числом битов

## Ubuntu 15.10

Все меньше и меньше дистрибутивов в наши дни предлагают 32-битные версии — в основном потому, что мало остается тех, кто использует это оборудование, а тестирование двух версий для всего подряд затратно по ресурсам. Но крупные дистрибутивы вроде Ubuntu располагают для этого сотрудниками, поэтому 32-битная версия Ubuntu до сих пор существует; и мы ее приложили. Мы рассматривали более легкие альтернативы, скажем, Lubuntu, но сочли, что большая часть поклонников Ubuntu захочет иметь основную версию. А если ваше оборудование не тянет Unity, может быть, стоит установить один из рабочих столов полегче.



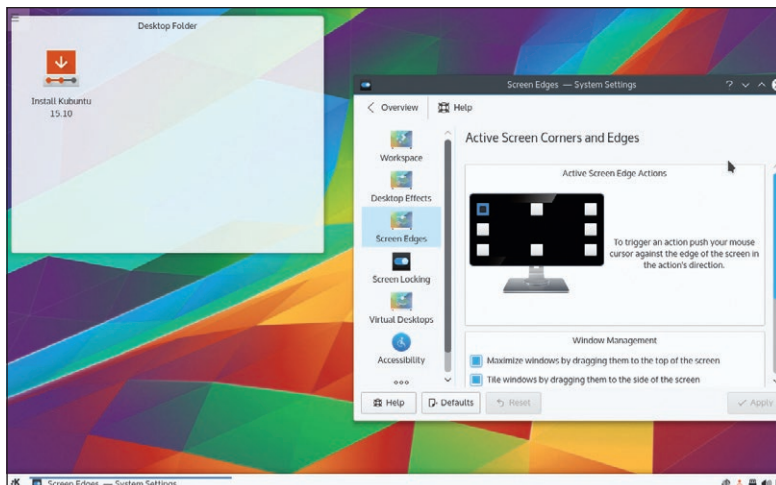
» Вы просили — мы сделали: встречайте 32-битную редакцию Ubuntu 15.10.

Редакция Frameworks 5

## Kubuntu 15.10

Возможно, вы обратили внимание на серьезный пробел в списке рабочих столов, предлагаемых нашим ремиксом. Добавление KDE приводило к неким компромиссам в прошлом, а сейчас и по-прежнему. Поскольку места у нас хватало (да и надо-то было всего 5 МБ), мы добавили Kubuntu как отдельный дистрибутив. Он отличается куда большей настраиваемостью, чем все прочие варианты, и подобный способ работы означает, что вы будете применять его именно так, как и предполагали разработчики. Не пугайтесь

черного экрана при загрузке: между моментом исчезновения всплывающего экрана и появлением рабочего стола есть небольшая задержка — это нормально, это не сбой. Загрузка с DVD всегда будет медленнее, отчетливо пауза заметнее. Затем вы увидите довольно стандартный рабочий стол KDE Frameworks 5. Ключевое слово здесь — «довольно» стандартный: KDE отлично поддается индивидуальной настройке, и разные дистрибутивы могут иметь ничуть не похожие друг на друга виды по умолчанию. **LXF**



» Вам понадобится терпение, чтобы дождаться загрузки KDE Frameworks 5, зато его настройка — сплошное удовольствие.

## И еще!

Системные инструменты

### Главное

**Checkinstall** Установка tar-архива с помощью менеджера пакетов.

**GNU Core Utils** Основные утилиты, обязанные быть в каждой операционной системе.

**Hardinfo** Инструмент тестирования системы.

**Kernel** Исходный код самого последнего стабильного релиза ядра.

**Memtest86+** Проверьте ОЗУ на предмет сбоев.

**Plop** Простой менеджер загрузки для запуска ОС с CD, DVD и USB.

**RaWrite** Создавайте загрузочные дискиеты в MS-DOS в Windows.

**SBM** Независимый от ОС менеджер загрузки с несложным интерфейсом.

**WvDial** Соединяйтесь с Интернетом через телефонный модем.

Чтение

### Книжная полка

**Расширенное руководство по скриптам Bash** Изучите написание скриптов еще лучше.

**Руководство Bash для начинающих** Осваивайте написание скриптов Bash.

**Руководство по созданию скриптов Bourne Shell** Начните осваивать скрипты оболочки.

**The Cathedral and the Bazaar [Собор и Базар]** Классический текст Эрика С. Реймонда [Eric S Raymond], объясняющий преимущества открытой разработки.

**Справочник администратора Debian** Базовое руководство для системных администраторов.

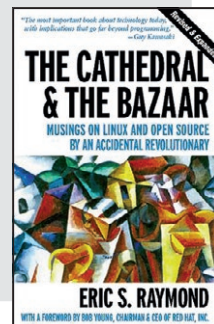
**Введение в Linux** Удобное руководство со множеством подсказок для начинающих пользователей Linux.

**Словарь Linux** Linux от А до Я.

**Linux Kernel in a Nutshell [Ядро Linux в Двух Словах]** Введение в ядро, написанное великим магистром ядра Греггом Кроа-Хартманом [Greg Kroah-Hartman].

**Руководство системного администратора Linux** Контролируйте свою систему.

**Обзор инструментов** Полный обзор инструментов GNU.



## Пропустили номер?



Закажите его через сайт [www.linuxformat.ru](http://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:




**LXF202**  
Ноябрь 2015

- » Академия от Тукса Учимся кодировать
- » Хромбуки Коммуникатор стал работагой
- » 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/](http://shop.linuxformat.ru/lxf_202/)  
PDF-версия: [shop.linuxformat.ru/elxf\\_202/](http://shop.linuxformat.ru/elxf_202/)



**LXF203**  
Декабрь 2015

- » 200 полезных советов Ради праздника!
- » Не Linux'ом единым Альтернативы есть
- » Откуда берутся пакеты Фермы по сборке
- » Немного ностальгии Предадимся воспоминаниям

**LXFDVD:** Fedora 22, Peppermint OS, Q4OS, Ubuntu, Linux Lite, Sabayon, Tails, IPFire, Sparky, 11 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: [shop.linuxformat.ru/lxf\\_203/](http://shop.linuxformat.ru/lxf_203/)  
PDF-версия: [shop.linuxformat.ru/elxf\\_203/](http://shop.linuxformat.ru/elxf_203/)



**LXF204**  
Январь 2016

- » Поток вещания Домашние медиа-центры
- » Быть в теме Синхронизируем свои девайсы
- » Кино и Linux Свободная 3D-графика в Голливуде
- » Печатаем и сканируем Подружим Linux с периферией

**LXFDVD:** Kodibuntu, OpenELEC, openSUSE, OSMC, Ubuntu 15.04, Q4OS, 10 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: [shop.linuxformat.ru/lxf\\_204/](http://shop.linuxformat.ru/lxf_204/)  
PDF-версия: [shop.linuxformat.ru/elxf\\_204/](http://shop.linuxformat.ru/elxf_204/)

А чтобы не упустить ничего, оформите подписку! Все, кто подписался на печатную версию журнала через [www.linuxformat.ru/subscribe](http://www.linuxformat.ru/subscribe) или [www.linuxcenter.ru](http://www.linuxcenter.ru), получают электронную версию в подарок! На сайте [shop.linuxformat.ru](http://shop.linuxformat.ru) вы также сможете приобрести предыдущие выпуски LXF.

Подписывайтесь на сайте [www.linuxformat.ru/subscribe](http://www.linuxformat.ru/subscribe)

Телефоны отдела подписки:

- » Санкт-Петербург (812) 309-0686
- » Москва (499) 271-4954



## Linux Format VKontakte

Вступайте в нашу  
группу [vk.com/linuxform](http://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

## » Содержание

**LINUX**  
FORMAT

### Страница 1

#### ДИСТРИБУТИВЫ

**Fedora 23 LXDE** (64-битный)  
**Ubuntu 15.10** (64-битный) с восьмью рабочими столами: LXDE, Ubuntu, Ubuntu Netbook, Mate, OpenBox, Unity, Xfce и Xubuntu  
**Ubuntu 15.10** (32-битный)

#### НОПРИСКС

**Blender 2.76** Инструмент 3D-моделирования  
**Exhalelight 2.0.4** Средство разработки  
**FLIC (G17)** Преобразователь формата файлов изображений  
**Lin 0.3** Web-браузер  
**PulseAudio 7.0** Аудиосервер  
**OMMP 1.0** Музыкальный плеер  
**SAHE 1.0.25** Система сканирования  
**Solar Wolf 1.5** Игра-аркада  
**SweetHome3D 5.1** Планировщик домашнего интерьера  
**TORCS 1.3.6** Имитатор гонок

#### СРАВНЕНИЕ: ВИДЕОПЛЕЙЕРЫ

Вопи 0.9.11  
OMPlayer 2 15.10.18  
Ropri 1.6  
SMPlayer 15.9  
VLC 2.2.1

#### УЧЕБНИКИ

Perl 6  
zsh/Gsurl

Окончание на обороте »

## Информация о диске

### Что-то потеряли?

Часто случается, что новые программы зависят от других программных продуктов, которые могут не входить в текущую версию вашего дистрибутива Linux.

Мы стараемся предоставить вам как можно больше важных вспомогательных файлов. В большинстве случаев, последние версии библиотек и другие пакеты мы включаем в каталог «Essentials [Главное]» на прилагаемом диске. Поэтому, если в вашей системе возникли проблемы с зависимостями, первым делом следует заглянуть именно туда.

### Форматы пакетов

Мы стараемся включать как можно больше различных типов установочных пакетов: RPM, Deb или любых других. Просим вас принять во внимание, что мы ограничены свободным пространством и доступными двоичными выпусками программ. По возможности, мы будем включать исходные тексты для любого пакета, чтобы вы могли собрать его самостоятельно.

### Документация

На диске вы сможете найти всю необходимую информацию о том, как устанавливать и использовать некоторые программы. Пожалуйста, не забывайте, что большинство программ поставляются вместе со своей документацией, поэтому дополнительные материалы и файлы находятся в соответствующих директориях.

### Что это за файлы?

Если вы новичок в Linux, вас может смутить изобилие различных файлов и расширений. Так как мы стараемся собрать как можно больше вариантов пакетов для обеспечения совместимости, в одном каталоге часто находятся два или три файла для различных версий Linux и различных архитектур, исходные тексты и откомпилированные пакеты. Чтобы определить, какой именно файл вам нужен, необходимо обратить внимание на его имя или расширение:

- » **имя\_программы-1.0.1.i386.rpm** — вероятно, это двоичный пакет RPM, предназначенный для работы на системах x86;
- » **имя\_программы-1.0.1.i386.deb** — такой же пакет, но уже для Debian;
- » **имя\_программы-1.0.1.tar.gz** — обычно это исходный код;
- » **имя\_программы-1.0.1.tgz** — тот же файл, что и выше этажом по списку: “tgz” — это сокращение от “tar.gz”;
- » **имя\_программы-1.0.1.tar.bz2** — тот же файл, но сжатый bzip2 вместо обычного gzip;
- » **имя\_программы-1.0.1.src.rpm** — также исходный код, но поставляемый как RPM-пакет для упрощения процесса установки;
- » **имя\_программы-1.0.1.i386.FC4.RPM** — двоичный пакет RPM для x86, предназначенный специально для операционной системы Fedora Core 4;
- » **имя\_программы-1.0.1.ppc.Suse9.rpm** — двоичный пакет RPM, предназначенный специально для операционной системы SUSE 9.x PPC;
- » **имя\_программы-devel-1.0.1.i386.rpm** — версия для разработчиков.

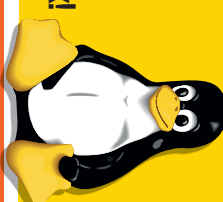
### Если диск не читается...

Это маловероятно, но если все же прилагаемый к журналу диск поврежден, пожалуйста, свяжитесь с нашей службой поддержки по электронной почте: [disks@linuxformat.ru](mailto:disks@linuxformat.ru)

Внимательно прочтите это перед тем, как использовать LXF DVD!

НАЧНИТЕ РАБОТУ В  
**LINUX**

Вставьте диск  
Загрузитесь  
Работайте в Linux!



УВЛИНТУ 15.10

Свежий релиз  
с 8 рабочими  
столами,  
готовыми  
к запуску  
и установке

**КУВЛИНТУ**  
Для рабочего стола Plasma 5

64-битная  
версия  
и 32-битная  
стандартная

# Содержание



## Страна 2

### ГЛАВНОЕ

Checkinstall  
Coreutils  
HardInfo  
Kernel  
Memtest86+  
Plop  
SBM  
Wvdial

### КНИЖНАЯ ПОЛКА (НА АНГЛ. ЯЗ.)

Расширенное руководство по скриптам Bash  
Руководство Bash для начинающих  
Руководство по созданию скриптов Bourne Shell.  
The Cathedral and the Bazaar (Собор и базар)  
Справочник администратора Debian  
Введение в Linux  
Словарь Linux  
Linux Kernel in a Nutshell [Ядро Linux в двух словах]  
Руководство системного администратора Linux  
Обзор инструментов GNU

### ДИСТРИБУТИВЫ

Koroga 23 Cinnamon (64-битный)  
Kubuntu 15.10 (64-битный)  
Tails 2.0 (32- и 64-битный)

Все дистрибутивы представлены ISO-образом, который можно и записать на отдельный носитель, и загрузить в live-режиме прямо с LXF DVD. У всех присутствует возможность установки на жесткий диск.

**Пожадуйства, перестаньте использовать один и тот же диск с инструкцией, опубликованной в журнале на стр. 109!**

**КОММЕНТАРИЙ** Присылайте ваши пожелания и предложения по электронной почте: [info@linuxformat.ru](mailto:info@linuxformat.ru)

**ДЕФЕКТИВНЫЕ ДИСКИ** В маловероятном случае обнаружения дефектов на данной диске, обращайтесь, пожалуйста, по адресу: [disk@linuxformat.ru](mailto:disk@linuxformat.ru)

Настоящий диск тщательно тестировался и проверялся на всех этапах производства, однако, как и в случае с любым новым ПО, мы рекомендуем вам использовать альтернативный сканер. Мы также рекомендуем всегда иметь под рукой актуальную резервную копию данных в вашем жестком диске. К сожалению, редакция Linux Format не в состоянии принимать на себя ответственность за любые повреждения, разрушения или иные убытки, которые могут повлечь за собой использование этого DVD, предоставленных нашей программой или данных. Прежде чем устанавливать какое-либо ПО на компьютер, пожалуйста, скачайте программу для загрузки с нашего сайта.

Тираж издательства ООО «Марком», 188652, Россия, Ленинградская область, Всеволожский р-н, дер. Юрки, Школьная ул., 7-а. Лицензия ИПТР ВАО № 77-03.

## Создание установочных дисков при помощи cdcrecord

Самый быстрый способ записать ISO-образ на чистую матрицу — это обратиться к программе *cdcrecord*. Для всех перечисленных ниже действий потребуются права суперпользователя-root. Сначала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdcrecord -scanbus
```

После этого на экране терминала должен отобразиться список устройств, подключенных к вашей системе. SCSI-адрес каждого устройства представляет собой три числа в левой колонке — например, 0,3,0. Теперь вы можете с легкостью записать образ на диск:

```
cdcrecord dev=0,3,0 -v /путь к образу/image.iso
```

Чтобы упростить дальнейшее использование *cdcrecord*, сохраните некоторые настройки в файле `/etc/default/cdcrecord`. Добавьте по одной строке для каждого устройства записи (скорее всего, в вашей системе присутствует только одно такое устройство):

```
Plextor= 0,3,0 12 16M
```

Первое слово в этой строке — метка; затем после адреса SCSI-устройства вы должны указать скорость и размер буфера. Теперь можете заменить SCSI-адрес в командной строке на выбранную вами метку. Все будет еще проще, если вы добавите следующее:

```
CDR_DEVICE=Plextor
```

Для записи ISO-образа вам осталось набрать команду

```
cdcrecord -v /path/to/image.iso
```

Если вы не принадлежите к любителям командной строки, в таком случае вам придет на помощь утилита *gcombust*. Запустите ее от имени root и выберите вкладку Burn и ISO 9660 Image в верхней части окна. Введите путь к образу, который вы хотите записать на диск, и смело нажимайте на Combust! Пока ваш образ пишется на диск, можете выпить чашечку кофе.

## Другая ОС?

Использовать Linux для записи компакт-диска не обязательно. Программы вроде *cdcrecord* просто переносят двоичные данные на чистую матрицу. Все необходимые файлы уже включены в ISO-образ, который распознается любой операционной системой, будь то Linux, Windows, Mac OS X или AmigaOS.

## Нет устройства для записи дисков?

Если у вас нет устройства, с помощью которого можно было бы записать образ на диск, можно найти какого-нибудь друга или организацию, у кого есть компьютер с дисководом, и прожечь диск у них. Опять-таки, вам подойдет любая операционная система, способная распознать пишущий привод (см. выше).

Некоторые дистрибутивы умеют монтировать образы дисков и выполнять сетевую установку или даже установку с раздела жесткого диска. Конкретные методы, конечно, зависят от дистрибутива. За дополнительной информацией обращайтесь на web-сайт разработчика дистрибутива.



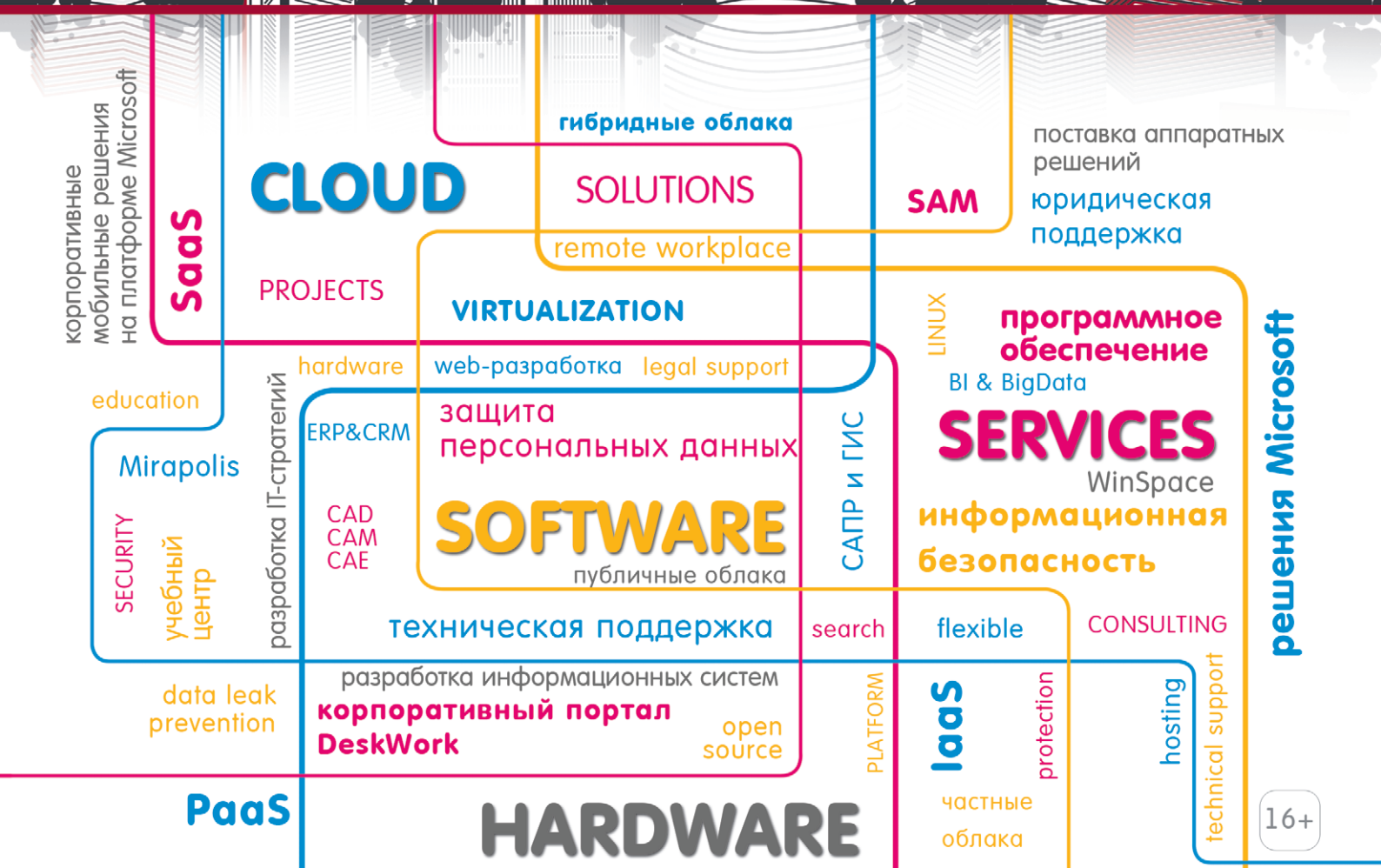
# softline®

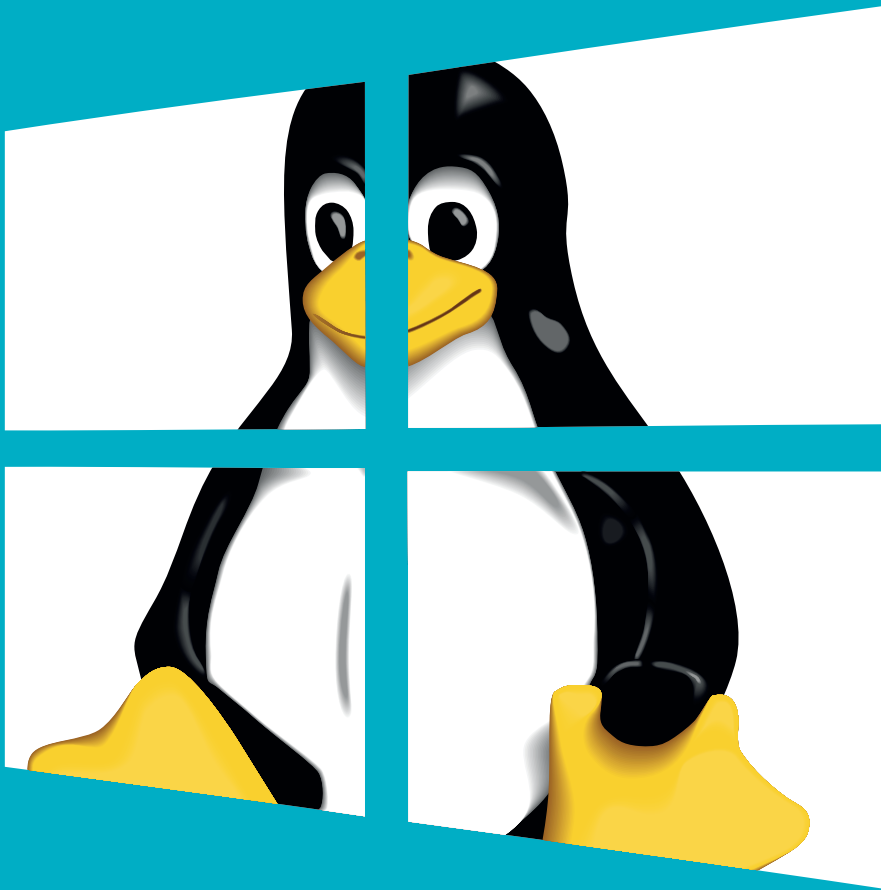
Cloud Software Hardware Services

# 20+

Years in IT

## IT-архитектура вашего бизнеса





В мартовском номере

# Сбежать с Windows 10

Обдумываешь побег из «запертого сада» Microsoft или хочешь помочь друзьям? Мы посодеествуем...

## Open Source в искусстве

Разберемся, как свободное ПО движет новым поколением художников, дизайнеров и прочих творцов.

## Перекодировать это!

Пришла охота перекодировать высококачественное HD-видео, но чтоб без мороки? Есть опции FLOSS.

## Резервная копия — это просто

Тестируем инструменты, с которыми это и правда семечки, так что отныне ничего не бойтесь.

Содержание будущих выпусков может меняться — вдруг нас завалил в туннеле под крепостью замка Microsoft...

# LINUX FORMAT

Главное в мире Linux

Журнал зарегистрирован Федеральной службой по надзору за соблюдением законодательства в сфере массовых коммуникаций и охране культурного наследия ПИ № Ф077-21973 от 14 сентября 2005 года. Выходит ежемесячно. Тираж печатной версии 2000 экз., распространение электронной версии 30000 экз.

### РЕДАКЦИЯ РУССКОЯЗЫЧНОЙ ВЕРСИИ

#### Главный редактор

Кирилл Степанов [info@linuxformat.ru](mailto:info@linuxformat.ru)

#### Литературный и выпускающий редактор

Елена Толстякова

#### Переводчики

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](mailto:advert@linuxformat.ru)

#### Генеральный директор

Павел Фролов

#### Учредители

Частные лица

#### Издатель

ООО «Линукс Формат»

#### Отпечатано в типографии ООО «ЛД-ПРИНТ»

196644, Санкт-Петербург, Колпинский р-н, пос. Саперный, территория предприятия «Балтика», д. 6/н, лит. Ф  
Тел. (812) 462-8383, e-mail: [office@dprint.ru](mailto:office@dprint.ru)  
Заказ 13753

### РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ

Редактор Нейл Мор [Neil Mohr] [neil.mohr@futurenet.com](mailto:neil.mohr@futurenet.com)

Научный редактор Джонни Бидвелл [Jonni Bidwell]

[jonni.bidwell@futurenet.com](mailto:jonni.bidwell@futurenet.com)

Выпускающий редактор Крис Торнетт [Chris Thornett]

[chris.thornett@futurenet.com](mailto:chris.thornett@futurenet.com)

Художественный редактор Эфраин Эрнандес-Мендоза

[Efraim Hernandez-Mendoza] [efrain.hernandez-mendoza@futurenet.com](mailto:efrain.hernandez-mendoza@futurenet.com)

### ПОДГОТОВКА МАТЕРИАЛОВ

Джонни Бидвелл [Jonni Bidwell], Нейл Ботвик [Neil Bothwick], Джолион Браун [Jolyon Brown], Дэйв Джонс [Dave Jones], Ник Пирс [Nick Peers], Лес Пландер [Les Pounder], Маянк Шарма [Mayank Sharma], Шашанк Шарма [Shashank Sharma], Александр Толстой [Alexander Tolstoy], Михалис Цукалос [Mihalis Tsoukalos], Евгений Балдин, Андрей Гондаренков, Алексей Федорчук, Максим Черепанов, Лада Шерышова

Иллюстрации Шейн Коллинж [Shane Collinge], Magic Torch

### КОНТАКТНАЯ ИНФОРМАЦИЯ

UK: Linux Format, 30 Monmouth Street, Bath BA1 2BW

Тел. +44 (0) 1604 251045, email: [linuxformat@myfavouritemagazines.co.uk](mailto:linuxformat@myfavouritemagazines.co.uk)

РОССИЯ: Санкт-Петербург, пр. Медиков, 5, корп. 7

Тел. +7 (812) 309-0686, e-mail: [info@linuxformat.ru](mailto:info@linuxformat.ru)

По вопросам сотрудничества, партнерства, оптовых закупок:

[partner@linuxcenter.ru](mailto:partner@linuxcenter.ru)

**Авторские права:** статьи, переведенные из английского издания Linux Format, являются собственностью или лицензированы Future Publishing Ltd (Future plc group company). Все права зарегистрированы. Никакая часть данного журнала не может быть повторно опубликована без письменного разрешения издателя.

Все письма, независимо от способа отправки, считаются предназначенными для публикации, если иное не указано явно. Редакция оставляет за собой право корректировать присланные письма и другие материалы. Редакция Linux Format получает неэксклюзивное право на публикацию и лицензирование всех присланных материалов, если не было оговорено иное. Linux Format стремится оставлять уведомление об авторских правах всюду, где это возможно. Свяжитесь с нами, если мы не упомянули вас как автора предложенных вами материалов, и мы постараемся исправить эту ошибку. Редакция Linux Format не несет ответственности за опечатки.

Ответственность за содержание статьи несет ее автор. Мнение авторов может не совпадать с мнением редакции.

Все присланные материалы могут быть помещены на диски — CD или DVD, поставляемые вместе с журналом, если не было оговорено иное.

**Ограничение ответственности:** используйте все советы на свой страх и риск. Ни при каких условиях редакция Linux Format не несет ответственности за повреждение или ущерб, нанесенные вашему компьютеру и периферии вследствие использования тех или иных советов.

Linux — зарегистрированный товарный знак Линуса Торвальдса [Linus Torvalds].

“GNU/Linux” заменяется на “Linux” в целях сокращения. Все остальные товарные знаки являются собственностью их законных владельцев. Весь код, опубликованный в журнале, лицензирован на условиях GPL v3. См. [www.gnu.org/copyleft/gpl.html](http://www.gnu.org/copyleft/gpl.html)

За информацией о журналах, издаваемых Future plc group company, обращайтесь на сайт [www.futureplc.com](http://www.futureplc.com).



© Linux Format 2005

© Future Publishing Ltd 2005

BATH • LONDON • MILAN • NEW YORK • PARIS • SAN DIEGO • SAN FRANCISCO

16+

# infobox

Виртуальные серверы нового поколения



## Серверы летают!

- ✓ 3 дата-центра в России и Европе
- ✓ Высочайшая скорость Enterprise SSD
- ✓ Тройная репликация всех данных
- ✓ Безлимитный трафик
- ✓ Докризисные цены — от 200 рублей!

Закажите прямо сейчас  
[infobox.ru/vps/](https://infobox.ru/vps/)

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



## 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



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**



**MADE IN  
GERMANY**



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!

\* 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.