

DVD ДАРОМ!



LIVE-ДИСК FEDORA 22

LINUX FORMAT

У нас день рождения!

15 ЛЕТ С LINUX!

Изменили ли они мир

- » Взлет Ubuntu
- » Революция Raspberry Pi
- » Приход игр от Steam

+ Войны Init: Как Systemd победил SysV и съел Linux

Сентябрь 2015 № 9 (200)

ЮБИЛЕЙНЫЙ СПЕЦВЫПУСК!

200 ЛУЧШИХ СОВЕТОВ ПО LINUX!

- » Ускорение рабочего стола
- » Сокращения в терминале
- » Секреты сисадмина
- » Трюки с приложениями и более того!



Герои Open Source

« О, это iPhone... вон отсюда!
Мне с вами говорить не о чем. »

Линус, Столлмен, Шаттлворт, Кроа-Хартман — все на с. 38

200й НОМЕР!

64

страницы обзоров,
статей & учебников
от экспертов
в Linux



Полное руководство
Освоим Debian 8
» Раскрываем секреты
этого мощного дистрибутива



Латаем на живую нитку
Внутри ядра 4.0
» Тестируем сборку Linux
нового поколения!



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

Linux center
www.linuxcenter.ru



iTeleRadio

ИНТЕРАКТИВНОЕ РАДИО&ТЕЛЕВИДЕНИЕ

Что мы делаем

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



Кто мы

Вот вопрос, заданный нашим экспертам: что было лучшим в Linux на протяжении последних 15 лет, охваченных LXF?



Джонни Бидвелл

Запомнилось, как ввели эффекты рабочего стола и через *Compiz* стало можно включать их всякие: подвешивающие ОС, снижающие производительность и радующие глаз. Колышущиеся окна были хитом; иногда по утрам я думаю, что они вернулись. Правда, потому, что меня самого малость колышет.



Нейл Ботвик

Я знаю, мне бы надо подумать для словоизлияния нечто супер-технократическое, но по жизни лучшие моменты были связаны со встречами на конференциях, собраниях LUG и тому подобных сборищах с людьми из разных частей сообществ Linux и СПО. ПО — это замечательно, но люди лучше.



Д-р Крис Браун

Мой поворотный пункт — когда я понял, как объявить указатель на функцию, которая возвращает указатель на функцию, которая возвращает целое число. Другой интересный момент для меня и семьи Браун — я заменил на ноутбуке жены Windows на Ubuntu. И все еще жду, чтобы она это заметила.



Пол Хадсон

Мне потребовалось 15 долгих, адски переполненных лет, чтобы понять: *Nethack* у реально нет конца. Но я осознал, что Амулет Йендора — это хитроумная интернет-уловка, чтобы приковать меня к терминалу Linux на всю мою оставшуюся жизнь, и вам не удастся меня переубедить никакими доводами.



Маянк Шарма

Я скажу, что это широкие временные рамки, но если надо выбрать что-то одно, это будет появление Live-CD, а точнее — Knoppix. Этот дистрибутив, пожалуй, привлек к Linux больше людей, чем все остальные, вместе взятые. Он — и простое в использовании настольное ПО для виртуализации, типа *VirtualBox*.



Валентин Синецын

То, что из экзотики, известной в основном технарям, Linux потихоньку превратился в явление, которое на слуху и у обычных пользователей. Да, нам пока не удалось захватить рабочие столы, но со временем выяснилось, что это не так уж и нужно.



Два юбилея

» Перефразируя классика, можно утверждать, что если журнал выходит уже 10 лет, значит, он кому-нибудь интересен. Русский LXF размечал второй десяток. Для издания классического «бумажного» формата это, согласитесь, выглядит достойно. Британские коллеги подготовили юбилейный номер именно к этой дате. Пятнадцатилетие «базового» издания отмечалось еще в мае.

Если в одной фразе попытаться подвести итог, получится примерно так: свободное ПО, и как идеология, и как метод организации разработки, одержало неоспоримую победу. 10 лет назад об этом еще можно было поспорить, сегодня — уже нет. Осмелюсь утверждать, что свой вклад в этот успех внес и каждый из вас, уважаемые читатели и авторы LXF.

Хочу поблагодарить авторов, переводчиков, редакторов и всех, причастных к созданию журнала. Желаю им в следующем десятилетии еще больших творческих успехов, оригинальных идей и нестандартных решений.

Но самая главная благодарность — вам, уважаемые читатели. Спасибо за то, что вы уже много лет столь высоко оцениваете наши усилия. Спасибо за постоянную поддержку, ценные замечания и предложения.

Надеюсь, что второе десятилетие будет столь же ярким и интересным. С днем рождения, *Linux Format*!

Кирилл Степанов

Главный редактор

» info@linuxformat.ru

Как с нами связаться

Письма для публикации: letters@linuxformat.ru

Подписка и предыдущие номера: subscribe@linuxformat.ru

Техническая поддержка: answers@linuxformat.ru

Общие вопросы: info@linuxformat.ru

Проблемы с дисками: disks@linuxformat.ru

Вопросы распространения: sales@linuxformat.ru

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

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

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

Знаете всё о Linux?

ДА

НЕТ

Станьте автором
в журнале Linux Format!
linuxformat.ru/avtoram.phtml

Зарегистрируйтесь
на сайте
shop.linuxformat.ru



СОМНЕНИЯ

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



Версия для iPad
и iPhone доступна
в App Store

Выберите вид подписки

PDF-версия
на 6 месяцев

990 ₺

PDF-версия
на 12 месяцев

1800 ₺

Печатная версия
на 6 месяцев

1890 ₺

Печатная версия
на 12 месяцев

3480 ₺

Выберите вид доставки

Оплатите

Читайте Linux Format!

Станьте Linux-гуру

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

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

Печатная версия Linux Format понравится читателям, которые:

- Любят читать бумажные журналы
- Хотят получить в подарок подписку на PDF-версию Linux Format...
- ...а также диск с архивом журнала 2005–2014 гг.
- Порадуются новинкам открытого ПО на DVD-приложении к Linux Format в каждом номере

Способы доставки

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

Способы оплаты

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



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

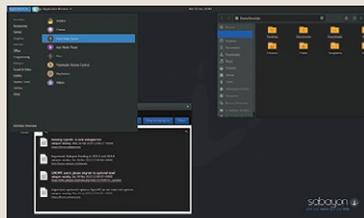
Содержание

«Цените свою свободу, иначе вы ее утратите» — Ричард Столлмен

Обзоры

Fedora 22 14
Fedora возвращается к шестимесячному циклу релизов — и вот вам прочный и надежный дистрибутив. Наслаждайтесь.

Sabayon 15.06 15
Сладкоежка: дистрибутив на базе Gentoo, но без осложнений своего родителя либо Arch Linux.



» С Sabayon вы забудете страх перед обновляемыми релизами.

Crunch Bang++ 16
Популярный дистрибутив разветвился! Ух ты! Он быстрый! Но похрустывает по-прежнему?

Acer Chromebook 15 17
Еще один хромбук для учеников и учителей, тяжеловес с убойным процессором и красивым экраном full HD.



» Такая производительность может и соблазнить, невзирая на вес.

Ultimaker 2 18
Эволюция исходной модели, которую жаль упустить. Высочайшее качество печати и бесшумная работа по разумной цене.

Kerbal Space Program 19
Мечтали заслать зеленых человечков в космос, а? Ваша мечта сбылась! Ракету постройте лично.

Knights of the Old Republic 20
Рыцари — и республика?.. Воскрешаем вселенную Звездных войн.

200 ЛУЧШИХ СОВЕТОВ ПО LINUX



Присоединяйтесь к когортам хакеров, вооружившись всякими трюками, которые начнутся со с. 30

Сравнение: Альтернативные ОС с. 24



Интервью

« Под волнением я подразумеваю, что начинаю крепко ругать людей. »

Ставшие классикой интервью с гигантами СПО с. 38

На вашем бесплатном DVD

FEDORA 22
Рабочая станция
Встроенная поддержка Docker и крутой рабочий стол GNOME

БЕСПЛАТНО №1
Ретро! Классический Linux славит свое прошлое для

UBUNTU 15.04
Веселая верстка
Получите на Ubuntu!

LINUX ЛУЧШЕЕ ПО ДЛЯ LINUX
ВСЕ, ЧТО ТРЕБУЕТСЯ ДЛЯ СТАРТА В LINUX

Fedora 22 64-битный,
Sabayon 15.06 64-битный,
Ubuntu 15.04 32-битный

» Каждый месяц — только лучшие дистрибутивы

ПЛЮС: Hotpicks, Сравнение и более того! **с. 106**

Побалуйте себя и любимых подпиской на LXF!

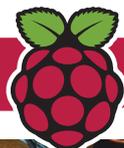


Доступно в AppStore!



www.linuxformat.ru/subscribe

Пользователям Raspberry Pi



Новости Pi 90

Piscademy едет в Америку — тренировать тысячи педагогов. Плюс, построим наладонник на Pi и порадуемся попаданию сразу двух Pi в тройку лидеров среди одноплатных компьютеров.

AstroPi SenseHAT 91

Заведите себе научную платформу сбора данных, которая метит в космос. Лес Паундер отважно вступает на доселе неизведанную обзором почву.

Играем с зуммером 92

В обычной канцелярской скрепке скрыт интересный потенциал. С ее (и Raspberry Pi) помощью Лес Паундер учит упражнять ловкость.



Ищите в номере

15/10 лет с Linux 43—56

Бойцы вспоминают минувшие дни... было ли добрее старое доброе время? Да когда как.



» Предадимся ностальгии вместе.

Откуда берутся пакеты 58

Денис Силаков по косточкам разбирает репозитории и архитектуру ферм.

Академия кодінга

Система: Процессы 82

Проведем вас за ручку по процессам ОС Linux и рассмотрим каналы с точки зрения системного программиста с широкой натурой.

PHP: Сыграем в слова 86

Не желаете кодировать? А придется. Мы выкрутили руки своему бывшему редактору, чтобы он вас заставил.



Учебники

Debian 8 Дебют в Debian 68

Разберемся, как в этом легендарном дистрибутиве решаются повседневные задачи.

Защита устройств Как уберечь ноутбук 72

Если нехорошие люди сперли ваш ноутбук, потеряно не всё. На выручку придут пароль, блокировка через Bluetooth и другие трюки.



» Блокировка по дистанции. Нет, это не минирование.

Ядро Linux 4.0 Чисть машину на ходу 74

Святой Грааль для сисадминов — динамические исправления ядра. Как они делаются и почему они не всемогущи.

IPv6 Наладим связь 78

Дни IPv4 практически сочтены. Кто же нас спасет? Оказывается, вы можете уже сейчас внедриться в сеть с IPv6, опередив толпу.

Постоянные рубрики

Новости 6 Интервью LXF 38 HotPicks 100

Индустрия ПК на спаде, ботнеты в затруднении, на подходе универсальная SIM-карта, доллар вредит продажам, формат ODF стал стандартом, СПО цветет в Испании, РОСА перешла к «Ангстрем», а США подстегивает свои суперкомпьютеры.

Притихните и внимайте: с вами говорят столпы свободного ПО — Ричард Столлмен, Линус Торвальдс, Марк Шаттлворт и Грег Кроа-Хартман.

Отвдайте горяченького! Лучшие в мире новинки свободного ПО: CVAssistant, Dust Racing 2D, Kid3, Krita, Lollypop, qBittorrent, Qt5CT, Scribus, Tanks of Freedom, Tmux, uGet.

Новости Android 22

Android ударяет автопробегом, SMS'ки способны на взлом, Суапоген теснит дуополию, а фрагментация устройств растет.

Рубрика сисадмина 62

М-р Джолион Браун вспомнил былое: «Это занимало вечность, и связь не раз обрывалась» (вот так тогда качали дистрибутивы). После чего попользовался Ansible с AWS.

Диск Linux Format 106

Содержимое двустороннего DVD этого месяца.

Сравнение 24

Не Linuxом единым живо свободное ПО: есть и замечательные альтернативы. Рассматриваем Chromium OS, Debian, GNU/Hurd, Naiku, OpenIndiana, PC-BSD.

Ответы 96

ВАШИ ПРОБЛЕМЫ РЕШЕНЫ!

Нейл Ботвик — про обновление SQL- и web-серверов, новые ядра и драйверы графического сопроцессора, начала работы с Linux, таймеры Systemd и задания Cron, домашнюю сеть, досаду от Firefox.

Пропустили номер? ... 108

Ваш шанс поползть на коленях вокруг собираемого ПК не потеряя: вы еще успеете приобрести этот журнал!

Через месяц 112

Мы заживем в новом десятилетии, а заодно и Windows 10 прибудет. Ну да Linux даст ему прикурить.



В ЭТОМ НОМЕРЕ: Что будет? » Наводят порядок » Универсальная SIM-карта » Доллар-вредитель » ODF как стандарт » СПО в Андалусии » Прощай, РОСА » Суперкомпьютеры США

БИЗНЕС

Ждем у моря погоды

Резкий спад продаж ПК вызвал снижение прибыли Intel.



» Рубрику готовил
**АНДРЕЙ
ГОНДАРЕНКОВ**

И выручка, и прибыль Intel в минувшем квартале сильно уменьшились. Основная причина — то, что в ожидании официального релиза Windows 10 пользователи не спешат с обновлением своих ПК.

По официальному заявлению Intel, доходы подразделения Client Computing Group, которое продает процессоры для настольных персональных компьютеров, ноутбуков и смартфонов, упали, по сравнению с аналогичным периодом прошлого года, на 14%, составив \$7,5 млрд. Чуть лучше ситуация в подразделении Data Center Group, производящем серверные процессоры Xeon — но и этого недостаточно, чтобы компенсировать продолжающееся падение в индустрии ПК.

Общая выручка Intel в завершившемся 27 июня квартале составила \$13,2 млрд, что на 5% ниже того же периода прошлого года. Чистая прибыль — \$2,7 млрд, на 3% меньше. При этом выручка от Data Center Group возросла на 10%, до \$3,9 млрд. Также выросли продажи на рынке Интернета Вещей, достигнув \$559 млн.

Однако большую часть своих доходов Intel получает от продажи микросхем для ПК, а этот рынок в настоящее время переживает спад. Согласно приводимым IDC данным, мировые продажи в минувшем квартале упали на 12% — отчасти из-за ожидания потенциальными покупателями официального выхода Windows 10.

По мнению исполнительного директора Intel Брайана Кржанича [Brian Krzanich],

выход Windows 10 и предстоящий выпуск процессора Intel Skylake, изготавливаемого по 14-нм техпроцессу, вызовут «волнение» на рынке ПК во второй половине года. Однако пока неясно, каким будет эффект от Windows 10, и поэтому Microsoft предлагает новую ОС в качестве бесплатного обновления существующим пользователям, снижая у них, тем самым, мотивацию к покупке новых ПК.

И еще одна новость, которую обнаружил Кржанич: Intel планирует выпустить свои первые 10-нанометровые процессоры во второй половине 2017 года. Проект получил кодовое имя Kabu Lake, он будет основан на наработках по микроархитектуре Skylake, но рассчитан на более высокую производительность.

ПРИХОТИ ВАЛЮТ

Доллар сдерживает продажи

В Q2 без подразделения серверов сегмента low-end продажи IBM упали на 13%.

По результатам II квартала IBM сообщила об общем падении выручки и прибыли, невзирая на спрос, которым пользуются ее новые мейнфреймы. Полученная выручка — \$20,8 млрд, по сравнению с \$24 млрд в прошлом году. Чистая прибыль составила \$3,45 млрд, снизившись на 16,6%. Доходы от такого гигантского сегмента IBM, как Global Technology Services, уменьшились на серьезные 10% (\$8,1 млрд).

Если бы не влияние колебаний валютного курса и продажа бизнеса серверов x86, доходы были бы выше на 1%, считают в IBM. Резко, на 32% (до \$2,06 млрд),

снизились доходы от аппаратных средств (опять же, без влияния бизнеса серверов x86 и курса валют, продажи оборудования были бы выше на 5%).

Зато положительная динамика наблюдалась в мейнфреймовом бизнесе IBM — несколько месяцев назад компания выпустила новый z13. В последнем квартале продажи мейнфреймов были выше на 9% (но на 15% меньше из-за влияния курса валют). Выручка от того, что IBM называет своими «стратегическими императивами» — это области, имеющие решающее значение для роста — выросли более чем на 20% по сравнению с прошлым годом.

» Сильный доллар отрицательно влияет на продажи IBM: ее продукция недоступна клиентам, расплачивающихся слабыми валютами



Это облачные вычисления, аналитика и «ангажированные» продукты.

Интересно, что уже не одна из самых крупных американских транснациональных корпораций отметила отрицательное влияние сильного доллара на свои продажи в 2015 г.: их продукты становятся недоступными для клиентов, расплачивающихся более слабыми валютами.

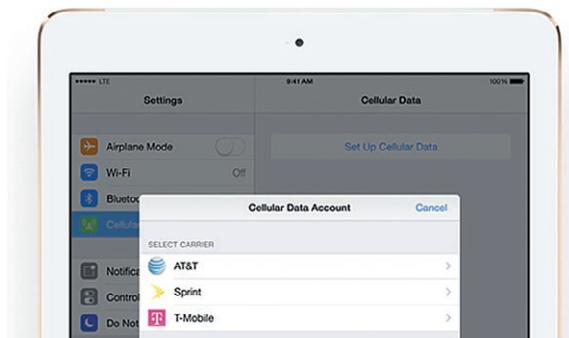
ИНТЕРЕСНЫЙ АЛЬЯНС

SIM-карта на все варианты

Apple и Samsung намерены вместе «убить» традиционные SIM-карты.

Компании Apple и Samsung — редчайший случай! — объединили усилия в попытке уничтожения традиционной SIM-карты. Сенсационную новость приводит газета Financial Times: заклятые друзья Apple и Samsung близки к достижению соглашения с ассоциацией GSMA, направленного на создание стандартизированных встроенных SIM, или e-SIM, для мобильных устройств. В отличие от традиционных SIM, e-SIM позволит пользователям смартфонов, планшетов и умных часов избежать привязки к тарифному плану единственного оператора, обеспечив свободное переключение между сотовыми сетями.

Данное решение подобно выпущенной в минувшем году универсальной SIM-карте Apple SIM. Доступная в настоящее время в Великобритании, США и Австралии, Apple SIM предоставляет пользователям iPad возможность переключать сети



➤ Выбор мобильного оператора с помощью Apple SIM.

без необходимости хлопотной замены SIM-карты. GSMA подтвердила газете наличие таких планов, отметив при этом, что новое SIM-решение вряд ли будет доступно на устройствах ранее 2016 г.

В числе мобильных операторов, которые, как ожидается, поддержат инициативу, названы AT&T, Deutsche Telekom, Etisalat, Hutchison Whampoa, Telefónica, Orange и Vo-

dafone. GSMA также подтвердила, что компания Apple вовлечена в эти планы: «продолжается работа с Apple, направленная на получение ее гарантий по поддержке данной инициативы. Мы преисполнены оптимизма, однако официальное соглашение еще не заключено».

Энн Буvero [Anne Bouverot], генеральный директор ассоциации GSMA, добавила, что в настоящее время мобильные телефоны есть у приблизительно 3,7 млрд человек — а это около половины населения земного шара; однако подключение оставшейся части является огромной проблемой для отрасли.

Появление на рынке стандартизированного SIM-решения, способного коренным образом изменить наш подход к приобретению мобильных телефонов и пользованию услугами того или иного оператора, является хорошим шагом к решению этой проблемы.

ЗА ЧИСТОТУ СЕТИ

Нет — ботнетам

Большим спамерским ботнетам стало труднее работать.

В июне спам составил менее 50% всей электронной почты, что является самым низким показателем за последнее десятилетие. Такую информацию приводит Symantec в очередном аналитическом отчете Intelligence Report. В июне Symantec зафиксировал 704 млрд отправленных электронных писем, из которых только 353 млрд были классифицированы как спам, тогда как на одном из пиков эпидемии спама в июне 2009 г. спамом было 5,7 трлн из 6,3 трлн сообщений.

Уровень спама неуклонно, хотя и медленно, снижается с 2010 г. по множеству причин. Поставщики сетевых услуг настроились на решение проблем и быстрее принимают меры, выявляя таковые в своих сервисах. Кроме того, в отличие от ситуации 6- или 7-летней давности, отправлять миллиарды сообщений в день стало крупным ботнетам много сложнее.

Правоохранительные органы, совместно с Microsoft и другими компаниями, в последние годы настойчиво выявляли

крупнейшие бот-сети и добивались их технического закрытия. И хотя в ряде случаев операторы бот-сетей смогли восстановить над ними контроль, повышенное внимание значительно затрудняет их дальнейшую деятельность.

Улучшенная фильтрация и блокировка снижают количество достигающих почтовых ящиков нежелательных маркетинговых

«Отправлять миллиарды сообщений с ботнетов стало труднее.»

сообщений. Вероятность ответа на спам, как известно, мала, поэтому спамерам для построения бизнеса необходима значительная база почтовых адресов. Это вовсе не значит, что спам собирается полностью остановиться, но поскольку затраты растут, а отдача падает, снижаются стимулы ведения такого бизнеса.

Специалисты Symantec также отметили, что в июне значительно сократились атаки с использованием фишинга и вредоносного программного обеспечения, пересылаемого через электронную почту. По их мнению, это является доказательством того, что «атакующие просто принялись перенаправлять усилия в другие области угроз».



➤ Из отправленных в июне 704 млрд электронных писем спамом были только 353 млрд, в то время как в июне 2009 г. спамом являлись 5,7 трлн из 6,3 трлн отправленных сообщений.

НОВЫЙ СТАНДАРТ

Принятие открытого формата

ODF 1.2 опубликован в качестве международного стандарта ISO/IEC.

В середине июля спецификация Open Document Format (ODF) v1.2 опубликована в качестве международного стандарта 26300:2015 ISO/IEC. Свою популярность формат ODF снискал благодаря широкому распространению пакетов *OpenOffice/LibreOffice*, а также помощи, оказываемой им в борьбе против засилия проприетарных типов файлов *Microsoft Office* — DOC и DOCX. Такой шаг приветствуется сообществом Open Source, поскольку большее количество документов будет должным образом отображаться предустановленным в дистрибутивах GNU/Linux пакетом *LibreOffice*, а *Microsoft Office* будет в меньшей степени удерживать пользователей от перехода на Linux.

Спецификация ODF состоит из трех частей: Общая схема ODF, OpenFormula (фор-



мулы для электронных таблиц) и Модель упаковки данных в ODF-контейнер. «ODF 1.2 — это „родной“ формат файлов *LibreOffice*. За счет своей широкой адаптивности к приложениям в различных областях, на сегодня ODF является лучшим выбором для обеспечения функциональной совместимости, — уверяет Торстен Беренс [Thorsten Behrens], председатель TDF (The Document Foundation). — ODF

➤ На сегодня ODF — лучший выбор для обеспечения функциональной совместимости.

обеспечивает реальную функциональную совместимость и превращает в пережиток прошлого необходимость использования проприетарных форматов документов. В будущем люди будут вспоминать о несовместимых форматах документов между двумя выпусками фирменных офисных пакетов как о давним-давно решенной проблеме».

В прошлом году британское правительство приняло ODF в качестве рекомендованного для государственного документооборота, что позволило сократить издержки на лицензиях *Microsoft Office* и сделать правительство более «прозрачным». В настоящее время ODF предложен в качестве рекомендованного стандарта *Référentiel Général d'Interopérabilité 1.9.9* правительством Франции.

СПО НА ПЛАНЕТЕ

У нас, в Испании!

Андалусия обновляет свободное ПО для управления рабочими станциями.

Правительство Андалусии обновило GECOS — свободное ПО с открытым исходным кодом, которое должно стать стандартной конфигурацией на рабочих станциях и ноутбуках этого автономного сообщества Испании. Изменения затронули возможности IT-менеджмента, системной интеграции, поддержки и масштабируемости. Новая версия стандартной корпоративной настольной системы, GECOS (Guadalínex Escritorio Corporativo eStandar), предназначена для управления десятками тысяч рабочих мест.

«Главная цель, которую мы преследуем — сделать GECOS государственным стандартом, — говорит Хуан Конде [Juan Conde], работающий в проекте по продвижению СПО Министерства финансов и государственного управления Андалусии. — Первая версия GECOS, представленная в 2012 г., позволила определить предельную величину централизованно управляемой инфраструктуры рабочих мест на Linux. Она хорошо работала на десятках компьютеров, однако ее инструменты управления не допускали расширения

до сотен или тысяч ПК. Вторая версия GECOS будет способна управлять десятками тысяч рабочих мест. Сейчас она находится в пилотной фазе, а затем будет задействована для управления около 2 тыс. рабочих станций».

В основе рабочих мест GECOS лежат Ubuntu 14.04 и Linux Mint 17. Новая версия GECOS допускает организацию иерархии рабочих станций, принтеров и устройств хранения данных, а также администрирование пользователей, групп пользовате-



➤ Новая версия GECOS позволяет управлять десятками тысяч рабочих мест.

«Постановление парламента призвало регион к переходу на СПО.»

лей и приложений. Встроенные в GECOS средства управления всесторонне протестированы в крупномасштабных ЦОД. Основной инструмент управления — *Chef*, который также используется одним известным интернет-ритейлером. Для создания web-интерфейсов управления выбраны библиотеки Javascript (*Backbone.js*

и *Marionette.js*). Интерфейсы служат для управления иерархиями, а все конфигурации и задачи хранятся в базе данных *MongoDB*.

Разработке GECOS предшествовало постановление парламента Андалусии, которое призвало регион к переходу на свободное и открытое ПО. Кроме того, местный парламент продолжает поддерживать Guadalínex — настольный дистрибутив Open Source, предназначенный для использования широкими слоями местного населения, в том числе и в школах.

Сегодня в Андалусии под управлением Guadalínex работают приблизительно 400 тыс. школьных ноутбуков.

7 советов быстрого поиска работы от hh.ru

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

И помните, что работа найдется для каждого!

ДИСТРИБУТИВЫ

Лебединая песня «Росы»

Вышел ROSA Desktop Fresh KDE R6 — последний под прежним флагом.

В конце июля компания «Роса» представила обновление своего дистрибутива семейства ROSA Desktop Fresh с рабочим столом KDE. Дистрибутив подойдет и для развлечения, и для работы. В его репозиториях пользователей ждет большая коллекция игр и эмуляторов (можно установить *Steam*), и привычный софт для текстовой и видеокommunikации, включая Skype. Работает просмотр видео всех видов и форматов. В дистрибутив входит свежий *LibreOffice*, но в репозиториях есть полный TeX-стек и все известные open-source программы для документирования, верстки и полиграфии. Разработчиков ждут полноценные LAMP/C++/... окружения для разработки.

А буквально через несколько дней стало известно, что контроль над ИТЦ ИТ «РОСА» переходит к производителю микроэлектроники НПО «Ангстрем», принадлежащему бывшему министру связи Леониду Рейма-

ну. Об этом сообщил генеральный директор «Ангстрема» Алексей Бочаров, а генеральный директор ИТЦ ИТ РОСА Аркадий Тагиев подтвердил сообщение. Сделка пока структурируется; на это, по мнению Бочарова, уйдет более месяца. Сумма сделки не раскрывается. Назвав «РОСу» «отличным активом», Бочаров добавил, что компания «будет жить».

В июне «РОСА» совместно с «Альт Линукс» выиграла конкурс Минкомсвязи на создание отечественной ОС. Однако, собрав сильную техническую команду, компания не уделила внимания коммерческим продажам, а государство ей также не помогало. При этом у РОСы есть хорошие продукты, но коммерциализировать их непросто: значительная доля спроса на них должна исходить от госорганов, которые, однако, предпочитают либо самостоятельные разработки, либо закупки готового коммерческого ПО.

СУПЕРКОМПЬЮТЕРЫ

Цель — на рывок

США планируют резко ускорить свои суперкомпьютеры.

Президент США Барак Обама издал распоряжение о запуске Национальной стратегической компьютерной инициативы [National Strategic Computing Initiative] по координации действий госучреждений, учебных заведений и частного сектора США в сфере высокопроизводительных вычислений (HPC). NSCI должна ускорить создание супер-систем с производительностью порядка одного экзафлопа (exaFLOP, квинтиллион операций с плавающей запятой в секунду).

Уже более двух лет в мировой «гонки суперкомпьютеров» лидирует Tianhe-2, разработанный в Национальном университете оборонных технологий Китая. Похоже, США намерены исправить ситуацию.

Одна из целей NSCI — выработка курса на следующие 15 лет, жизнеспособного и при достижении предела текущей полупроводниковой технологии, что означает

конец эры закона Гордона Мура. Тремя лидерами NSCI будут Министерства энергетики и обороны и Национальный научный фонд. Каждый из них фокусируется на различных областях HPC, сотрудничая с Агентством передовых исследований в сфере разведки (IARPA) и Национальным институтом стандартов и технологий (NIST): IARPA займется альтернативами стандартным полупроводниковым технологиям, а NIST — метрологией.

Конкретные сроки не названы, но Министерство энергетики уже заявило о планах разработки и поставки экзафлоповых вычислительных систем, в сотни и тысячи раз превосходящих текущие возможности, к 2023 или 2024 г. В 2014 г. экспертная группа выдала заключение, что для создания экзафлопового суперкомпьютера потребуются \$3 млрд дополнительных инвестиций и срок более 10 лет. **LXF**

Новости короткой строкой

» Пользователи недовольны *Ubuntu Software Center*, и разработчики Ubuntu MATE уберут его из выпуска 15.10. Источник: ubuntu-mate.org

» Разработка среды KDE для Solaris (OpenSolaris, OpenIndiana, Illumos) прекращена: нет разработчиков. Источник: euroquis.nl

» «МультиКарта» и «Альт Линукс» внедрили ПО автоматизации управления сетями терминалов МультиКарты на базе технологии BPM. Источник: www.altlinux.ru

» *LibreOffice* стал 5.0 благодаря реформам его вида, выпуску первого 64-битного релиза для Windows, сообщениям маркетинга и 5-й годовщине The Document Foundation. Источник: blog.documentfoundation.org

» Курируемая Linux Foundation организация The Core Infrastructure Initiative стала первым платиновым спонсором проекта OpenBSD в 2015 г. Источник: undeadly.org

» В ООО «ЛИССИ-Софт» подготовили новые версии *Mozilla Firefox 38* и *Thunderbird 38* для платформ Linux и Windows с поддержкой ГОСТов российской криптографии. Источник: soft.lissi.ru

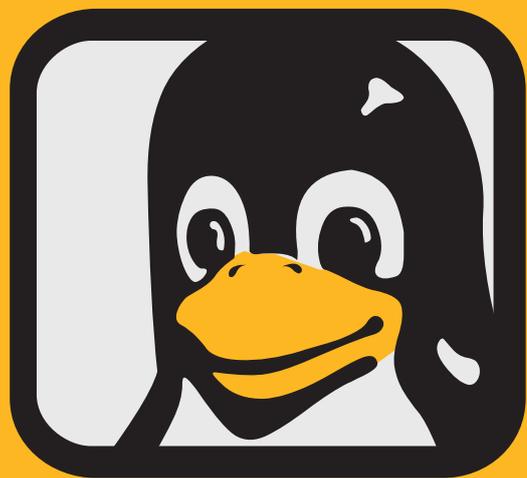
» Около 70% федеральных информационных систем РФ используют в качестве базового ПО закрытые продукты Microsoft и Oracle. Источник: www.tadviser.ru

» Правозащитная организация EFF представила Privacy Badger 1.0 — блокировку трекинга переходов пользователя по сайтам, для *Firefox* и *Chrome*. Источник: [www. EFF.org](http://www EFF.org)

» В Корнельском университете (США) разработали технику атаки на уязвимость RowHammer в чипах памяти DRAM: достаточно запустить код на JavaScript с любого сайта. Источник: arstechnica.com

» Проблемы с применением OpenSSL в GPL-продуктах сняты переходом на типовую лицензию Apache 2.0. Источник: www.openssl.org

» Появился метод идентификации пользователей браузера, анализом ввода с клавиатуры JavaScript-кодом на сайте, даже в случае *Tor Browser*. Источник: www.opennet.ru



LINUXPITER

INTERNATIONAL
LINUX CONFERENCE

ST.PETERSBURG

21. 11. 2015

TOPICS:

NETWORK

SERVER AND STORAGE

VIRTUALIZATION

CLOUD

www.LinuxPiter.ru

Приглашаем детей на фестиваль
программирования и робототехники!

Робот Фест



18 октября 2015, с 12⁰⁰ до 20⁰⁰

Санкт-Петербург, Политехническая ул., 29 (ФабЛаб Политех)

РобоМарафон ×4

12⁰⁰, 14⁰⁰, 16⁰⁰, 18⁰⁰

На РобоМарафоне дети смогут за час изучить основы программирования при помощи робота, а затем принять участие в увлекательных соревнованиях с роботами.

Мейкертон

16⁰⁰

На Мейкертоне дети будут учиться азам робототехники и собирать своего первого робота с нуля.



ScratchDuino

WWW.SCRATCHDUINO.RU

(812) 309 06 86



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

Десять лет — не один день

Истина в заглавии вполне себе банальна, но от этого не перестает быть истинной: ровно 10 лет назад, в сентябре 2005 г., вышел из печати первый номер русскоязычной версии журнала *Linux Format* (кодовое имя, **LXF70**, унаследовано от исходной, английской). В столь знаменательный, что называется, день принято, оглядываясь на пройденный путь, говорить о свершениях и достижениях. Позволю себе нарушить эту традицию, по крайней мере, частично. И поговорить не только о том, что было сделано, но и о том, чего сделано не было. Для чего, опираясь на материалы номеров **LXF70—71**, вспомним освещенные в них наиболее значимые события осени 2005 г:

- » стабилизация дистрибутива Mandrake, отразившаяся в объединении его с Connectiva под именем Mandriva;
- » превращение Ubuntu, в лице 5.10 Breezy, в полноценную систему, закрепленное «долгоиграющим» статусом релиза следующего;
- » выход, после 3-х лет разработки, дистрибутива Debian 3.1 Sarge, с обещанием кодеров ускорить релизы.

Конечно, это не значит, что прочие тогдашние события были не важны. Ведь почти тогда же происходили оформление openSUSE и стабилизация Fedora (уже без Core). Но именно эти три вселяли надежды, что «десктопизация» Linux'a, бурно, хотя и с переменным успехом шедшая в прошлую пятилетку, продолжится столь же интенсивно, и притом успешнее. Оправдались ли надежды, и если да, то насколько? Об этом — см. стр. 52 в данном номере.
alv@posix.ru

Сегодня мы рассматриваем:

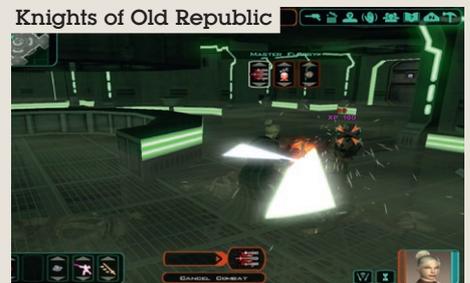
- | | |
|---|---|
| Fedora 22 14 | Acer Chromebook 15 17 |
| Дистрибутив, снимавший славу авангардного, отныне один в трех ипостасях: настольной, серверной и облачной. Мы рассмотрим первую из них. | Если вы собираетесь не разрезать с этим хромбуком по командировкам, а учиться за партой, то останетесь довольны его мощным процессором. |
| Sabayon 15.06 15 | Ultimaker 2 18 |
| Самообновляющиеся дистрибутивы внушают одновременно и уважение, и трепет. Но данный конкретный экземпляр очень дружелюбен к новичкам. | 3D-печать неуклонно совершенствуется: из полкустарной модели вытупился отличный принтер, работающий бесшумно и с высоким качеством. |
| Crunch Bang++ 16 | Kerbal Space Program 19 |
| CrunchBang вроде и умер, но дело его живет! От него ответвился самостоятельный дистрибутив, превосходящий прародителя как минимум на два плюса. | Наконец-то все желающие получили возможность строить ракеты в открытом космосе. Зеленые человечки вам в помощь. |
| | Knights of Old Republic 20 |
| | Узнаете знакомый антураж Звездных войн? Злобные враги и живой мир с хорошо прописанной историей не дадут вам заскучать. |



» Ноутбук понравится ученикам и учителям.



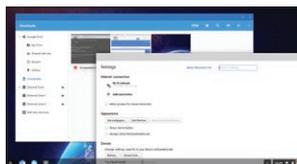
» Fedora становится дружелюбнее к новичкам, но и не отказывается от авангардных приложений.



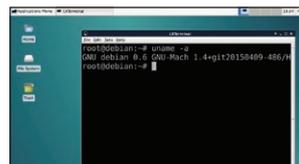
» Лучше одного светового меча — только два световых меча! Враг разлетается на осколки.

Сравнение: Альтернативные ОС с. 24

Chromium OS



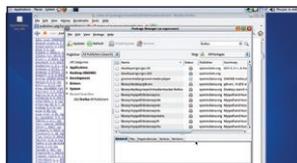
Debian GNU/Hurd



Haiku



OpenIndiana



PC-BSD



Есть ли жизнь без Linux? Ответ — да, есть, и довольно-таки интенсивная. Наша любимая ОС на этой планете не одинока: у нее хватает достойных соратников. Мы рассматриваем лучших.

Fedora 22

Тестируя свежий релиз популярного дистрибутива на Gnome, **Шашанк Шарма** выявил, что тот готов отказаться от имиджа чисто для продвинутых.

Вкратце

» Последнее издание поддерживаемого Red Hat, но разрабатываемого сообществом дистрибутива. Доступно в трех редакциях; обычный настольный релиз называется Workstation. Дистрибутив продолжает вводить изменения, чтобы привлечь новых пользователей и при этом удовлетворять опытных. См. также: openSUSE и Korora.

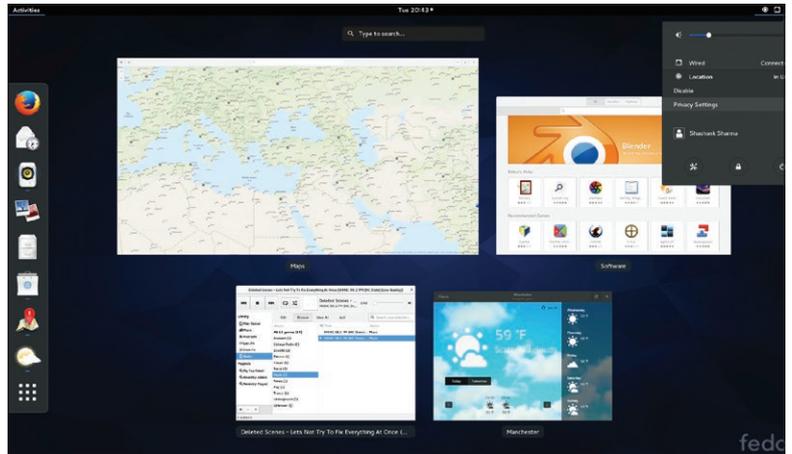
Для своего последнего релиза Fedora нарушила устоявшийся шестимесячный цикл релизов, чтобы внести ряд организационных и системных изменений. Самым главным был выход трех разных редакций: Workstation, Cloud и Server. Еще одним основным изменением был анонс Fedora.next, это название дорожной карты Fedora Project на следующее десятилетие. С новым релизом дистрибутив возвращается к шестимесячному циклу, но продолжает выпускать три редакции. Каждая редакция строится на общей базе, а затем добавляются необходимые пакеты, чтобы привести издание в соответствие с выбранной функцией.

Мы порешили рассмотреть редакцию Workstation, доступную в виде ISO-образа для 32- и 64-битных компьютеров. Это редакция live с возможностью установки, разработанная для домашних пользователей, но прекрасно подходящая для профессионалов и быстро превращающаяся в раздолье для разработчиков.

Передовые программы

В качестве тестового полигона для Red Hat Enterprise Linux (RHEL), у Fedora есть уникальная возможность предоставлять новым пользователям передовые технологии. Это всегда были дистрибутивы, поставляемые с самым свежим ПО, и данный релиз — не исключение.

Gnome Display Manager (GDM) отныне вместо *X.org* по умолчанию предлагает сервер отображения *Wayland*. Хотя сессия Gnome по умолчанию по-прежнему использует *X*, следующий релиз Fedora, скорее всего, по умолчанию примет *Wayland*.



» Fedora 22 стоит на прочной основе предыдущего релиза и предлагает новые функции и полезные улучшения.

Дистрибутив оснащен Gnome 3.16, получившим немало визуальных улучшений и новых функций. Область уведомлений подверглась пересмотру, и уведомления теперь привязаны к центру верхней панели. Ненавязчивый указатель информирует вас обо всех непрочитанных уведомлениях, таких как задачи фонового терминала, сообщения чата, обновления, и т. д.

Приложение *Files*, экран приглашения, обзоры *Activities* и другие части рабочего стола теперь выглядят по-новому. *Voxes*, приложение по умолчанию для управления функциями виртуальных и удаленных машин, предлагает, помимо всего прочего, переосмысленный ассистент создания окон и различные улучшения диалогов *Preferences* [Параметры].

Приложение *Software*, прошедшее долгий путь с его появления в Fedora 20, помимо установки обновлений и поиска нового ПО сейчас также позволяет устанавливать дополнения, такие как шрифты и кодеки. Это очень надежный инструмент, не уступающий в дружелюбии *Ubuntu Software Center*, и столь же быстрый и эффективный.

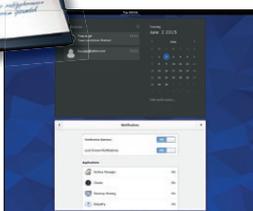
Fedora 22 — первый релиз, не использующий *Yum* в качестве менеджера пакетов. Вместо него применяется *DNF* — ответвление *Yum*, предлагающее схожую функциональность и использующее те же репозитории пакетов RPM, что и всегда. *DNF* использует улучшенную программу разрешения зависимостей, *hawkey*, и поскольку *DNF* совместим с *Yum* в командной строке, большинство команд *Yum*, типа *install*, *groupinstall*, *remove* и т. д., работают и в *DNF*.

Fedora часто считают не самой идеальной для новых пользователей, но улучшенная программа установки *Anaconda* весьма дружелюбна. Мы заметили, что она, хоть и не особенно изменилось по сравнению с последним релизом, кажется немного тормозной. Расположение клавиш вверху программы установки продолжает вызывать головокружение, но сама программа функционирует не хуже, чем у других дистрибутивов.

Существующие пользователи Fedora по достоинству оценят различные визуальные улучшения и добавленные функции последнего релиза. Для тех, кто уже использует RPM-дистрибутив, Fedora 22 достаточно надежна, стабильна и быстра, чтобы переманить вас с вашего текущего дистрибутива. **LXF**



Свойства навскидку



Уведомления

Всплывающие окна под названием *Notifiers* теперь переместились на верх экрана, став более заметными.



Менеджер пакетов

Dandified Yum, сокращенный до *DNF*, является ответвлением *Yum* и менеджером пакетов по умолчанию.

LINUX Вердикт
FORMAT

Fedora 22 Workstation

Разработчик: Fedora Project
 Сайт: www.fedoraproject.org
 Лицензия: Разные

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

» Солидное ПО для офиса, Интернета и мультимедиа; дружелюбные инструменты делают его универсальным дистрибутивом.

Рейтинг 9/10

Sabayon 15.06

Шашанк Шарма тестирует последний релиз дистрибутива, названного в честь итальянского десерта — захочет ли он добавки?

Вкратце

» Основанный на Gentoo, Sabayon поставляется в виде live-образа и образа установки, наравне с Fedora и Ubuntu. Этот дистрибутив с плавающим обновлением доступен только в 64-битных ISO, но для разных рабочих окружений, в том числе Gnome, KDE и Xfce. Последний релиз посвящен оформлению и внешнему виду рабочего стола. См. также: Fedora, Ubuntu.

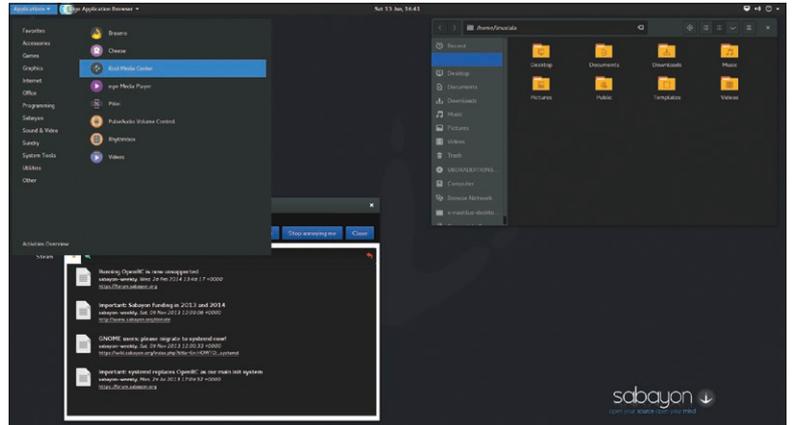
Как подлинный наследник Gentoo, Sabayon 15.06 Gnome демонстрирует невероятную скорость, особенно в сравнении со своими более именитыми соперниками. Установка Gentoo иногда может затянуться на много дней, в зависимости от настроек оборудования, а стремление Sabayon обеспечить готовую к работе настольную систему прямо из коробки делает его идеальным для тех, кому нужен дистрибутив с непрерывным циклом обновления, не требующий хлопот.

Sabayon обеспечивает идеальное сочетание передовых функций, таких как плавающие релизы на базе исходников [source based], с простым устанавливаемым live-образом и с комплектом исполняемых файлов. Помимо облегченной в плане графики версии с рабочим столом *Openbox*, дистрибутив предлагает 64-битные образы для Gnome, KDE и Xfce.

Дружелюбный

Дистрибутив желает быть востребованным и стремится угодить пользователям. Так что приложения здесь собраны по категориям, таким как мультимедиа или Интернет; имеется обширная коллекция полезных дополнений. Однако, несмотря на «вес» в 1,7 ГБ, в версии Gnome нет ни офисного пакета, ни даже текстового редактора и некоторых других традиционных приложений, например, *VLC*. И что еще хуже, несколько приложений, таких как *Videos* и *Steam*, не запускаются, а дистрибутив даже не выдает отчет о падении или какой-либо ошибке.

Компенсацией за отсутствующие приложения становится впечатляющий менеджер пакетов, *Rigo*. Хотя внешне *Rigo*



» Несмотря на отсутствие ряда важных приложений и кое-какие огрехи, этот дистрибутив излечит новичков от боязни плавающих обновлений.

не имеет ничего общего с *Ubuntu Software Center* или *Synaptic*, это мощное приложение. Помимо поиска и настройки дополнительных пакетов, *Rigo* умеет устанавливать обновления и настраивать репозитории. *Rigo* также служит своеобразной доской объявлений, информируя вас об известиях сообщества.

Помимо *Rigo*, который подключается к программным репозиториям Sabayon, содержащим двоичные пакеты, можно также использовать репозитории Gentoo для установки исходников на Sabayon с помощью *Portage*. Благодаря этой уникальной функции, у пользователей есть выбор между исходником и двоичным дистрибутивом.

В дистрибутиве используется установщик *Anaconda*, но, в отличие от последних версий Fedora, в установщике Sabayon кнопки расположены логично. По сравнению с другими дистрибутивами установка идет довольно медленно, и хотя ваш часовой пояс дистрибутив определит автоматически, традиционной процедуры выбора пакетов для установки здесь нет. Вы также можете в live-режиме выделить место для дистрибутива, используя *GParted*, если система разметки *Anaconda* вам не по душе. На наших тестовых машинах установка аварийно завершилась при выборе раздела LVM, однако на форуме по этому поводу ничего нет, так что, возможно, это разовый сбой, а не крупная ошибка. Также индикатор загрузки во время установки замирает на середине, хотя установка продолжается.

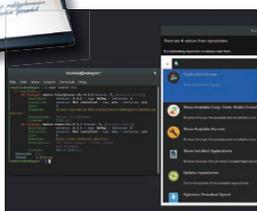
Дистрибутив нарочито скрывает свое родство с Gnome. Меню приложений в рабочем столе расположено в верхнем левом

углу экрана и представляет собой разбитый по категориям список имеющихся приложений. Можете нажать на кнопку *Activities Overview* [Режим обзора] внизу, чтобы стол принял привычный вид Gnome 3. В составе дистрибутива также предусмотрены драйверы Nvidia и AMD GPU, ядро 4.0, *Kodi Media Center*, *Chrome* и различные другие Интернет- и мультимедиа-приложения с поддержкой различных открытых источников и собственных форматов прямо из коробки.

Sabayon похож на Arch или Gentoo, но гораздо проще в установке. Последний релиз выглядит под стать любому современному дистрибутиву, хотя можно было бы его побольше потестировать. Если вы ищете настраиваемый, но простой в установке многофункциональный дистрибутив — рекомендуем. **LXF**



Свойства навскидку



Двоичный или исходный
Альтернативы в виде *Entory* и *Portage* дают доступ к одному из богатейших репозиториев.



Тонкая настройка
При наличии времени, можно с помощью *Portage* сделать из Sabayon идеальную машину на Gentoo.

LINUX **Вердикт**
FORMAT

Sabayon 15.06 Gnome

Разработчик: Фабьо Эркулани (Fabio Erculiani) и др.
Сайт: www.sabayon.org
Лицензия: GPL и др.

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

» Прекрасный выбор для тех, кто хочет перейти на Gentoo, но опасается сложностей.

Рейтинг 8/10

CrunchBang++ 1.0

Памятуя часы, отданные кодированию, **Шашанк Шарма** проверяет, сколько ли в CrunchBang++ плюсов, сколько заявлено в названии.

Вкратце

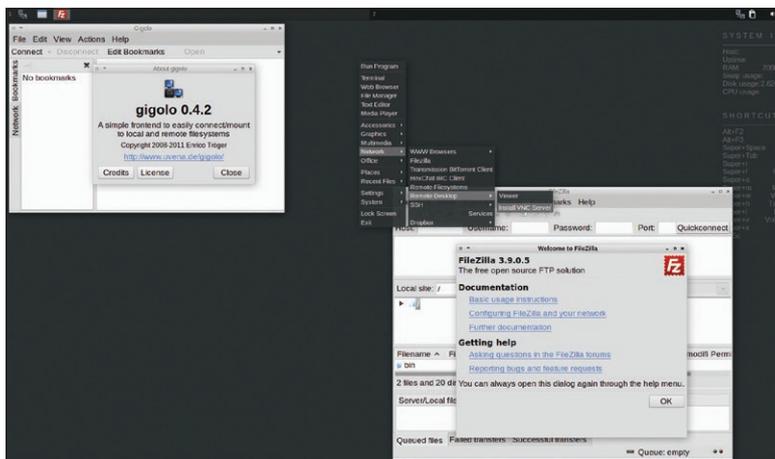
» Также известный как СВРР или #!++, этот дистрибутив на базе Debian 8 имеет легкий оконный менеджер *OpenVox* и предлагает хороший баланс скорости и функциональности. В состав входят популярные приложения, такие как *VLC* и *GIMP*, вместе с облегченными альтернативами для большинства других. См. также: *Puppy Linux*, *Porteus*, *Slitaz*.

Проекты Linux приходят и уходят. По тем или иным причинам угасло немало дистрибутивов с процветающими сообществами. И тем не менее, объявление о кончине CrunchBang вызвало шок. Но благодаря упорству и воле авторов этого проекта на свет появился его преемник, известный как Bunsen Labs. Несмотря на то, что в нем не осталось ни одного разработчика из CrunchBang, CrunchBang++ (#!++) является попыткой продолжить наследие одного из самых любимых дистрибутивов в духе минимализма.

#!++ основан на Debian 8 и использует его установщик. Еще несколько лет назад кое-кого это бы остановило, но с последними релизами установщик Debian становится все дружелюбнее к пользователю. Однако в live-режиме дистрибутив запустить нельзя, и это минус для новичков, желающих сначала понять, стоит ли утруждаться с установкой; зато устанавливаемые образы доступны как для 32-, так и 64-битных машин.

Еще Bang?

После установки дистрибутива вы попадаете на минималистский рабочий стол на базе *OpenVox*, и автоматически запускается послеустановочный скрипт конфигурации, который проведет вас через дополнительные параметры настройки, такие как установка поддержки принтера или Java Runtime Environment, *LibreOffice*, стека LAMP и т. д. Обязательным данный 12-шаговый скрипт не является, поскольку требует соединения с Интернетом, и вы можете запустить его в любое время из терминала, набрав `cbpp-welcome`.



» Внешне неказистый CrunchBang++ — отлично укомплектованный дистрибутив, легко превращаемый в богатую функциями настольную систему или сервер LAMP.

В рабочем столе нет специального меню приложений — в стиле *OpenVox*, оно отображается по щелчку правой кнопки мыши. Несмотря на значительную коллекцию приложений по умолчанию, таких как браузер *Iceweasel*, медиа-плеер *VLC*, *GIMP*, *AbiWord*, просмотрщик документов *Atril*, *Transmission*, *Filezilla* и т. д., более многофункциональные и привычные приложения, вроде *LibreOffice* и *Chrome*, отсутствуют. Но есть скрипт для установки этих важных приложений, что избавляет вас от необходимости вручную устанавливать их через графический менеджер пакетов *Synaptic* или инструмент *apt-get*. Что поразительно, дистрибутив также включает инсталляторы для *DropBox* и сервера VNC.

Если вы привыкли к любому из ныне популярных пафосных дистрибутивов, скромное оформление в серых тонах навевает на вас тоску. Но *OpenVox* — штука очень гибкая: при желании вы можете изменить почти каждый элемент на рабочем столе, сделав его более красочным.

Помимо необходимой статистики, системный монитор *Conky*, находящийся в правом углу рабочего стола, также отображает ряд горячих клавиш по умолчанию, например, для запуска файлового менеджера, терминала, медиа-плеера и т. д. Учтите: стандартные ссылки используют клавишу Super, и если ее у вас нет, понадобится перенастройка.

Хотя изначально в дистрибутиве только два инструмента мультимедиа — *VLC* и *Xfburn* для записи дисков, «из коробки» он поддерживает множество форматов,

а репозиторий с несвободными компонентами Debian подключен по умолчанию.

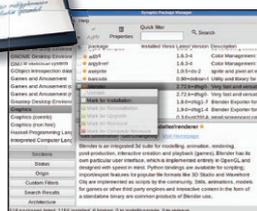
Если вы проводите много времени в командной строке, то скоро обнаружите, что команды вроде `ifconfig` могут быть запущены только с правами суперпользователя. Это потому, что `/sbin` не входит в `$PATH`. Проблему решает команда

```
export PATH=$PATH:/sbin
```

Есть много легких дистрибутивов, сменивших KDE или Gnome на *Xfce* или *LXDE*. И море таких, где вообще нет рабочего стола, а графический интерфейс является надстройкой к оконному менеджеру, например, *OpenVox*. Хотя #!++ определенно попадает во вторую категорию, в этом дистрибутиве гораздо больше вложились в создание очень быстрой системы, экономной в плане ресурсов. **LXF**

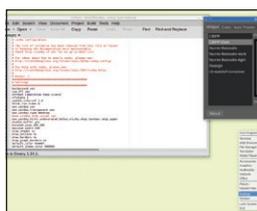


Свойства навскидку



На базе Debian

Основанный на Debian, он невероятно стабилен. Плюс, быстрота и легкая настройка для любых целей.



Максимум гибкости

Работает на легкоуправляемом *OpenVox*; ему помогают монитор системы *Conky* и менеджер файлов *Thunar*.

LINUX FORMAT Вердикт

CrunchBang++ 1.0

Разработчик: Бен Янг [Ben Young]
Сайт: www.crunchbangplusplus.org
Лицензия: Разные

Функциональность	8/10
Производительность	9/10
Удобство в работе	9/10
Документация	5/10

» Молодой дистрибутив, у которого пока даже нет форума (они используют Reddit), но имя уже есть.

Рейтинг 8/10

Acer Chromebook 15

C910 54M1

Это самый большой Chromebook, который когда-либо видел **Хуан Мартинес**, но всё ли решает размер?

Спецификация

- » **CPU** Dual-core 2,2 ГГц Intel Core i5-5200U
- » **Видеокарта** Intel HD Graphics 5500
- » **ОЗУ** 4 Гб, DDR3L SDRAM
- » **Экран** 15,6-дюймовый full HD (1920×1080)
- » **Диск** 32 Гб SSD
- » **Порты** HDMI, 1×USB 2.0, 1×USB 3.0
- » **Связь** Wi-Fi 802.11a/b/g/n/ac
- » **Камера** 720p HD
- » **Габариты** 1,0×15,1×10,1 дюйма (в миллиметры сами переведете :)
- » **Вес** 2,19 кг

По сравнению с большинством других портативных ПК на ChromeOS, у Acer Chromebook 15 (C91054M1) больший экран и бóльшая скорость обработки данных, что ведет к бóльшему ценнику. Рассчитанный, в частности, на студентов и учителей, благодаря своей надежной конструкции и превосходному дизайну Acer подходит для любого потребителя, согласного таскать с собой несколько лишних фунтов и дюймов.

Размер — первое, что вы заметите в этой модели, которая потянет на 2,19 кг. Этот увесистый ноутбук не слишком подходит для людей, путешествующих с места на место. Зато он идеален для студентов и учителей, проводящих много времени за столом. Облагоченный в великолепный черный корпус с матовой отделкой, C910 немедленно делает стильным любое рабочее пространство.

15,6-дюймовый экран ноутбука (full HD, разрешение 1920×1080) прекрасен. Вы сможете длительное время наслаждаться фильмами, просматривать сайты и играть в игры, не ощущая напряжения глаз. А любители смотреть фильмы компанией оценят углы обзора, которые позволяют троем людям на расстоянии пяти футов друг от друга видеть экран без затенений по краям.

На нормальные для ноутбука, но не самые расчудесные колонки, направленные вверх, выделено около трех дюймов клавиатурной панели шириной в 15,1 дюйма.

» Не столь портативный, как некоторые хромбуки, но более мощный.



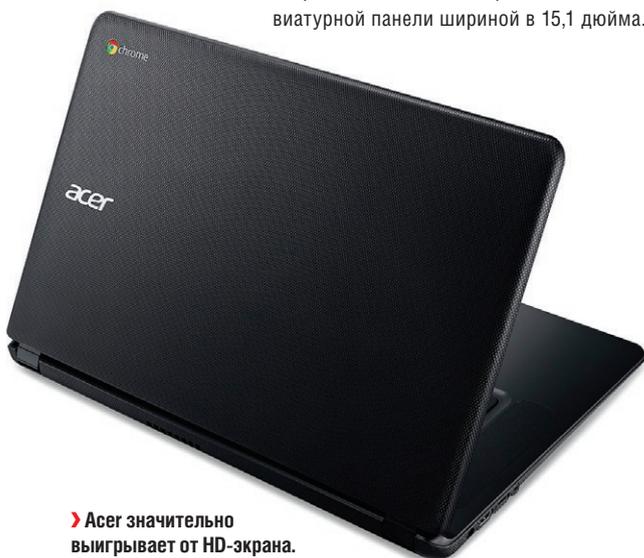
Тачпад, кажущийся пустым и скрипучим, занимает три дюйма, а клавиатура — чуть более пяти. Это несколько затрудняет управление, особенно для людей с маленькими пальцами и скромным опытом печатания (скажем, детей). Acer стоило бы увеличить клавиатуру и клавиши, уменьшив ширину колонок и вполтину уменьшив тачпад.

Этот негодник был в состоянии воспроизводить видео в течение 8 часов и 48 минут с установленными на 50% яркостью и громкостью — что, конечно, не совсем равно обещанным Acer 10 часов работы, но ушло недалеко. По сравнению с Pixel 2, C910 несколько сдает позиции. Нам удалось промучить его более суровым тестом 8 часов и 22 минут. **LXF**

Влияние CPU

Acer удивила всех, впервые поставив процессоры Broadwell — более быстрые и энергоэффективные, чем процессоры Celeron, обычно встречаемые в хромбуках. Это добавление значительно повысило класс хромбуков: из группы устройств, не предназначенных для серьезных дел, они превратились в устройства, способные выполнять кое-какую работу.

В сравнении с Pixel 2, C910 по производительности почти отстает от более дорогой модели. В сравнительных тестах Octane, измеряющих производительность движка JavaScript, Pixel получил 24,564, а C910 — 25,240. Если кого интересуют сравнительные тесты, он получит 1301,5 в Mozilla Kraken и 192,5 в SunSpider.



» Acer значительно выигрывает от HD-экрана.

LINUX FORMAT Вердикт

Acer Chromebook 15 C910

Разработчик: Acer
Сайт: www.acer.co.uk
Цена: £ 379

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

» Выносливая рабочая лошадка, вознаграждающая скоростью тех, кто не гонится за миниатюрностью.

Рейтинг **9/10**

Ultimaker 2

Аластер Дженнингс исследует на дому тихую 3D-печать с высоким разрешением — и клонировал целую армию себя.

Вкратце

» Эволюция оригинала, предлагающая куда более тихую печать и превосходные результаты. Поставляется в комплекте с прекрасной разработанной и интуитивно понятной программой Cura для загрузки моделей. См. также: RepRap, Lulzbot Mini.

Ultimaker 2 основан на открытом оборудовании и ПО, но это вполне готовый продукт, доступный в собранном виде. У него три модели: Ultimaker 2 Go, Ultimaker 2 и Ultimaker 2 Extended. Канули в прошлое нарезанные лазером деревянные бока и панель управления, придававшие Ultimaker явно самопальный вид; новые материалы, примененные для обшивки, лучше обработаны и увесисты.

Ultimaker 2 — это эволюция исходного дизайна, и хотя он отражает внешний вид своего предшественника, но во всех аспектах стоит выше уровнем по качеству.

Поверхность печати — важный фактор, и Ultimaker 2 предлагает впечатляющие 23×22,5×20,5 см своей нагреваемой печатной площадки. Вверху и спереди принтер открыт, и это означает, что во время печати следует беречь пальцы — при использовании в образовательных учреждениях это надо учитывать. 3D-модели загружаются с поставляемой SD-карты через разъем спереди, а модель, отправляемая на печать, выбирается через LCD-дисплей, тоже спереди. Сами модели, вместе с настройками качества, записываются на карту через программу *Ultimaker Cura*. Сзади расположены держатель подаваемой пластиковой жилы, включатель питания и USB-порт, используемый для обновления прошивки.

Вытащив Ultimaker из коробки, вы обнаружите, что после установки держателя жилы и удаления упаковки принтер будет готов к работе. Держатель просто защелкивается в гнезде, затем с помощью передней панели управления выбирается материал, и жила поступает на подачу

» Полный комплект 3D-печати для дома, работы или учебы.



(сзади) — как только это случится, инструкции на LCD-дисплее скажут вам, что делать, и спустя пару минут пластик загрузится и будет готов к работе. Потом останется подготовить производственную площадку с помощью клеящего карандаша, вставить в разъем SD-карту, нажать Print, выбрать одну из предзагруженных моделей, и процесс печати пойдет.

Нашему образцу не потребовалось никаких подстроек, и печать прошла успешно и на высоком уровне.

Приятный принтер

Когда вы созреете для печати моделей на Ultimaker 2, вначале загрузите свою модель в программу *Cura*. Данное приложение сейчас официально поддерживается в Ultimaker, но используется и другими производителями, например, Lulzbot, и оно на редкость хорошо разработано и интуитивно. Оно позволяет уточнять размер, вращение и масштаб, показанные на виртуальной печатной площадке; кроме того, здесь можно уточнить качество и настройки печати.

Ultimaker 2 удивительно тих, и хотя его отнюдь не назовешь абсолютно беззвучным, его работа не нарушит покой коллег по офису, домашних или соседей. Определяющим фактором шум работы станет только если ваше рабочее место находится в непосредственной близости от принтера, поскольку его шум изрядно отвлекает. И опять же, загрузка файлов с SD-карты означает, что можно с успехом поставить принтер в другой комнате — это большое преимущество перед принтерами, которые надо присоединять к компьютеру, чтобы печатать.

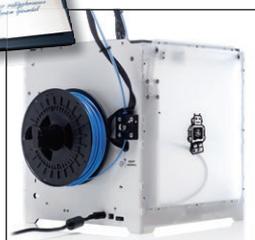
Наша тестовая печать коробки передач длилась от одного часа при настройке Fast Print до 10 часов при настройке Ulti — на фоне других принтеров, таких как Lulzbot Mini, это хорошее время.

На высшем уровне настроек в 20 микрон качество печати превосходное, и хотя слои всё-таки заметны, немного шлифовки быстро преобразит модель. Настройка Normal также дает хорошее качество печати, а вот настройку Low лучше использовать только для очень простых объектов или пробной печати сложных моделей.

Ultimaker 2 возводит 3D-печать на новый уровень. Дизайн, функции и ПО предоставляют вам всю полноту комплектации. Если вы ищете принтер для дома, то маленькое основание и большая печатная площадка очень привлекательны, а при работе и обучении отсутствие шума очень приветствуется. LXF

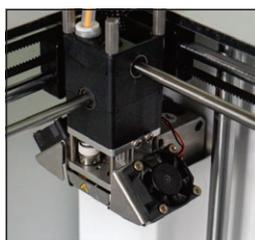


Свойства навскидку



Подача материала

Устройство подачи спрятано позади стандартной бобины для печатного пластика.



Тихая работа

Огромное улучшение в этой модели — практическое отсутствие шума, благодаря ровной работе.

LINUX FORMAT Вердикт

Ultimaker 2

Разработчик: Ultimaker
Сайт: <https://ultimaker.com>
Цена: £ 1749

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

» Легко установить, печать качественная, поддерживаются ABS и PLA, а работа автономная и тихая.

Рейтинг 8/10

Kerbal Space Program v1.0

Мы отважно устремляемся туда, где еще не ступала людская нога, в сопровождении **Фила Сэвиджа** и его команды зеленых человечков.

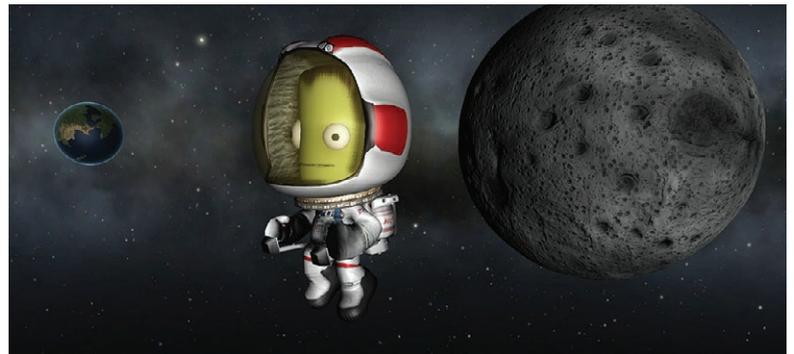
Спецификация

- » ОС Ubuntu14.04 (поддерживается большинство дистрибутивов)
- » CPU Intel Core i5
- » ОЗУ 4 Гб
- » Диск 2 Гб
- » Видеокарта Любая с Shader Model 4, 1 Гб VRAM

Кerbal Space Program (*KSP*) — игра о постройке ракет и полетах в открытом космосе. Вероятно, вы уже слышали о ней, поскольку ее альфа-релиз вышел в 2011 г. Последнее обновление вносит ряд серьезных улучшений, но ни одно из этих отдельно взятых дополнений игру радикально не меняет: основные идеи *KSP* уже сформировались раньше. Обновления были разработаны, чтобы расширить эти идеи, и с течением времени превратили игру в то, что она представляет собой сейчас. Добавлены новые типы миссий, корабельные части, астронавты-кербалы женского пола; переделаны графические эффекты, и появилась новая панель Engineer Report [Отчет инженеров], сообщающая об очевидных ошибках в конструкции корабля.

Режим песочницы — квинтэссенция сути *KSP*. Вам предоставляется неограниченный бюджет и полный доступ к большинству корабельных частей. Экспериментальные игры по определению бесструктурны. В то же время в них есть естественный путь развития, которому вы можете следовать. В игре имеются луна и планеты — каждая из которых достаточно далека, чтобы визит к ней потребовал значительных усилий.

Постоянство природы солнечной системы также играет важную роль для неопытных игроков. С некоторого момента становится невозможно построить достаточно большую и эффективную ракету для посещения соседних планет. Позабывшись о стационарных орбитальных станциях



» Кербалы смягчили то, что иначе было бы сухой и холодной имитацией.

и дозаправщиках, вы сумеете заготовить на своем пути серию заправочных стоянок.

Если вы жаждете большей структурированности, здесь также есть режим карьеры. В начале этого режима вы крайне ограничены в своих постройках. Однако это чуть более легкое и гладкое начало. С меньшим числом доступных корабельных запчастей у вас высвобождается время разобраться в том, что делает каждая из них. Типы контрактов могут предложить сценарии, которые окажутся испытанием даже для знакомых с игрой.

Имитатор с душой

KSP могла бы производить впечатление сухого и холодного имитатора. Однако за то, что это не так, мы должны благодарить презентацию и природу самих кербалов. Это корявые, неуклюжие ремонтники, которые ценятся в игре за отвагу и глупость. Именно кербалы низводят физику до дилетантского уровня — в более серьезной игре ошибки и сбои казались бы трагичными или, еще хуже, теоретическими; и именно кербалы приносят в имитацию столь необходимую душу, а также элемент фарса. Когда один из них гибнет, вы чувствуете себя плохо, но не слишком.

Ну и легко понять, что в *KSP* не последнее место отводится инженерному делу и разработке. Часто это происходит так: игра в доводку центра тяжести корабля, увеличение стабильности или использование орбитальной карты для выполнения управляемого сгорания в отдаленных

пунктах назначения. Игра не особенно привлекательна внешне. Планетарные текстуры примитивны и с низким разрешением. Это выглядит утилитарно, но кажется уместным. Иногда вы будете блуждать в космосе, и солнце будет вставать из-за Кербина, или вы уловите отдаленный отблеск другой планеты. Издали освещение выглядит превосходно. В движении *KSP* полностью способна передать величие и красоту освоения космоса.

Это напоминание: мы сделали это. На фоне науки и неутраченной и почти апокалиптической конкуренции между двумя сверхдержавами мы захпнули людей в гигантские машины и отправили их с нашей планеты. Мы сделали это, и вышло чудесно. Это редкая и прекрасная игра. **LXF**



» Осилит ли мы взлет на 10000 м? Может быть...

LINUX **Вердикт**
FORMAT

Kerbal Space Program

Разработчик: Squad
Сайт: <https://kerbalspaceprogram.com>
Цена: £30

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

» Идеальная смесь науки и фарса, и надежная среда для всяческих экспериментов. Выдающаяся игра.

Рейтинг 9/10

Knights of the Old Republic II — The Sith Lords

Евгений Балдин решил проверить на практике, что хорошо забытое старое вполне себе может быть новым.

В некотором царстве, в некотором государстве спала отравленная Митра Сурик... Нет, что-то не так: в Старой республике случилась большая война, и ветеран этой войны стал пацифистом, обрубив свою связь с силой, но темные силы не дремлют... Хотя разница между сказкой и фантастикой Звездных войн не такая уж и большая.

Миру Звездных войн многие тысячи лет. Все, что есть сейчас (или было в саге Лукаса на экране), когда-то уже существовало, и в гораздо больших масштабах. На каждого современного дарта Вейдера был не менее жуткий дарт Вишэйт. Надо только найти голокорн, обрести суперсилу и сразить дракона, то есть Императора... Поэтому правила классической ролевой игра по типу «Подземелья и Драконы» так замечательно наложились на эту вселенную. Все, что есть в хороших компьютерных ролевых играх, тут имеется. Вполне приличная (хоть и скомканная к концу) история, интересные последователи (правда, некоторые сильно интереснее других, особенно железный здоровяк ржавого цвета), лихорадочный поиск артефактов (напульсник +5 к силе, поясок +4, имплант +2 и колдун



➤ Сначала оглушаем, затем шинкуем. Рутинная. Концовка, к сожалению, довольно скучна.

превращается в светящуюся мясорубку), злобные враги и живой мир с хорошо прописанной историей не дают скучать.

На персональных компьютерах эта игра — настоящая классическая компьютерная ролевая игра во вселенной Звездных войн — появилась благодаря Obsidian Entertainment и Крису Авеллону [Chris Avellone] в частности аж десять лет назад, а пользователям GNU/Linux стала доступна только в этом году, уже благодаря фирме Aspyr. Надо сказать, что по меркам второго десятилетия XXI века картинка выглядит угловато. К сожалению, игра вышла не без багов. В частности, пропадают текстуры (решается переключением по Alt+Tab) и звук в сюжетных вставках (тут помогает выход из игры и загрузка последнего сохранения); но в целом проблем при прохождении не возникает.

Десять лет или около шести поколений GeForce назад на полигонах было принято экономить, но если абстрагироваться от угловатой фигурки главного героя и включить самый мощный ускоритель, а именно свои мозги, то за немного квадратной спиной Митры Сурик (Изгнанник по канону был женщиной) можно разглядеть весьма интересный и реально неоднозначный мир. Почти на каждое положительное действие тут можно совершить злодеяние. Дждедай отнюдь не обязан быть светлым — более того, темным быть гораздо веселее, ибо разить молниями врагов проще, чем оглушать,

а затем шинковать мечами. История же независимо от этого будет вознаграждать погрузившегося в нее, хотя и по-разному.

На беду, изрядная часть игрового содержания была принесена в жертву издателю LucasArts, который заставил разработчиков выпустить игру к Рождеству. Поэтому концовка вышла весьма куцей. К счастью, вот уже десять лет (sic!) группа энтузиастов занимается восстановлением вырезанных сцен. Я не знаю, как эти моды согласуются с последней версией игры, но ничто не мешает поэкспериментировать — ведь компьютер, внутри которого затаилась вселенная Звездных войн, целиком в ваших руках! LXF

➤ Предполагается, что сила джедая в правде, но данный конкретный герой опирается на своих спутников, которых весьма легко заводит, и это неспроста.



Свойства навскидку



Немного багов

В игре частенько пропадают текстуры. К счастью, это лечится.



Как исправить

Достаточно будет переключиться по Alt+Tab из игры и обратно.

LINUX FORMAT Вердикт

Knights of the Old Republic II — The Sith Lords

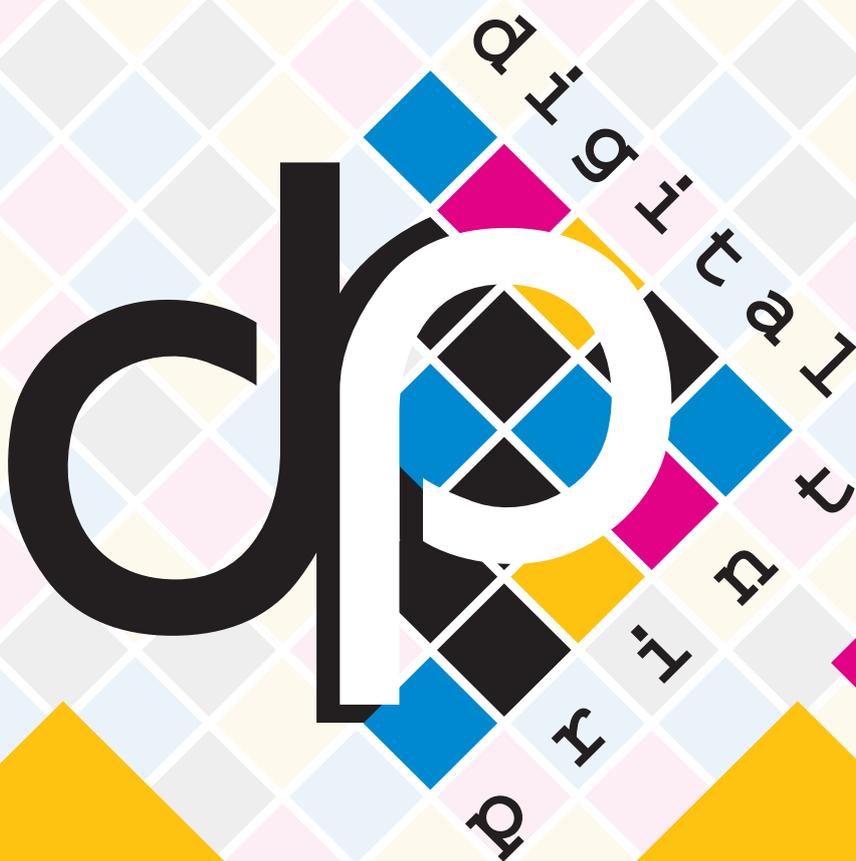
Разработчик: Obsidian Entertainment
Сайт: Нет
Цена: 249 руб. в Steam

Сюжет	8/10
Графика	5/10
Увлекательность	10/10
Оправданность цены	10/10

➤ Эта старая сказка даст фору множеству новых.

Рейтинг 8/10

Производство CD, DVD



CD, DVD, mini-CD, CD-визитки

- ◆ Производство
- ◆ Мультимедиа презентации
- ◆ Электронные каталоги на CD
- ◆ Печать и тиражирование (в том числе с мастер-диска заказчика)

Оформление дисков

- ◆ Дизайн CD/DVD
- ◆ Печать на CD/DVD/
mini-CD/CD-card
- ◆ Выпуск полиграфии
к CD, CD-визиткам

Наши преимущества

- ◆ Собственное производство
- ◆ Высокое качество
- ◆ Доставка
- ◆ Минимальные сроки
и экономия ваших сил
- ◆ РИА и РА – скидки на всё!

Комплексное решение

по оперативному
выпуску ваших CD/DVD,
электронных CD-визиток,
к выставкам, семинарам,
конференциям, с вашей
рекламой, каталогом
продукции

Digital Print

(812) 591-79-80

(812) 591-79-81

diprint.spb.ru

Санкт-Петербург,
ул. Смольячкова, 19,
офис 405А
(м. «Выборгская»)



ВСТРОЕННОЕ ПО

Плюсы и минусы

Передний край мобильных платформ буквально «отправился в путь».

Э то лучше, чем бортовая информационно-развлекательная система (БИРС) стандартного автопроизводителя, однако ПО ограничено, а возможности обмена мгновенными сообщениями посредственны: Google предстоит большая работа по совершенствованию Android Auto. К такому выводу пришел Рон Амадео [Ron Amadeo] (издание Ars Technica), проведя тест-драйв автомобиля Hyundai Sonata 2015 модельного года, БИРС которого работает с Android Auto.

Первое, что удивило тестировщика: реализация Android Auto для Sonata выполнена на версии Android 2.3 Gingerbread (выпущена Google в декабре 2010 г.) — довольно ограниченной по сравнению с новейшими редакциями. Так, поддержка разрешения всего лишь 800×480 напоминает «дрянные планшеты 2011 года». Функционирование голосового управления отстает от своего аналога на телефонах. Ограничен интерфейс.

Не всякое приложение может работать с автомобильной системой: в настоящее время только 17 приложений из Play Store совместимы с Android Auto, и все они связаны с обменом сообщениями и воспроизведением медиа-контента. Впрочем, с точки зрения безопасности интерактивные эффекты на приборной панели в процессе движения автомобиля совершенно излишни; более целесообразны здесь такие приложения, как Pandora, Spotify, TuneIn, Google Play Music и т. п. Наличие отвлекающих внимание водителя от дороги приложений для обмена сообщениями — Skype, WhatsApp, Google Hangouts — также достаточно спорно. Тексты входящих сообщений зачитываются Android Auto вслух, но без привязки к какому-либо контексту или истории сообщений. Если входящее сообщение содержит ссылку — система зачитает ее посимвольно.

К положительным сторонам тестировщик отнес ключевую функциональность: добавление вида



» Android Auto: уже лучше стандартных автомобильных информационно-развлекательных систем, но впереди у Google еще много работы.

со спутника во время движения, работа с картографическим сервисом Maps.

Наконец, не стоит забывать: телефон, подключаемый к бортовой системе Android Auto, должен быть оснащен операционной системой Android 5.0 или выше.

ВИРУСЫ

Легкая добыча

Большинство телефонов на Android можно взломать простым MMS-сообщением.

Э кспloit, которому надо знать лишь телефонный номер жертвы, разработал Джошуа Дрейк [Joshua Drake], вице-президент компании Zimperium. Дрейк выявил массу уязвимостей в Stagefright — компоненте Android для работы с мультимедиа-файлами. Некоторые из них допускают удаленное выполнение кода и могут инициироваться при получении MMS-сообщения, загрузке через браузер специально обработанного видеофайла или открытии web-страницы со встроенным мультимедиа-контентом. Уязвимы устройства с Android 2.2 и выше.

Направлений для атак много: помимо воспроизведения медиа, Stagefright применяется для автоматической генерации миниатюр и извлечения метаданных и подобной информации. Для активации уязвимости пользователю не обязательно запускать вредоносные файлы — достаточно их скопировать. Выгоды жуликов по эксплуатации уязвимости зависят от конкретного устройства:

полномочия у вредоносного кода те же, какими Stagefright располагает в системе. В основном это доступ к микрофону, камере и внешнему хранилищу, и нельзя устанавливать приложения или получать доступ к внутренним данным. Но на около 50% затронутых устройств у Stagefright системные привилегии, что упрощает получение доступа root и полного контроля над устройством.

Дрейк отослал необходимые патчи Google, где их в течение 48 часов применили к внутренней кодовой базе Android. Этот код заранее выдан производителям — участникам партнерской программы Android — до его публикации в AOSP. Но из-за медленного в целом темпа обновлений более чем 95% устройств на базе Android все еще уязвимы. Даже среди устройств Google Nexus до сих пор отчасти исправлен один Nexus 6. Патчи, не будучи в AOSP, недоступны производителям устройств, не являющимся партнерами Google. То есть стопроцентное встроенное микроПО, в т. ч. CyanogenMod,



» Выявленные в библиотеке Stagefright уязвимости несут потенциальную угрозу для 95% Android-смартфонов.

все еще уязвимо. Но в частном порядке Дрейк выдал патчи некоторым заинтересованным сторонам, включая Silent Circle и Mozilla. Silent Circle исправила свои прошивки Privatos 1.1.7 для Blackphone на базе Android, а Mozilla устранила проблему в выпущенном в мае Firefox 38.

КОНКУРЕНЦИЯ

Третий — не лишний!

Суаноген намерен разрушить дуополию Android и iOS.

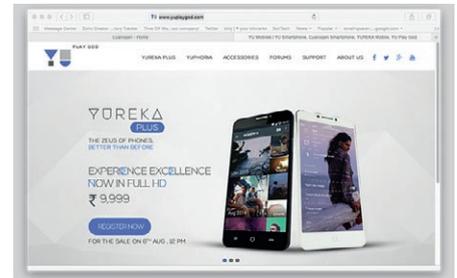
В последние годы на мировой арене появились несколько версий ОС на базе Android для смартфонов. Одной из них — Суаноген OS — удалось привлечь к себе общественное внимание и именитых инвесторов (Twitter, Qualcomm, Foxconn); в эту версию Android делают взносы свыше 9 тыс. разработчиков. В июльском отчете аналитик компании IDC Уильям Стофегга [William Stofega] утверждает, что потенциал Суаноген позволяет ему разрушить текущую дуополию Android и iOS, предложив конечным пользователям более насыщенный опыт. Однако сделать это совсем не просто, учитывая, что сегодня Apple и различные производители телефонов на Android контролируют более 95% рынка смартфонов.

Сейчас в мире циркулирует около 2,5 млрд смартфонов. Согласно данным исследовательских компаний (Statista и ряд других), в 2014 г. телефонами на базе Android владели 1,6 млрд пользователей, смартфонами на iOS — чуть менее 400 млн. Начиная с 2009 г., с момента зарождения Суаноген, на свои смартфоны ее скачали 50 млн пользователей. Теперь же прогноз аналитиков — резкий рост

популярности Суаноген: к 2020 г. число смартфонов в мире возрастёт до 6 млрд, и Суаноген отводится немалая часть.

Благоприятные перспективы Суаноген на ближайшие годы вполне очевидны: его покровители хотят привлечь разработчиков приложений, которые чувствуют себя слишком контролируемыми, работая на Apple или Google. В конце концов будет запущен и свой магазин приложений. Стофегга уверен, что Суаноген несет серьезную угрозу Google, если производители телефонов начнут отказываться от совместимости с магазином и приложениями последней. По его мнению, эта ОС позволяет целиком использовать ядра процессора, обеспечивая лучшую защиту ряда приложений, возможность модификации интерфейса и работу с камерой. Наконец, согласно тестам, ее быстродействие выше, чем у стандартного Android.

В текущем году уже вышло несколько смартфонов с Суаноген OS. Версия 12 этой ОС установлена на смартфон Yureka Plus, продажи которого 6 августа начало YU Televentures — совместное предприятие, которое создали в Индии в декабре



➤ 6 августа начались продажи работающего под управлением Суаноген OS 12 смартфона Yureka Plus (\$156, дисплей 5,5" full HD).

2014 г. Суаноген Inc. и Micromax Informatics. Цена устройства — 9999 рупий (приблизительно \$156), оно обладает 5,5-дюймовым дисплеем высокой четкости и камерой на 13 мегапикселей. Суаноген также предустановлен на Alcatel OneTouch Hero2+, OnePlus One, Oppo N1, а позже в текущем году появится на устройстве от Blu, компании из Флориды, продающей телефоны в Латинскую Америку, а также через Walmart и Best Buy.

ЭКОСИСТЕМА

Тысячи их...

Samsung доминирует, фрагментация растёт.

В мире ныне существуют по крайней мере 24 тыс. разнообразных моделей устройств на базе Android, утверждает ежегодный отчет компании OpenSignal, в котором собраны аппаратные и программные данные, рисующие полную картину современной экосистемы Android. Данные получены от последних 682 тыс. загрузок мобильного приложения OpenSignal, алгоритмы интеллектуального анализа данных которого позволяют определить установленную версию Android и базовые аппаратные спецификации устройства.

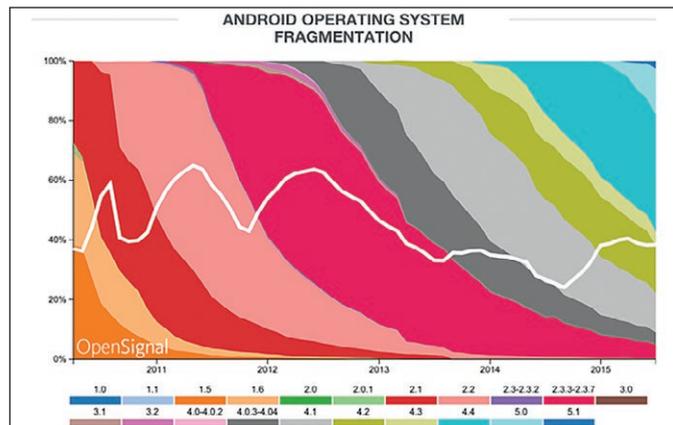
OpenSignal придерживается величины 682 тыс. для прямого сравнения одного года с другим. Так, отчет 2014 г. указывал на наличие 18 796 различных моделей под управлением Android, а в этом году их стало 24 093. Компания Samsung явно лидирует: ее доля рынка составляет почти 38%.

Отмечены и другие тенденции для устройств Android. Размеры экранов (что не удивительно) все увеличиваются; растут объемы ОЗУ и количество ядер процессора, расширяется внедрение технологии NFC. А поскольку мобильные устройства становятся основным компьютером для всё большего

числа людей, следует ожидать дальнейшего прогресса их технических характеристик.

Основной проблемой мобильной ОС от Google остается фрагментация. Отчет показывает, что наиболее популярные версии Android, как правило, занимают долю рынка, не превышающую 40%. Спустя 9 месяцев после выхода Lollipop эта версия

Android установлена всего лишь на 18% устройств. Фрагментация не выгодна ни пользователям, которые хотели бы, чтобы их устройства были обеспечены новейшей функциональностью, ни разработчикам, которым приходится обеспечивать совместимость своего ПО с множественными версиями Android. LXF



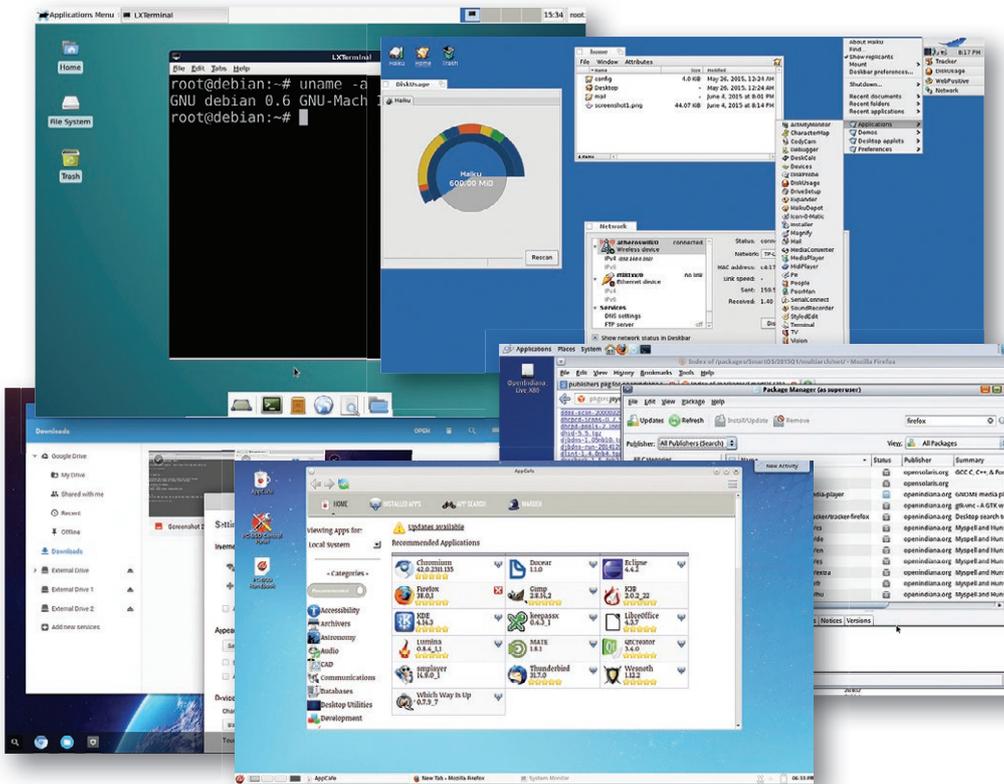
➤ По данным OpenSignal, самая популярная версия Android занимает не более 40% рынка.

Сравнение

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

Альтернативные ОС

Мы любим Linux во всех его видах, но это не единственное развлечение в городе открытого кода; и в нашем 200-м выпуске Александр Толстой рассматривает альтернативы.



Про наш тест...

Не секрет, что при оценке операционной системы используемые методы тестирования и сравнения могут довольно значительно повлиять на результат. Для данного Сравнения мы постарались свести этот риск к нулю, проводя тесты на виртуальной машине (в *VirtualBox*) и на реальном оборудовании (ноутбук HP с двоядерным AMD CPU и видеокартой Radeon).

Тестирование нишевой ОС на реальном компьютере может выдать противоречивые результаты, потому что опыт пользователя зависит от настоящих драйверов; но нам кажется, что если одни будут играть с ОС в безопасной виртуальной среде, то другим будет любопытно запустить их на свободном разделе или отдельном жестком диске. Мы сравним эти ОС по производительности, удобству в использовании, числу имеющихся функций и приложений, онлайн-поддержке и состоянию разработки.

Есть множество ОС с открытым кодом, которые, однако, не используют ядро Linux или, по крайней мере, имеют свои пакеты пользовательских программ. Однако с чего бы вам с ними знакомиться? Ну, изучить структуру другой ОС, системные инструменты и работу в целом бывает полезно; и надо признать, что некоторые альтернативные ОС весьма сильны для определенных задач. Например, OpenIndiana предлагает функции хранения уровня предприятия (благодаря Sun Microsystems былых времен); PC-BSD обладает всеми преимуществами FreeBSD и очень хороша для web-серверов

«Выделим ОС с наилучшим практическим применением для среднего пользователя Linux.»

(и не только); Haiku — уникальный проект и вообще не имеет отношения к системам на UNIX, но она очень быстрая; а Chromium OS является открытой версией Google Chrome OS, на которой работают быстрые и экономно расходующие заряд батареи хромбуки, чья популярность растет.

Так может ли основанная на облаке ОС от Google конкурировать с классическим

подходом других? Давайте посмотрим; и мы также постараемся найти самый чистый проект открытого кода GNU/Hurd и поставить его в ряд с другими участниками. Наши интересы сосредоточены в основном на настольных ПК, и нашей целью будет выделить ОС, которые являются наилучшими для практического применения среднестатистическим пользователем Linux.

Наша подборка

- » Chromium OS
- » Debian
- » GNU/Hurd
- » Haiku
- » OpenIndiana
- » PC-BSD

По оборудованию

Будут ли они по умолчанию работать на вашем ПК?

OpenIndiana, вероятно, отлично загрузится live с USB-устройства или DVD, и большинство системных компонентов у вас заработают. Здесь имеется поддерживаемый сообществом Hardware Compatibility List (<http://wiki.openindiana.org/oi/Components>), который показывает наличие даже проприетарного драйвера Nvidia для определенных чипов в OpenIndiana. Чипсеты Radeon поддерживаются базовым драйвером VGA, и есть отчеты о работе большинства чипов Wi-Fi.

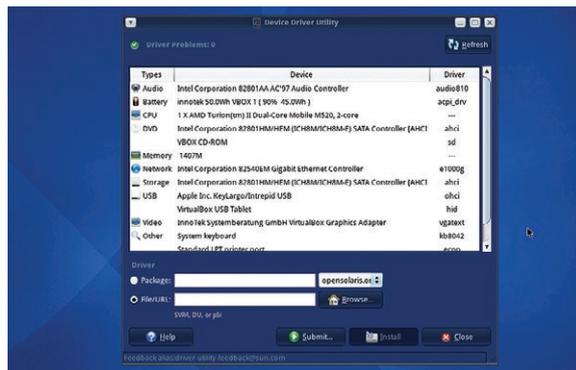
PC-BSD предлагает наилучшую поддержку оборудования, и очень близка к тому, что имеется в Linux. Система предлагает официальные бинарники Nvidia и драйверы Intel для ускорения работы оборудования, а также поддержку Gallium3D для большинства чипов Radeon. Однако наилучшая производительность OpenGL достигнута в Kwin, в среде рабочего стола Plasma.

В других аспектах PC-BSD соответствует таблицам совместимости оборудования FreeBSD, поскольку использует то же самое ядро. Это означает, что вы можете установить PC-BSD на настоящий компьютер — и убедиться, что большинство

его компонентов работают по умолчанию, включая беспроводную сеть, печать и т.д. Конечно, не все доделано: например, Nvidia Optimus будет работать, но без удобных опций переключения между чипами, и некоторые периферийные устройства со специальными объектами Linux могут остаться без поддержки.

У Chromium OS ядро Linux, и она должна приемлемо работать с большинством устройств. Однако из-за вынужденных ограничений в основанной на облаке ОС в ней в некоторых моделях не хватает некоторых жизненно важных функций — например, поддержки тачпада, что весьма странно, поскольку эта ОС предназначена для ноутбуков. В остальных аспектах Chromium OS продемонстрировала хорошее качество с, вероятно, лучшей поддержкой внешних периферийных устройств (опять же благодаря ядру Linux).

Haiku — совсем другая история. Над кодом Haiku на постоянной основе трудятся всего лишь два разработчика, поэтому мы не можем реально требовать достойной поддержки оборудования от этой крошечной ОС. Тем не менее, Haiku отлично



» OpenIndiana предлагает утилиту *Device Driver Utility*, чтобы показать вам, какие драйвера на данный момент используются.

справилась с разными Wi-Fi адаптерами, которые мы смогли отыскать, и удивила нас мгновенным доступом к защищенным WPA2 сетям.

Худшим случаем по сравнению с остальными в плане поддержки оборудования оказался Debian на GNU/Hurd. Нет никакой поддержки AGP GART внутри Mach, поэтому почти любой видеочип будет использоваться с драйвером VESA; задействуется максимум 1,7 ГБ ОЗУ (остальное тихо игнорируется); поддержки звука нет вообще; и нет поддержки USB (хотя некоторые клавиатуры и мыши будут работать, благодаря унаследованному интерфейсам, эмулирующим BIOS). Найти правильную конфигурацию ПК, на которой будет работать GNU/Hurd, крайне непросто.

Вердикт

Chromium OS

★★★★★

PC-BSD

★★★★★

OpenIndiana

★★★★★

Haiku

★★★★★

Debian GNU/Hurd

★★★★★

» Можете попробовать загрузить все системы, но Hurd избегайте.

Простота установки

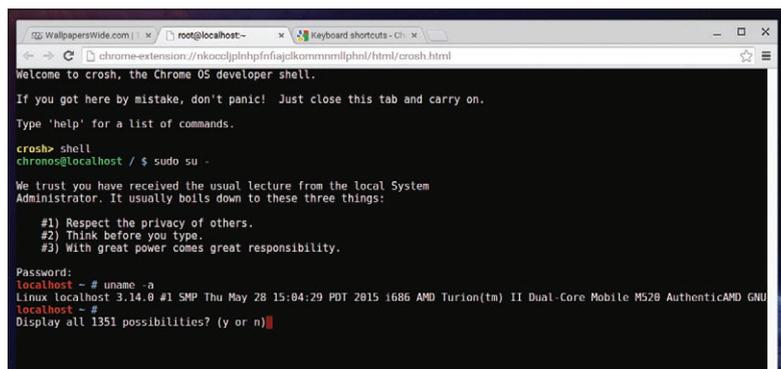
Что нужно, чтобы они заработали?

Все участники нашего Сравнения было легко установить в виртуальной среде. Выбор ISO в качестве первичного средства загрузки в *VirtualBox* позволило нам запустить их все — или установив, или в режиме live. Мы также хотели рассмотреть каждую ОС на реальном

оборудовании, записав ISO на физический носитель, например, на USB-накопитель.

Сайт OpenIndiana предлагает образ USB, который, как оказалось, довольно просто на USB записать. Проще использовать обычный ISO, если вы знаете, откуда его скачать. (Образов целая коллекция на FTP:

<http://bit.ly/1fPr1m>). Что способно воспрепятствовать первому знакомству с этой ОС. PC-BSD записывается на USB легко и предлагает понятную и очень симпатичную программу установки на Qt. На вход во вновь установленную систему BSD, уходит всего несколько минут безо всяких проблем. Chromium OS в этой области нас разочаровала, ибо работает только в одном режиме. Когда образ ОС записан на USB-накопитель, он автоматически становится загружаемым устройством с готовой к работе системой. В руководстве разработчика Chromium OS есть опция установки на жесткий диск, но это не настоящая программа установки, а просто скрипты, которые перепрограммируют драйвер из рабочей среды Chromium OS. Haiku проще всех в работе и установке. Она предлагает режим live и очень хороший (и быстрый) установщик. У Debian GNU/Hurd несколько режимов установки, как и у Debian Linux, но нет режима live, и на установку уходит добрый час.



» Chromium OS очень легка в использовании, но столь же тяжела в настройке, если вы не разработчик.

Вердикт

Haiku

★★★★★

PC-BSD

★★★★★

OpenIndiana

★★★★★

Chromium OS

★★★★★

Debian GNU/Hurd

★★★★★

» Ничья между беспрепятственными установками PC-BSD и Haiku.

Производительность

Насколько они шустрые?

Привыкнув к молниеносной, за десятки доли секунды, загрузке Linux, вы, естественно, начинаете ожидать подобного поведения и от других ОС.

Производительность сильно влияет на впечатление от ОС, скрадывая даже недостаток функциональности: какой бы замечательной ни была

некая авангардная функция, проку от нее мало, если отклика приходится дожидаться годами.

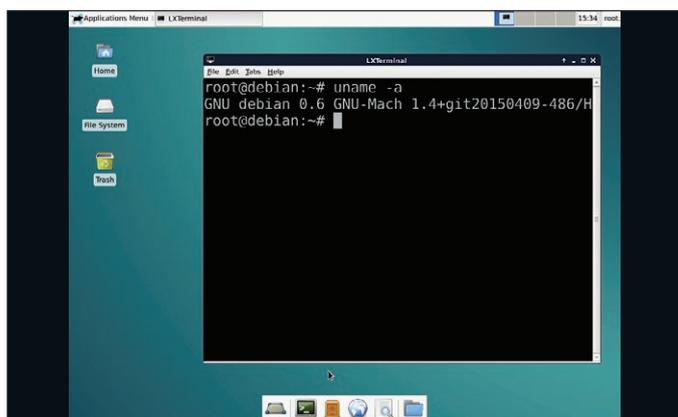
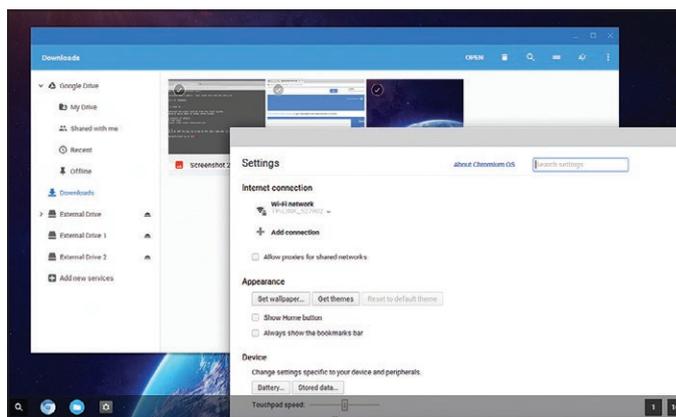
Любому пользователю, понятное дело, нужны быстрая установка, быстрая загрузка и небольшой период ожидания между нажатием мышью на значок приложения и его запуском. Если наблюдается отклонение, мы хотим знать, чем оно вызывается

и мешает ли оно нормальному взаимодействию с пользователем. Разница между производительностью операционных систем была очевидна в виртуальной среде, но уже не столь очевидна на голом металле, хотя ваш опыт будет другим, поскольку он сильно зависит от состава и спецификаций вашего оборудования.

Chromium OS ★★★★★

Производительность Chromium OS вряд ли может вызывать нарекания — или очень незначительные нарекания: она использует ядро Linux, основана на Gentoo и использует самые современные версии всех системных компонентов, убирая почти все локальные приложения. После запуска сервера X.org остальное делает браузер, включая логин пользователя и сессию, управление окнами и т.д. Chromium OS старается использовать чистые версии исходных пакетов Gentoo, хотя от Google и от сообщества идет немало патчей для оптимизации Chrome/Chromium OS на определенном целевом оборудовании, таком, как хромбуки.

Поэтому производительность ОС очень хороша, и если на настольном Chromium что-то тормозит, это вина кода браузера, а не основной системы. Помня об этом, вы можете поиграть с этой ОС на оборудовании с 1 ГБ ОЗУ и не слышном производительным CPU.



Debian GNU/Hurd ★★★★★

В Debian GNU/Hurd наконец-то добавили поддержку X.org в Hurd по умолчанию. Однако с точки зрения пользователя рабочего стола система невероятно нестабильна и содержит огромное количество ошибок. Для начала, она не выводит графический рабочий стол для обычного пользователя — нам удалось пробиться на рабочий стол LXDE, введя \$ startx от имени root.

Система, работающая на Hurd, подтормаживает и выдает очень медленный отклик, и крайне сложно понять, происходит это из-за неспешной производительности 2D-драйвера VESA или из-за проблем с вводом/выводом микроядра. Мы попытались запустить тесты Phoronix для Debian/GNU Hurd, но оказалось, что работают только несколько тестов, таких, как кодирование LAME MP3, C-Ray, сжатие 7-Zip и т.д. — и все они показали небольшое отставание 4–6% от обычного дистрибутива Debian GNU/Linux; однако это так и не проливает свет на причину, почему же Hurd настолько тормозит.

Уровень и качество поддержки

Сможете ли вы получить достойную помощь в Сети?

Выход за пределы мира Linux означает, что рано или поздно вы столкнетесь с некоторыми проблемами, поэтому важным вопросом будет: где можно найти ответы?

OpenIndiana имеет солидный сайт (wiki.openindiana.org) с подробными разделами по компиляции установке и использованию системы, с одной лишь оговоркой: информация в основном предназначена для разработчиков и сисадминов, а среднестатистическому пользователю больше заглянуть некуда, разве что погуглить.

PC-BSD в плане поддержки намного лучше, поскольку у нее отличная страница поддержки сообщества (www.pcbsd.org/en/community) с большим количеством ссылок на разные форумы, списки рассылки, IRC, блоги и т.д. Есть также много неофициальных ресурсов PC-BSD и сайтов Free-BSD, применимых к обоим.

Chromium OS обладает целым рядом руководств на www.chromium.org/chromium-os, включая инструкции по быстрому старту. Однако создается впечатление, что с широкой общественностью делятся

лишь небольшим количеством информации, а основные события происходят внутри Google.

Проект Haiku имеет официальное руководство пользователя и другие материалы на www.haiku-os.org/guides, и вся приводимая там информация весьма точная.

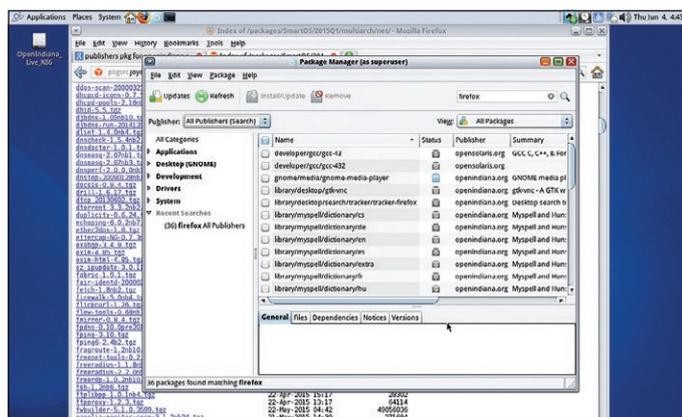
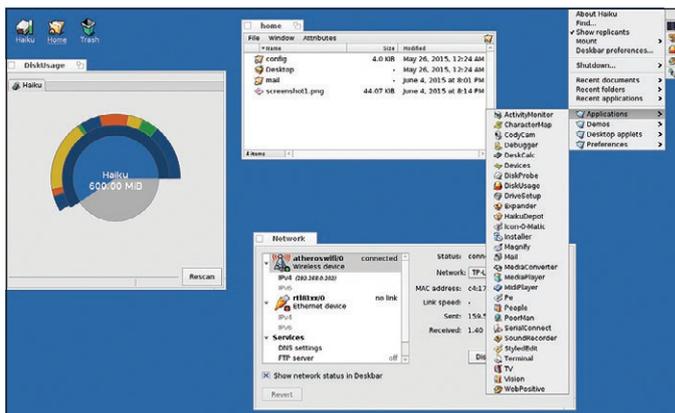
И, наконец, документация на www.gnu.org, которая исправлена на страницах www.debian.org/ports/hurd — это, вероятно, лучший элемент во всем проекте GNU/Hurd. Удивляться тут нечему, ведь он существует с середины 1990-х.

Вердикт

- PC-BSD ★★★★★
 - Debian GNU/Hurd ★★★★★
 - Haiku ★★★★★
 - Chromium OS ★★★★★
 - OpenIndiana ★★★★★
- » Системы BSD — сильные соперники Linux по части поддержки.

Haiku ★★★★★

Мы нахваливали Chromium OS за то, что она очень быстрая и подвижная, поэтому вы, возможно, решите, что она окажется самой быстрой ОС в этом Сравнении. Тогда почему только четыре звезды? Что ж, Haiku работает быстрее, чем Chromium OS; быстрее, чем любая разновидность Linux; и намного быстрее, чем любая другая система в наших тестах. Haiku нас удивила, показывая наилучшие цифры по всем задачам, которые перед ней ставили. У нее уходило 10 секунд на загрузку и 1–2 секунды на открытие любого приложения. Учитывая, что Haiku — только 32-битная, без всяких опций, скомпилированная в основном на древнем компиляторе GCC2 без графического ускорения, эти результаты просто поразительны. Haiku стала чистым победителем, с идеально оптимизированным графическим пакетом и с крошечным отпечатком во всех аспектах. Haiku не будет блистать в требовательных к CPU операциях и сжатии, но при обычных действиях на рабочем столе невероятно быстра.



OpenIndiana ★★★★★

Производительность OpenIndiana падает, когда настройки системы используют базовый видеодрайвер VESA, который снижает реакцию рабочего стола. Независимо от видеодрайвера, ОС показывает значительную задержку при запуске и работе с разными приложениями. ОС использует на своем разделе root файловую систему ZFS, и это добавляет некоторые чудесные функции (например, моментальные снимки), но одновременно добавляет и издержки производительности.

OpenIndiana также использует смешанный режим 32/64-бит. Ядро UNIX может работать в полном 64-битном режиме, а большинство компонентов системы 32-битные — вот почему они работают медленнее. PC-BSD тоже использует ZFS, и хотя они отличаются в большинстве других аспектов, в плане производительности рабочего стола видимой разницы нет: операции с файлами и 2D-графика медленнее, чем в Linux, но отставание не чрезмерно.

PC-BSD ★★★★★

После установки PC-BSD загружает экран приглашения с черепашей скоростью: на это уходит около минуты. Однако общая производительность рабочего стола в KDE4 довольно неплоха — и в режиме по умолчанию, и в виртуализации (плюс PC-BSD автоматически включает Guest Additions). Она менее быстра, чем средний дистрибутив Linux, из-за более медленной производительности рабочего стола ZFS по сравнению с ext4 и более примитивной поддержки Gallium3D, но все равно очень удобна в использовании. Такие приложения, как Firefox или LibreOffice, загружаются за несколько секунд, но у AppCafe (менеджер программ PC-BSD) уходит несколько минут на инициализацию, поиск обновлений и их установку — очень неприятно. Во многих других аспектах эта операционная система работала хорошо, она автоматически включила VirtualBox Guest Additions и обеспечила ускоренную графику для нашего чипсета Radeon.



Состояние разработки

Развивается ли команда, работающая над вашей альтернативной ОС?

Активная поддержка крайне важна для выживания любой операционной системы, и у каждой из наших ОС разное число разработчиков, вкладывающих на них, потому-то так сильно различается время между релизами.

Может, OpenIndiana и сильна благодаря солидному наследию сообщества OpenSolaris, однако текущая скорость разработки — это скорость улитки. Последний релиз — 151a8, и он вывалился из раковины улитки в августе 2013 г., спустя год после предыдущего. Ветвь разработки oi_151a9,

похоже, вполне себе жива, однако мы совершенно не уверены в ее будущем.

PC-BSD куда более оживленная, новый релиз появляется раз в 3–5 месяцев, а Chromium OS может похвастаться сотнями разработчиков, и ее версия синхронизируется с релизами браузера Chromium. Однако для Google OS нет официальных ISO-образов, только набор случайных сборок от разных энтузиастов, что напоминает полу-возобновляемую модель релиза.

Разработка Haiku OS идет на редкость медленно: последний ее «официальный»

релиз (Alpha 4) появился в 2012 г. Однако развитие Haiku куда более многообещающее, с постоянными событиями и участием в Лете Кода Google. Компиляции Haiku от Nightly демонстрируют постоянство разработки, хотя ее официальные релизы далеко не столь частые.

У Debian GNU/Hurd вышел новый релиз в 2015 г., основанный на кодовой базе Debian 8 (Jessie), и она дает человечеству основания надеяться, что мечта Ричарда Столлмена [Richard Stallman] о полностью свободной ОС не умрет.

Вердикт

- PC-BSD ★★★★★
 - Chromium OS ★★★★★
 - Debian GNU/Hurd ★★★★★
 - Haiku ★★★★★
 - OpenIndiana ★★★★★
- » Нам немного тревожит будущее OpenIndiana.

Функции и приложения

Много ли удобных приложений рабочего стола они предлагают?

OpenIndiana предлагает базовый набор приложений рабочего стола в своей свежей установке (ISO весит менее 900 МБ) и немного больше в двух репозиториях: основном и унаследованном зеркале старой Opensolaris.org). Имеются дополнительные репозитории на <http://sfe.opensw.org> и <http://smartos.pkg.ec>, но там приложений рабочего стола очень мало.

PC-BSD предлагает *AppCafe*, ворота в собственный репозиторий системы, и поддержку портов классической FreeBSD (только из командной строки). А еще не будем забывать о чудесной технологии Kldload, которая позволяет системе BSD запускать бинарники Linux, включая *Skype*, *Adobe Flash* и некоторые другие компоненты, которые доступны для Linux, но не BSD (по крайней мере, официально).

Chromium OS имеет единственное место, откуда можно установить дополнительные приложения, и это — Chrome Web Store, открытая торговая площадка для web-приложений для *Google Chrome* и Chromium OS. Однако печальный факт в том, что эти web-приложения не могут конкурировать с классическими локальными приложениями. Они не то чтобы совершенно ужасны, но не соответствуют качеству приложений рабочего стола, и было бы вовсе неуместно сравнивать таких тяжеловесов рабочего стола, как *GIMP*, с расширениями браузера в магазине.

В Haiku весьма скромный набор приложений, в основном доступных из *Haiku Depot*, это менеджер пакетов системы. Устаревшие сайты, типа BeBits и Haikuware, больше не поддерживаются, хотя можно найти случайные приложения для Haiku — например, *Scribus* и несколько основанных на Qt4 приложений. Это немного, но, по крайней мере, хоть что-то.

В противоположность им, команда GNU/Hurd, Debian умудрилась портировать около 78% пакетов Debian для работы на ядре GNU/Mach; но там по-прежнему нет никак сред рабочего стола, кроме *Xfce* и *LXDE*.

```
File Edit View Bookmarks Settings Help
[ato1stoy@pcbsd-5823] ~% su -
Password:
[root@pcbsd-5823] ~# kldstat
Id Refs Address Size Name
1 124 0xffffffff80200000 14eb000 kernel
2 2 0xffffffff81996000 24e76 drm.ko
3 1 0xffffffff819bb000 409cc vboxguest.ko
4 1 0xffffffff819fc000 105a vboxvideo.ko
5 3 0xffffffff819fe000 337a9 crypto.ko
6 1 0xffffffff81a32000 4ef9 aesni.ko
7 1 0xffffffff81a37000 1f848 geom_eli.ko
8 1 0xffffffff81a57000 24dd67 zfs.ko
9 2 0xffffffff81ca5000 5a75 opensolaris.ko
10 1 0xffffffff81cab000 10725 tmpfs.ko
11 3 0xffffffff81cbc000 a84e8 linux.ko
12 1 0xffffffff81d65000 22949 geom_journal.ko
13 1 0xffffffff81d88000 22bc2 geom_mirror.ko
```

➤ Команда *kldstat* показывает, какие модули Linux в данный момент используются BSD.

Вердикт

- PC-BSD ★★★★★
- Haiku ★★★★★
- OpenIndiana ★★★★★
- Chromium OS ★★★★★
- Debian GNU/Hurd ★★★★★

» Chromium OS не хватает приложений рабочего стола, имеющих у других.

Первые шаги

Легко ли к ним привыкнуть?

OpenIndiana располагает режимом live и встретит вас усовершенствованным рабочим столом Gnome 2.30 и *Firefox 10*; однако в ее репозиториях нет *LibreOffice*, хотя *OpenOffice* имеется. Если вас не смущают староватые версии пакетов OpenIndiana, вы будете чувствовать себя комфортно. Единственным подлинным препятствием может оказаться сетевая карта — если соединение не обнаружится автоматически, вам придется столкнуться с ручной настройкой (подсказку ищите на <http://bit.ly/1SYUk2>).

У PC-BSD нет режима live, и единственный сложный этап ее установки — это программа разбиения диска на разделы. Системы BSD имеют другой принцип именования разделов диска (например, */dev/sda1* будет */dev/ada0s1a*), но в остальном PC-BSD может быть благополучно установлена вместе с дистрибутивом Linux на разных разделах одного диска. Программа установки KDE использует по умолчанию рабочий стол KDE4,

однако в *AppCafe* предусмотрены и другие варианты.

С Chromium OS сложно начать, но в работе он становится проще. Если вы не разработчик Chromium, то можете отыскать прекомпилированный образ на <http://bit.ly/ArnoldtheBat> и записать его на USB-накопитель с помощью *dd*. После загрузки надо перейти в командную строку с помощью *Ctrl+Alt+t*, чтобы вывести оболочку, и войти в оболочку для доступа к классической CLI. После этого вы вводите *\$ sudo /usr/sbin/chromeos-install* и затем пароль root (для образов по вышеприведенной ссылке это *password*) и выбрать целевое устройство. (Учтите: вся информация с этого устройства будет стерта.)

Если Haiku определит вашу сетевую карту, вы будете лазить по Интернету через браузер *WebPositive* через считанные секунды. Haiku очень проста в использовании и настройке, и логика ее интерфейса очень похожа на OS X, во многом благодаря



➤ Браузер Haiku по умолчанию, *WebPositive*, это умное web-приложение на основе WebKit.

Жану-Луи Гассе [Jean-Louis Gasse], бывшему руководителю Apple.

Debian GNU/Hurd требуют дополнительных шагов после установки, например, *\$ dpkgreconfigure x11-common*, чтобы пользователи смогли запустить сессию X, и некоторых других хитростей (см. <http://bit.ly/HurdConfig>). Но в общем и целом работа с системой ужасна, и в ней нет режима live.

Вердикт

- Haiku ★★★★★
- OpenIndiana ★★★★★
- PC-BSD ★★★★★
- Chromium OS ★★★★★
- Debian GNU/Hurd ★★★★★

» Haiku перешибет любой легковесный дистрибутив Linux, и выглядит тоже круто!

Альтернативные ОС

Вердикт

Огромное число ОС с открытым кодом подтверждает, что сообщество разработчиков с широкими взглядами могут делать великие вещи, с которыми, как минимум, стоит познакомиться на вашем домашнем ПК. Мы не настаиваем, чтобы вы перешли с Linux на другую ОС, поскольку любим Linux, однако почти все они более-менее пригодны для настольного использования.

PC-BSD вышла в победители, продемонстрировав хорошую производительность почти во всех тестах, которым мы ее подвергли. ОС быстрая, надежная, и умеет распознавать почти все элементы оборудования и периферии. Возможно, будь у нее режим live, это привлекло бы к ней больше энтузиастов открытого кода, но качество рабочего стола здесь практически то же, что у достойного дистрибутива Linux.

Haiku — умная ОС, не похожая на остальные. Ее сборки древним компилятором GCC 2 по-прежнему способны работать

с приложениями исходной BeOS и относительно современными приложениями Qt4. Разработка Haiku не отличается быстротой, но небольшие изменения накапливаются в функции, достойные похвалы, такие, как рабочие WPA2-защищенные соединения Wi-Fi, лучшая поддержка USB и многое еще.

OpenIndiana стареет, в этом нет сомнений. Ранее большие надежды возлагались на OpenSolaris, а позднее — на проект OpenIndiana/Illumos; думали, что он вдохнет новую жизнь в «настоящую» UNIX System V и приведет ее на рабочий стол. Однако сейчас ведется очень слабая работа, и за последние годы свежих релизов не было. Но с OpenIndiana все равно стоит познакомиться: она вполне ладит с современным оборудованием.

Chromium OS является попыткой заменить классическую настольную работу т.н. «облаком». Если работа вашего ПК впишется в браузер Chromium,



возможно, это ОС для вас: быстрая, элегантная, и намеренно прячет свои настройки. Нам по душе этот подход, но правда в том, что облачным технологиям, на тонких клиентах, не сравниться с классическими локальными приложениями по функциям и гибкости.

Пятое место — у GNU/Hurd, бесславной попытки создать самую чистую основу ОС GNU. В Debian провели огромную работу, создавая рабочий дистрибутив с микроядром GNU/Mach, но пока ему далеко до стабильности и удобства в использовании.

» Вероятно, вы и не заметите, что вы не в Linux, пока не войдете в терминал.

«PC-BSD быстрая, надежная и умеет распознавать почти все элементы оборудования.»

I PC-BSD ★★★★★

Сайт: www.pcbbsd.org Лицензия: BSD Версия: 10.1.2
» Наш первый выбор после Linux, если надо выбрать ОС с открытым кодом.

IV Chromium OS ★★★★★

Сайт: www.chromium.org/chromium-os Лицензия: BSD Версия: 41
» Браузер вместо полнофункциональной ОС? Извините, не в этот раз...

II Haiku ★★★★★

Сайт: www.haiku-os.org Лицензия: MIT Версия: Nightly
» Удивительно удобная в работе, супер-быстрая и стабильная ОС для ПК любого возраста.

V Debian GNU/Linux ★★★★★

Сайт: www.debian.org/ports/hurd Лицензия: GPL Версия: Hurd 0.6
» Столько лет — и все еще пребывает на ранней стадии разработки.

III OpenIndiana ★★★★★

Сайт: openindiana.org Лицензия: В основном CDDL Версия: 151a8
» Старый конь, который не портит борозды; правда, края у борозды неровные.

Обратная связь

Какая у вас любимая ОС с открытым кодом, если не Linux? Мы бы хотели услышать о вашем опыте. Пишите нам на lx.f.letters@futurenet.com.

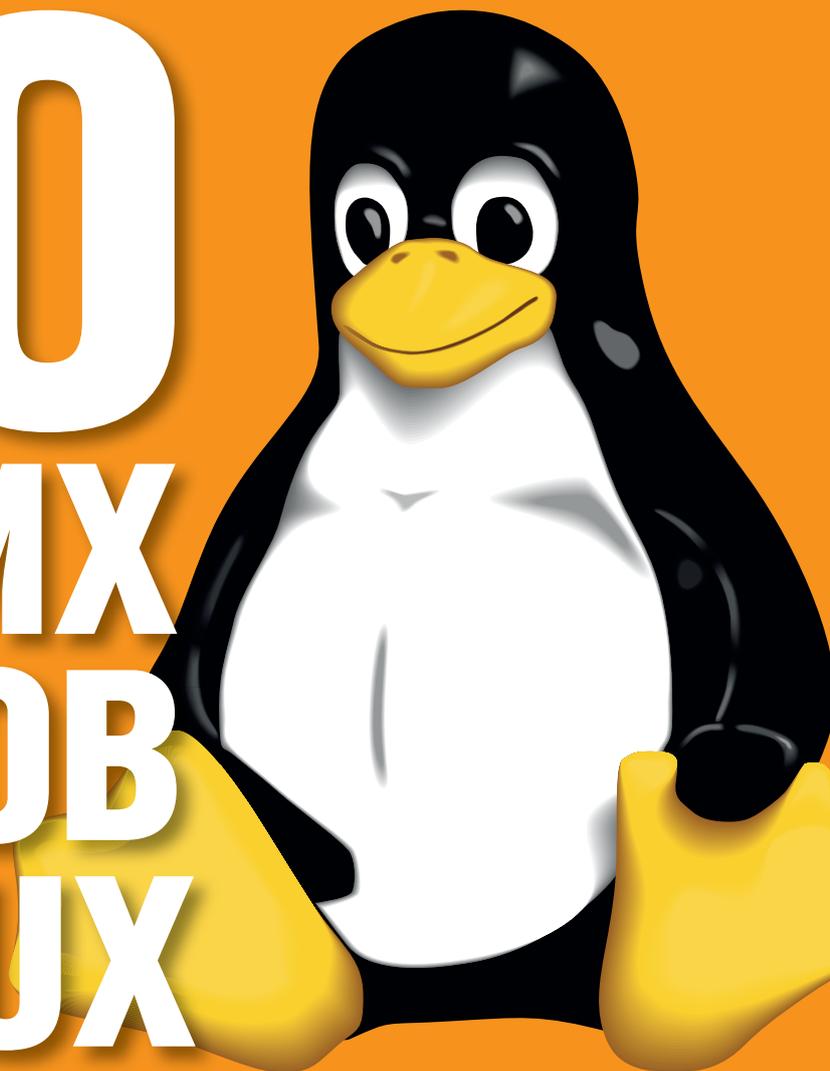
Рассмотрите также...

Мы могли бы включить множество разных операционных систем, но ограничимся упоминанием четырех. Если вам симпатично то, что делает Google, а среди наших знакомых таких не слишком много, почему бы не попробовать Android x86 (www.android-x86.org), неофициальный порт Android, который отлично работает на настольных ПК. В нем отсутствуют

недостатки Chromium OS, и вдобавок он всё-таки остается основанной на ядре Linux операционной системой. Android x86 может и не стать вашей любимой настольной системой, но он идеально работает с тысячами приложений Android, что может стать безболезненным выходом, если у вас нет возможности обзавестись смартфоном на Android.

Еще одна опция — ReactOS, клон Windows с открытым кодом. У нее множество преимуществ, хоть мы и боимся, что она может увести среднестатистического пользователя Linux в неверном направлении. Есть немалое количество других систем с открытым кодом, от крошечной KolibriOS до массивных ответвлений Darwin, так что смело исследуйте их все. LXF

200 ЛУЧШИХ СОВЕТОВ ПО LINUX



В честь 200-го выпуска *Linux Format* Маянк Шарма предлагает 200 своих лучших советов, собранных за годы исследований закоулков Linux.

Начало работы

Тестируем и устанавливаем дистрибутивы Linux, как профессионалы.

1 Создайте дистрибутив Live на устройстве постоянного хранения

Самые популярные дистрибутивы, вроде Fedora и Ubuntu, идут с инструментами, которые бронируют место на live USB-диске для хранения данных, которые будут доступны при последующих перезагрузках.

2 Поместите на диск несколько live-дистрибутивов

Если вы хотите протестировать несколько live-дистрибутивов, можете поместить их все на один

USB-накопитель с помощью скрипта *MultiCD* (который вы сможете отыскать на <http://multicd.us>) или инструмента *French MultiBoot LiveUSB* (<http://liveusb.info/dotclear>).

3 Используйте внешний инструмент разбиения диска на разделы

Хотя инструменты разбиения диска на разделы в дистрибутивах значительно улучшились и стали предоставлять больший контроль над вашим диском, лучше все-таки подготовить разделы для установки Linux с помощью

сторонних инструментов, типа *GParted*, который также предустановлен в live-версии некоторых дистрибутивов.

4 Используйте разделы LVM

Одно из крупнейших преимуществ использования LVM (Logical Volume Manager — менеджер логических томов) в том, что, в отличие от стандартных разделов, вам не придется рассчитывать разделы при установке, поскольку вы сможете увеличить (или уменьшить) объем LVM без потери данных.

Больше опций рабочего стола

Увеличьте продуктивность своего любимого рабочего стола.

5 Средняя кнопка для вставки

При выделении текста мышью тот копируется в особый буфер. По щелчку средней кнопкой в области ввода текста копия изначально выделенного текста вставится в поле ввода текста.

6 Определите клавиши быстрого запуска

Почти все популярные рабочие столы позволяют индивидуально задать клавиши быстрого запуска. Вы найдете эту опцию в настройках клавиатуры в соответствующей панели настройки.

7 Хитрости тачпада

Перемещайте палец вверх и вниз по правой стороне тачпада для вертикальной прокрутки, и стукните в нижнем правом углу, чтобы получить эффект щелчка правой кнопкой.

8 Включите рабочие области

В Ubuntu для этого перейдите в System Settings > Appearance > Behavior [Параметры системы > Вид > Поведение] и отметьте опцию Enable workspaces [Активировать рабочие области].

9 Установите Dock

На таких рабочих столах, как Gnome, ускорьте запуск любимых приложений, поместив их в Dock — скажем, легкий Cairo-Dock, он есть в репозиториях большинства дистрибутивов.

10 Контекстное меню менеджера файлов

В меню по щелчку правой кнопки (внутри менеджера файлов большинства настольных дистрибутивов) масса полезных опций, которые вы могли пропустить: например, отправка файлов электронной почтой, их сжатие или откат к более ранней версии.

11 Создайте Favourites

Перетащите любимые приложения из соответствующего вида в Ubuntu Launcher и Gnome Dash.

12 Разместите значки на рабочем столе

Чтобы изменить Gnome, установите удобный *Gnome Tweak Tool* из репозитория своего дистрибутива. Запустите приложение, перейдите во вкладку Desktop [Рабочий стол] и включите опцию Icons on Desktop [Значки на рабочем столе].

13 Меню Quick Launch

Щелкните правой кнопкой по значкам в Ubuntu Launcher или по имени приложения в верхней панели Gnome, чтобы увидеть индивидуальные опции и действия приложения.

14 Запускайте команды из меню Mint

Щелкните правой кнопкой по апплету Menu, выберите Configure > Open the menu editor [Настройка > Открыть редактор меню]. Затем выберите подменю или создайте новое и выберите New Item [Создать]. Введите команду в предоставленное поле и отметьте запуск в окошке терминала для приложений CLI.

15 Настройте поведение кнопки питания

Для настройки кнопки питания Power в основанном на GTK Cinnamon перейдите в System Settings > Power Management [Параметры системы > Управление питанием] и используйте выпадающее меню кнопки питания, чтобы выбрать ее отклик.

16 Измените структуру панели

Чтобы изменить раскладку панели Cinnamon по умолчанию, перейдите в Settings > Panel [Параметры > Панель] и используйте выпадающее меню Panel Layout [Раскладка панели] для выбора другого стиля.

17 Добавьте апплеты в панель

Cinnamon предлагает несколько интересных апплетов, и вы можете поместить их на любую панель, щелкнув по панели правой кнопкой и выбрав Add Applets [Добавить апплеты] в опциях Panel.

18 Включение наложения изображений

Для красоты включите наложение изображений в Mate, задав опцию Enable software compositing window manager в Control Center > Windows [Центр управления > Окна].

19 Разместите свои виджеты для каждого рабочего стола

Чтобы настроить виртуальные рабочие столы в KDE, щелкните правой кнопкой по Pager, перейдите во вкладку Virtual Desktops [Виртуальные рабочие столы] и включите опцию. Тогда на всех рабочих столах виджеты будут разные.

20 Запускайте приложения от имени другого пользователя

Чтобы запустить приложения в качестве другого пользователя (как, например, root) в KDE, щелкните правой кнопкой по значку меню и выберите Edit Applications [Редактировать приложения], выберите существующий пункт и щелкните по Copy [Копировать]. Потом перейдите туда, где вам нужен новый пункт, нажмите New Item, дайте ему название и нажмите Paste [Вставить]. Перейдите во вкладку Advanced [Дополнительно], включите опцию Run as a different user [Запуск под другим пользователем] и введите имя любого пользователя.

21 Обои в виде слайд-шоу

Щелкните правой кнопкой по рабочему столу KDE и нажмите Default Desktop Settings [Настройки рабочего стола по умолчанию]. В выпадающем меню Wallpaper [Обои] выберите опцию Slideshow [Слайд-шоу] и покажите ей набор изображений.

22 Растягивайте окно по горизонтали

Пользователи Xfce могут щелкнуть правой кнопкой по кнопке Maximise window [Максимизировать окно], чтобы горизонтально растянуть его по экрану.

Полезные сочетания клавиш

23 Alt+F
Выводит диалоговое окно Run [Запустить].

24 Alt
Поиск по меню приложений через HUD в Ubuntu.

25 Alt+~
Переключается между окнами в одном приложении.

26 Alt+Ctrl+стрелки
Переключается между рабочими областями.

27 Alt+PrintSc
Делает скриншот текущего окна.

28 Shift+Ctrl+Alt+r
Записывает скринкаст в Gnome.

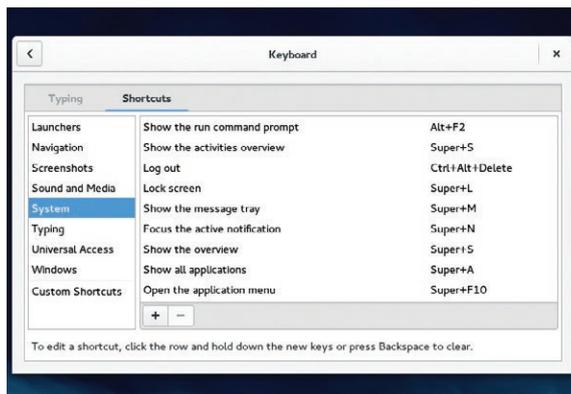
29 Super+↑
Максимизирует окна в Gnome.

30 Super+↓
Минимизирует окна в Gnome.

31 Super+← или →
Фиксирует окна в Gnome.

32 Super+m
Просматривает пропущенные уведомления в Gnome.

➤ **Настраивайте и используйте сочетания клавиш, чтобы сэкономить время поиска меню.**



Подсказки для любимых программ

Экономьте время и станьте продуктивнее благодаря этим скрытым сокровищам.

LibreOffice

33 Быстрая смена регистра
Выделите слова, щелкните правой кнопкой, перейдите в меню Change case [Сменить регистр] и выберите нужную опцию.

34 Включение автодополнение
Перейдите в Tools > AutoCorrect Options > Word Completion [Инструменты > Автоисправления > Автодополнение] и включите опции Enable word completion [Активировать автодополнение] и Collect words [Копить слова].

35 Управление клавиатурой
Войдите в Tools > Customise [...] > Индивидуальная настройка] и нажмите вкладку Keyboard [Клавиатура], для смены горячих клавиш.

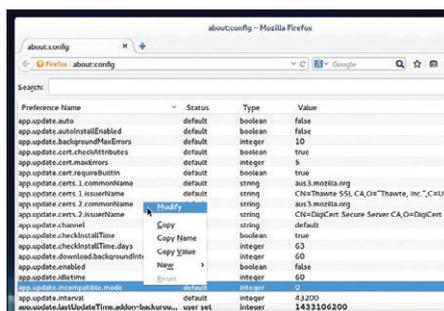
36 Воспроизведение мультимедиа-файлов
Войдите в Insert > Media > Audio or Video [Вставить > Медиа > Аудио или Видео] и выберите медиа-файл. Выделите значок мультимедиа в документе и включите управление мультимедиа.

37 Используйте Navigator
Быстрая навигация по любым документам или электронным таблицам — через окно навигатора во View > Navigator [Просмотр > Навигатор].

38 Автоформатирование таблиц
Выделите несколько ячеек и в Format > Autoformat [Формат > Автоформат] укажите тип форматирования. Всё делается автоматически.

39 Условное форматирование
Форматируйте ячейки согласно условиям из Format > Conditional Formatting > Condition [...] > Условное форматирование > Условие].

40 Защитите таблицу
Перейдите в Tools > Protect Document > Sheet [...] > Защита документа > Таблица], чтобы заблокировать доступ к таблице паролем.



С помощью функции about:config можно приподнять завесу и заглянуть внутрь любого приложения Mozilla.

41 Значения строки состояния
По умолчанию строка состояния показывает сумму значений выделенных ячеек. Измените это, щелкнув по строке состояния правой кнопкой.

Evince

42 Автоматическая прокрутка PDF
Щелкните правой кнопкой в документе, выберите опцию Autoscroll [Автопрокрутка] и контролируйте скорость прокрутки мышью.

43 Сделайте текст более читаемым
Зайдите во View > Inverted Colors [Просмотр > Инверсные цвета] для отображения белых букв на черном фоне.

44 Добавьте аннотации
Для этого выберите опцию Annotations [Аннотации] в выпадающем меню на боковой панели и перейдите во вкладку Add [Добавить].

Интернет-приложения

45 Ускорьте работу браузера (Firefox)
Введите about:config в адресной строке, затем network.http в поле фильтра, и установите параметры network.http.pipelining и network.http.proxy.pipelining в True.

46 Ограничьте использование ОЗУ (Firefox)
Если у вас 2 ГБ ОЗУ, войдите в about:config, просмотрите кэш браузера browser.cache и задайте параметр browser.cache.disk.capacity как 30000.

47 Восстановите папки (Thunderbird)
Щелкните правой кнопкой по поврежденной папке, перейдите в Properties [Свойства] и нажмите на кнопку Repair Folder [Восстановить папку].

48 Создайте список рассылки (Thunderbird)
Войдите в Tools > Address Book > New List [...] > Адресная книга > Новый список], укажите список, куда добавить адреса, и добавьте их.

49 Храните меньше почты локально (Thunderbird)
Ищите нужную учетную запись в Edit > Account Settings > Synchronisation & Storage [Изменить > Параметры учетной записи > Синхронизация & хранение]. Отметьте Synchronise the most recent [Синхронизировать недавнее] и укажите срок хранения.

50 Поиск по всем сообщениям (Thunderbird)
Чтобы осуществить поиск по всем сообщениям, включая полностью доступные только на сервере, войдите в Edit > Find > Search Messages [...] > Найти > Поиск сообщений] и отметьте опцию Run search on server [Искать на сервере].



Плагин Pidgin Autoaccept files переместит файлы в папку, выбранную в совете №54.

51 Вставить фоновое изображение (Evolution)
Выбрав опцию Format > HTML [Формат > HTML], войдите в Format > Page [...] > Страница], нажмите Browse [Просмотр] в разделе Background image [Фоновое изображение] и выберите изображение.

52 Расширенный поиск (Evolution)
Перейдите в Search > Advanced Search [Поиск > Расширенный поиск], чтобы создать правила сложного поиска. Параметры задавайте кнопкой Add Condition [Добавить условие].

53 Оптимизируйте скорость Torrent (Transmission)
Определите рекомендуемые настройки на <http://bit.ly/AzureuaUploadCalc>; затем их можно ввести во вкладки Edit > Preferences > Speed [...] > Предпочтения > Скорость] и Network [Сеть].

54 Мониторинг директорий (Transmission)
Перейдите в Edit > Preferences > Downloading [Изменить > ... > Закачки], отметьте опцию Automatically add .torrent files from [Автоматически добавлять файлы торрентов из] и выберите папку.

55 Управляем торрентами издали (Transmission)
Интерфейс Transmission, основанный на браузере, можно включить из Edit > Preferences > Remote [...] > ... > Удаленно].

56 Используйте приватный профиль (Firefox)
JonDoFox — профиль Firefox — автоматически интегрируется в установленный браузер, для анонимной работы в Интернете через прокси.

Медиа-плееры

57 Автоматом добыть субтитры (Gnome Videos)
Откройте диалоговое окно Movie Subtitles [Субтитры к фильму] — Ctrl+Shift+s, выберите язык и нажмите Find, чтобы просмотреть субтитры на сайте www.opensubtitles.org.

58 Конвертируйте медиа-файлы (VLC)
Перейдите в Media > Convert/Save [...] > Конвертировать/Сохранить], добавьте файл, нажмите на Convert/Save и выберите кодек, в который следует конвертировать.

59 Скачивайте онлайн-видео (VLC)

В Media > Open Network Stream [...] > Открыть сетевой поток] введите URL видео и в выпавшем меню Play [Воспроизведение] выберите Convert. Затем выделите ранее заданный Profile, введите имя сохраняемого файла и нажмите Start [Пуск].

60 Запишите рабочий стол (VLC)

Перейдите в Media > Convert/Save > Capture Device [...] > Захват устройства]. В выпадающем меню Capture mode [Режим захвата] выберите Desktop [Рабочий стол], а затем частоту кадров. Наконец, нажмите Convert/Save, задайте имя и нажмите Start.

61 Удаленный контроль VLC из браузера (VLC)

Перейдя в Tools > Preference, нажмите на кнопку All [Все] в настройках Show [Показать]. Затем в Interface > Main Interfaces [Интерфейс > Основные интерфейсы] и включите опцию Web. В Main Interface > Lua [...] > Lua] задайте пароль Lua HTTP Password.

62 Узнайте песню (Amarok)

Щелкните правой кнопкой по песне, которую не узнали, перейдите в Edit Track Details > Tags [Изменить данные трека > Тэги] и нажмите Get Tags from MusicBrainz [Получить тэг с MusicBrainz].

Редакторы изображений**63** Переместите маску выделения (GIMP)

Выделите объекты, затем нажмите на инструмент Move [Переместить]. Если опция Move настроена в панели на Selection [Выделение], то можно перетаскивать выделенное куда надо.

64 Скругленные углы (GIMP)

Перейдите в Filters > Decor > Rounded Corners [Фильтры > Украшения > Скругление углов]. Выберите Edge Radius [Радиус скругления], задающий кривизну, и настройте другие опции.

65 Массовая обработка изображений (GIMP)

Скачайте и установите плагин David Batch Processor (<http://bit.ly/DavidsBP>), чтобы активировать индивидуальную настройку.

66 Автоматически впишите метаданные в картинки (Shotwell)

Перейдите в Edit > Preferences и отметьте флажок Write tags, titles and other metadata to photo files [Писать тэги, названия и прочие метаданные в фотофайлы].

67 Упорядочивайте фото по событиям (Shotwell)

По умолчанию Shotwell объединяет все фотографии, загруженные за один раз, в одно событие. Для лучшей организации вы можете создать новые события из выделенной группы фотографий из Events > New Event [События > Новое событие].

68 Правильно делайте рендеринг RAW-файлов (Shotwell)

Чтобы Shotwell использовал проявитель RAW камеры, откройте изображение и отметьте опцию Photo > Developer > Camera [Фото > Проявитель > Камера].

Приложения KDE**69** Расположение закладок (Konsole)

Через меню Bookmarks [Закладки] добавляйте в закладки любую папку. Опция Bookmark Tabs as Folder [Вкладки в виде папки] добавит закладки на все открытые вкладки в одну папку.

70 Отметьте вкладки (Konsole)

Добавив в закладки несколько вкладок, которые вы используете постоянно, дайте им имя, дважды щелкнув по вкладке.

71 Запускайте команду в нескольких сессиях (Konsole)

Используйте Edit > Copy Input To All Tabs [...] > Копировать вывод во все вкладки] в Current Window [Текущее окно], или Select Tabs [Выбрать вкладки], если хотите запустить команду, например, на нескольких хостах SSH.

72 Мониторинг деятельности (Konsole)

Включите опцию View > Monitor for Activity [Просмотр > Следить за действиями], и KDE уведомит вас во всплывающем окне в панели задач, есть ли какая-то деятельность во вкладке Konsole.

73 Новая вкладка в индивидуальной директории (Konsole)

Перейдите в Settings > Edit Current profile [Параметры > Изменить Текущий профиль], отключите Start [Пуск] в той же директории, что и текущая опция вкладки, а затем введите расположение индивидуальной директории start в поле выше.

74 Создайте специальные профили (Konsole)

Можно создать новые профили со специальными шрифтами и разрешениями, перейдя в Settings > Manage Profiles > New Profile [...] > Управление профилями > Новый профиль]. Настройте его, перейдя в другие вкладки, например, Appearance [Вид].

75 Редактор только для чтения (Kate/Kwrite)

Включите опцию Tools > Read Only Mode [...] > Только для чтения], чтобы случайно не изменить важный документ.

76 Измените выделение (Kate/Kwrite)

Выберите должный режим выделения для открытого документа, перейдя в Tools > Highlighting [...] > Выделить].

VirtualBox**77** Создайте мгновенные снимки VM

Перейдите во вкладку Snapshot [Снимок] в главном интерфейсе и нажмите на кнопку Take Snapshot [Сделать снимок]. Потом можно восстановить снимки из этого интерфейса.

78 Используйте устройства USB

Перейдите в Devices > USB Devices [Устройства > Устройства USB] и выберите устройства USB, которые хотите присоединить и которые потом будут отключены от хоста и доступны в VM.

79 Перенаправляйте виртуальные порты

Настройте перенаправление портов, чтобы убедиться, что любое серверное ПО внутри VM доступно из Интернета, через Settings > Network > Advanced > Port Forwarding [...] > Сеть > Дополнительно > Перенаправление портов].

80 Включите удаленное отображение

Если вы запустите VirtualBox на сервере без клавиатуры, мыши и монитора, то сможете включить удаленное отображение, перейдя в Settings > Display > Remote Display [Параметры > Отображение > Удаленное отображение] и отметив флажок Enable Server [Активировать сервер].

81 Управляйте VirtualBox из браузера

Еще одно полезное приложение для управления VirtualBox с удаленного компьютера — phpVirtualBox, которое воссоздает интерфейс внутри web-браузера.

82 Расширенный доступ к буферу обмена

Если вы установили Guest Editions [Гостевое редактирование], включите соответствующую опцию в Devices > Shared Clipboard [Устройства > Общий буфер обмена] для копирования/вставки текста между гостем и хостом.

Полезные сочетания клавиш**83** F4 (KDE Dolphin)

Отображает встроенную командную строку.

84 F3 (KDE Dolphin)

Разделяет одно окно на два разных вида.

85 Ctrl+I (KDE Dolphin/Gnome Nautilus)

Отображает адресную строку, если она скрыта (Внимание: это L маленькое).

86 Shift+Enter (Gnome Nautilus)

Открывает выделенную папку в новой вкладке.

87 Ctrl+Shift+перетащить файл (Gnome Nautilus)

Создает гибкую ссылку на файл.

88 Пробел (Gnome Nautilus)

Предпросмотр выделенного файла, если установлена программа предпросмотра Sushi.

Управляем программами умнее

Используйте командную строку, чтобы повысить эффективность своего менеджера пакетов.

Советы для RPM/Yum/Fedora

89 Установите RPM с помощью `Yum`
Чтобы разрешить и скачать зависимости, установите пакеты RPM по `yum install <пакет.rpm>`.

90 Обновите определенный пакет
Пользуйтесь `yum check-update <пакет>` для проверки наличия обновлений пакета, которые можно установить с помощью `yum update <пакет>`.

91 Поиск пакетов
Имя пакета, содержащего указанный файл, выяснит `yum whatprovides <имя-файла>`.

92 Установите группы пакетов
Перечислите все имеющиеся группы с помощью `yum grouplist` и установите любые из них с помощью `yum groupinstall <имя-группы>`.

93 Отмена обновления
Получите список действий с их ID обновлений через `yum history` и отмените любое с помощью `yum history undo [update-id]`.

94 Разгоните Yum
Установите плагин `fastestmirror` по `yum install yum-plugin-fastestmirror` и всегда используйте ближайшее зеркало для установки пакета.

Советы для Apt/DPKG/Ubuntu/Mint

95 Создайте резервную копию списка пакетов
Чтобы установить те же самые пакеты на другую машину, создайте список установленных пакетов — `dpkg --get-selections > pkgs.list`.

96 Дублирование на другой системе
В свежей инсталляции сперва импортируйте список пакетов — `dpkg --set-selections <pkgs.list>`, а затем установите их — `apt-get dselect-upgrade`.



➤ Используйте инструмент **Additional Drivers** в Ubuntu для установки проприетарных драйверов для видеокарты и другого оборудования.

97 Удаление приложений
`apt-get remove --purge <app>` полностью удалит приложения вместе с их файлами настройки.

98 Использование более ранних версий пакетов, установленных из PPA
Установите инструмент PPA `purge` с помощью `apt-get install ppa-purge` и верните более раннюю версию обновленных пакетов через `ppa-purge <ppa-геро>`.

99 Установка библиотек разработчика
Для компиляции более новой версии приложения найдите библиотеку `dev lib` нужной версии в своих репозиториях с помощью `apt-get build-dep <app-name>`.

100 Удаление архивов
`apt-get autoclean` удалит скачанные архивы пакетов, которые уже обновились до более новых версий. Или избавьтесь от них по `apt-get clean`.

101 Удаление ненужных пакетов
Команда `apt-get autoremove` полностью стирает более не используемые зависимости.

102 Исправление поврежденных зависимостей
Используйте `apt-get -f install`, если у вас произошла ошибка при установке Deb-пакета, без предварительной установки его зависимостей.

103 Использование самого быстрого зеркала
В *Ubuntu Software & Updates* выберите **Other [Другое]** из **Download from menu** [Загрузить из меню] и нажмите на кнопку **Select best server** [Выбрать лучший сервер].

Советы для URPMI/Mageia

104 Скачайте список зависимостей
Команда `urpmi -d <pkg-name>` выдаст список требуемых зависимостей пакета.

105 Обновите медиа
Используйте `urpmi --auto-update` для обновления списка имеющихся пакетов.

106 Сохранение RPM
Присоедините опцию `--noclean`, чтобы не разрешить `urpmi` автоматически удалить скачанные RPM-пакеты после установки приложения.

107 Установка из локальной директории
Поместите RPM в директорию, а затем добавьте ее в качестве средства установки с помощью `urpmi.addmedia backup <directory>`.



➤ В Linux Mint имеются отличные специальные инструменты управления программами для облегчения управления зеркалами и PPA.

108 Установка из URL
Вы можете не скачивать пакеты, а установить их напрямую из сети с помощью `urpmi <URL-to-the-rpm>`.

Советы для ZYpp/openSUSE

109 Список установленных пакетов
Команда `rpmqack` отображает список всех установленных пакетов.

110 Обновление пакета
Используйте `zypper` в `<app-name>` для обновления пакета. Команда также установит пакет, если он еще не был установлен.

111 Ускорение с Zypper
Используйте `zypper sh` для входа в оболочку `Zypper`, которая установит пакеты быстрее, поскольку хранит все нужные данные в памяти.

112 Имитация обновления
Прежде чем обновлять свою инсталляцию, сделайте формальный прогон с помощью `zypper -v dup -D`.

113 Резервное копирование репозитория
Сохраните все настроенные репозитории посредством `zypper lr --export ~/backup-repos.repo`.

114 Восстановление репозитория
Используйте `zypper ar ~/backup-repos.геро` для восстановления репозитория из файла резервного копирования.

115 Просмотр необходимых заплаток
Скачайте список необходимых вам заплаток по `zypper lp`.

116 Установка заплаток
Обновите приложения, применив все имеющиеся заплатки с помощью `zypper patch`.

Советы для продвинутых

Станьте мастером в своей области.

Системное администрирование

117 **Мониторинг удаленных систем**
Запустите KDE KSysGuard и войдите в File > New Tab [Файл > Новая вкладка]. Затем переключитесь в новую вкладку, перейдите в File > Monitor Remote Machine [...] > Наблюдать за удаленной машиной] и введите IP-адрес и информацию о соединении целевой машины.

118 **Монтирование файлов ISO**
Используйте монтирование `-o loop <path-to-ISO-file> /tmp/iso-file` для изучения содержимого ISO-образа.

119 **Создайте виртуальные консоли**
С *tmux* можно создать несколько сессий, запустить разные задачи в каждой и затем переходить из сессии в сессию, не прерывая запущенных в них задач.

120 **Эффективное использование tar**
Архиватор *tar* может определять форматы сжатия, и все, что вам нужно для распаковки файла, это `tar xf <сжатый-файл>`.

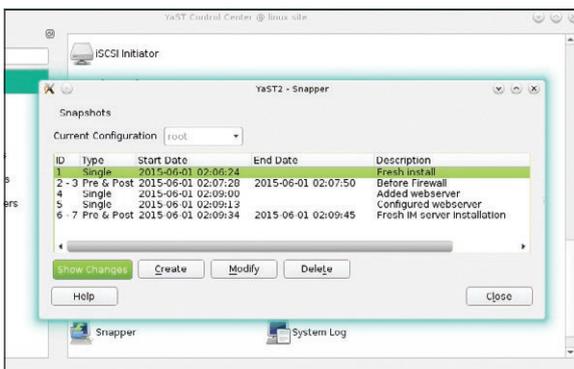
121 **Настройка одноразовых напоминаний**
Вы можете использовать *at* в `notify-send` для настройки краткосрочных напоминаний, таких, как `echo notify-send "Check on the tea" | at now +4 min`.

122 **Запланируйте запуск задачи в разные времена**
Времена можно перечислить в файле `crontab`. Скажем, `00 11,16 * * * <task>` выполнит задачу ежедневно в 11 и 16 ч. (11,16).

123 **Запустите задачу в течение указанного периода**
Аналогично, дефис укажет период: так, `00 10-17 * * 1-5 <task>` выполнит задачу с Пн по Пт (1-5) с 10 до 17 (10-17).

124 **Выполнение команды после каждой перезагрузки**
Используйте ключевое слово `@reboot` для запуска задачи при каждом запуске компьютера.

125 **Одновременный просмотр нескольких лог-файлов**
Вы можете установить из репозитория *multitail*, чтобы просматривать сразу несколько файлов — например, следующим образом: `multitail /var/log/syslog /var/log/boot.log`.



➤ Инструмент openSUSE *Snapper* помогает управлять мгновенными снимками *btrfs*, файловой системы дистрибутива.

Советы для Bash

126 **Просмотр команд, отвечающих шаблону**
Поиск по ранее выполненным командам, которые соответствуют шаблону, делается с помощью `history | grep -i <первые-несколько-букв-команды>`.

127 **Использование аргументов более ранней команды**
Используйте клавишу двоеточия (`:`) для повторного употребления опций из предыдущей команды: например, `!!:2` укажет на второй аргумент предыдущей команды.

128 **Просмотр команды перед исполнением**
Проверьте свои сложные инструкции *Bash*, добавив в конце `:p` — например, `ls -lltar:3:p`.

129 **Создание ярлыков для команд**
Вы можете включить часто повторяемые сложные команды в специальные команды через `alias`, например, `alias sshbox1='sudo ssh bodhi@192.168.3.111'`. Чтобы алиасы стали постоянными, добавьте их в файл `~/.bashrc`.

130 **Автоисправление опечаток CLI**
Вы можете использовать `shopt` для автоисправления обычных опечаток *Bash*, которые вы часто допускаете. Сначала введите `shopt` для отображения всех имеющихся шаблонов и включите их с помощью `shopt -s`. Например, `shopt -s cdspell` найдет ближайшее совпадение в неверно введенном имени директории.

131 **Создание файлов, которые трудно удалить**
Файл с начальным или завершающим пробелом в имени или с дефисом (`-`) нельзя случайно удалить из командной строки.

132 **Удаление трудноудаляемых файлов**
Когда вы создадите трудный для удаления файл, вам понадобятся способы удаления файлов со специфическими именами. Для этого вы можете заключить имя файла в кавычки или использовать двойной дефис, например, `rm "example"` или `rm --example`.

133 **Частичное удаление файлов**
Используйте оператор `!` для удаления всех файлов, кроме тех, которые отвечают указанному шаблону. Например, `rm !(*.txt)` удалит в директории все файлы, которые не заканчиваются на `.txt`.

Производительность

134 **Узнайте подробности об оборудовании**
Команда `dmidecode` выдаст подробную информацию об устройствах вашего компьютера. Например, `dmidecode -t 16` предоставит информацию о физической памяти. Загляните в `man`-страницу `dmidecode`, где вы найдете список поддерживаемых типов DMI.

135 **Иерархический список процессов**
Вы можете использовать `ps --forest` для представления древа процессов в ASCII и четкого различия родительских и дочерних процессов.

Клавиши быстрого запуска CLI

136 **Ctrl+a**
Отправляет курсор в начало команды.

137 **Ctrl+e**
Отправляет курсор в конец команды.

138 **Ctrl+I (L маленькое)**
Очищает экран, сохраняя то, что находится в текущей командной строке.

139 **Ctrl+k**
Вырезает текст, начиная с командной строки.

140 **Ctrl+y**
Сокращение для 'yank'. Вставка текста в буфер.

141 **Ctrl+Shift+c/v**
Копирует и вставляет текст в CLI.

Клавиши быстрого запуска Bash

142 **Shift+PgUp/PgDown**
Прокрутка консоли.

143 **Ctrl+r**
Поиск по истории команд.

144 **!`<номер-события>`**
Повтор команды из истории.

145 **!!**
Повтор последней команды.

146 **Alt+. (точка)**
Печать последнего аргумента предыдущей команды.

147 **><имя>**
Опустошает указанный файл.

148 Обнаружение утечек памяти

Чтобы выяснить, какие процессы съедают ОЗУ, используйте `ps --sort mem`, отсортировав процессы в обратном порядке по объему потребляемой памяти; главные обжоры — внизу списка.

149 Память определенного процесса

Подробный отчет о потреблении памяти определенным процессом выдаст `mpstat -x <PID>`; отобразится количество резидентной, анонимной с нераспределенным доступом и заблокированной памяти по каждому процессу.

150 Отслеживание исполнения бинарника

Если у вас есть неизвестный бинарник, отследите его исполнение с помощью `strace <имя-бинарника>` для просмотра всех системных вызовов и сигналов, которые он производит.

151 Отслеживание вошедших в систему пользователей

Используйте команду `w`, чтобы вывести список всех пользователей в системе и их процессов. Добавьте опцию `-f` для включения в результат имени хоста удаленных пользователей.

152 Выключение графического приложения

Введите `xkill` в терминал или запустите диалоговое окно, которое изменит указатель на крестообразный курсор. Теперь щелкните по любому окну, которое не отвечает, чтобы его выключить. Щелкните правой кнопкой, чтобы отменить `xkill` без выключения процесса.

153 Сократите использование подкачки

Если у вас достаточно ОЗУ, оптимизируйте использование `swap` [подкачки], отредактировав файл `/etc/sysctl.conf` и изменив значение параметра `vm.swappiness` на 10.

Резервное копирование

154 Резервное копирование загрузочного сектора

Резервная копия загрузочного сектора окажется весьма кстати, если вы случайно сотрете свою MBR. Создайте резервную копию здорового загрузочного сектора с помощью `dd if=/dev/sda of=disk.mbr count=1 bs=512` и восстанавливайте ее с помощью `dd if=disk.mbr of=/dev/sda`.

➤ Используйте *Nethogs* для просмотра в реальном времени поглощаемого приложением трафика.



➤ Mageia также включает Родительский контроль для ограничения определенными приложениями и контроля за временем.

155 Резервная копия таблицы разделов

У вас также должна быть резервная копия таблицы разделов на случай, если эта важная информация будет уничтожена. `sfdisk -d /dev/sda > disk.sf` сделает резервную копию таблицы разделов, а `sfdisk /dev/sda < disk.sf` ее восстановит.

156 Мониторинг прогресса dd

Установите инструмент *Pipe Viewer (pv)* из репозитория своего дистрибутива и используйте его для мониторинга `dd`. Например, `pv -tpreb some-distro.iso | sudo dd of=/dev/sdb bs=4096`.

157 Увеличьте скорость резервного копирования на медленных машинах

Если скорость соединения — не проблема, используйте `rsync -W` для передачи целых файлов и экономии времени на подсчет измененных блоков и байтов.

158 Отслеживайте работу rsync

Добавьте опцию `--progress` в команду `rsync`, чтобы приглядывать за передачей данных.

159 Просмотр изменений между исходным пунктом и пунктом назначения

Используйте опцию `-i` для просмотра списка пунктов, измененных операцией `rsync`, например, `rsync -avzi [source] [destination]`.

160 Используйте rsync через ssh

Для передачи данных `rsync` через SSH используйте опцию `-e ssh`, например, `rsync -avhze ssh [source] [destination]`.

161 Исключение файлов

`Rsync` также позволяет пропускать определенные файлы, которые можно указать с помощью опции `--exclude`: например, `rsync -avhz --exclude '*.*tmp*'` будет игнорировать файлы с расширением `.tmp`.

162 Тест rsync

Пользователи, запускающие `rsync` впервые, должны перед настоящим запуском добавлять опцию `--dry-run` ко всем операциям `rsync` и проверять результат на предмет непредвиденных выводов.

163 Ограничение скорости соединения

Чтобы не позволить операциям `rsync` отобрать весь трафик, ограничьте его использование с помощью опции `--bwlimit`, вот так: `rsync -avhz --bwlimit=50`.

164 Не делайте резервных копий файлов на внешних системах

`Tar` — популярный выбор для создания архива диска. Используйте с `tar` опцию `--one-file-system`, чтобы убедиться, что он не сделал копии монтированных разделов (`/media`) или виртуальных разделов (`/proc`, `/sys`).

Безопасность и брандмауэр

165 Выясните, на каком порте работает программа

Используйте `netstat -ap | grep [app-name]`, чтобы увидеть список портов, с которыми работает определенное приложение.

166 Отключите ответ ping

Ping можно использовать для лавинной маршрутизации сети и вызвать перегрузку сети. Отключите его временно с помощью `echo "1" > /proc/sys/net/ipv4/icmp_echo_ignore_all` или постоянно, отредактировав файл `/etc/sysctl.conf`, добавив в него `net.ipv4.icmp_echo_ignore_all = 1`.

167 Резервное копирование iptables

Если у вас ушло много сил на настройку брандмауэра ядра `iptables`, сделайте его резервную копию: `iptables-save > ~/iptables.backup`.

168 Заблокируйте определенный домен

Сначала выясните IP-адрес домена с помощью `host -t a www.example.com`. Затем используйте IP-адрес, чтобы получить его CIDR через `whois [IP Address] | grep CIDR`. Затем используйте CIDR, чтобы заблокировать доступ к домену, например: `iptables -A OUTPUT -p tcp -d [CIDR] -j DROP`.

169 Смена пароля для любого пользователя

Если вы забыли пароль какого-то пользователя, можете задать новый с помощью `sudo passwd [username]`, причем у вас не спросят старый пароль.

170 Дублируйте разрешения

Используйте опцию `--reference`, чтобы скопировать разрешения одного файла в другой,

PID	USER	PROGRAM	DEV	SENT	RECEIVED
7639	bodhi	wget	wlan0	4.512	204.180 KB/sec
2531	bodhi	..ope-home/unity-scope-home	wlan0	0.255	1.016 KB/sec
7666	bodhi	transmission-gtk	wlan0	0.000	0.000 KB/sec
?	root	..07:37918-192.168.3.1:1900		0.000	0.000 KB/sec
?	root	..07:37917-192.168.3.1:1900		0.000	0.000 KB/sec
?	root	..07:37916-192.168.3.1:1900		0.000	0.000 KB/sec
?	root	..7:41715-192.168.3.1:49152		0.000	0.000 KB/sec
?	root	..7:41714-192.168.3.1:49152		0.000	0.000 KB/sec
?	root	..:35104-173.194.36.103:443		0.000	0.000 KB/sec
?	root	..:35570-157.19.134.147:443		0.000	0.000 KB/sec
?	root	..:33455-173.194.36.120:443		0.028	0.000 KB/sec
?	root	..:39280-173.194.36.114:443		0.000	0.000 KB/sec
?	root	..7:49652-74.125.130.95:443		0.000	0.000 KB/sec
6595	root	ssh	wlan0	0.000	0.000 KB/sec
?	root	unknown TCP		0.000	0.000 KB/sec
TOTAL				4.795	205.196 KB/sec

например, `chmod --reference=[copypermission-from-this-file] [apply-on-this-file]`.

171 Безопасное удаление файлов
Установите и используйте утилиту `shred` для удаления файлов без возможности их восстановления. Например, `shred [file]` несколько раз перезапишет блок файла случайными данными.

172 Включите встроенный брандмауэр
Некоторые дистрибутивы, такие, как Ubuntu, идут с более простым интерфейсом брандмауэра `iptables`, под названием `UFW`. По умолчанию он отключен, но вы можете включить его через `ufw enable`.

173 Разрешение входящего соединения
По умолчанию `UFW` определяет все входящие соединения. Чтобы настроить эту политику и разрешить соединения для общих серверов, скомандуйте `ufw allow ssh, sudo ufw allow www, ftp`.

Сеть и Интернет

174 Удаленный запуск команд
Вы можете также использовать SSH для запуска команд на удаленной машине, например: `ssh [hostname] [command]`.

175 Копирование ключей SSH на другую машину
Используйте `ssh-copy-id [remote-host]` для безопасного копирования публичного ключа вашей идентификационной информации по умолчанию на удаленный хост.

176 Оставьте соединение открытым
Если вы часто отсоединяетесь от удаленных сессий SSH из-за отсутствия активности, можете включить опцию `KeepAlive`, добавив строку `ServerAliveInterval 60` в файл `/etc/ssh/ssh-config`.

177 Работа через канал SSH
Сначала создайте канал SSH к удаленному хосту с помощью `ssh -f -N -D 1080 user@remotehost`, затем измените настройки прокси своего браузера и установите хост SOCKS в `127.0.0.1`, а порт в `1080`.

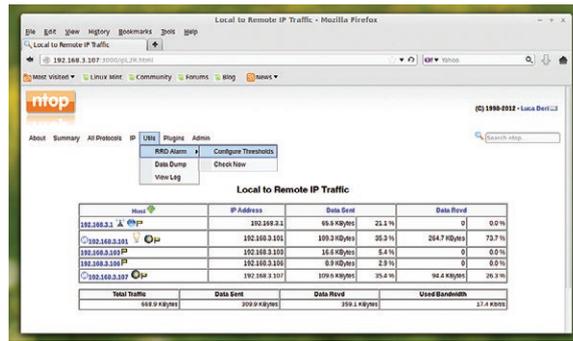
178 Воспроизводите музыку через SSH
Команда `ssh user@remotehost cat ~/Music/audio.ogg | mplayer` переадресует удаленный медиа-файл на `mplayer` локальной машины.

179 Монтируйте разделы через SSH
Используйте `sshfs` для монтирования удаленных разделов — например, `sshfs user@remotehost:/home/bodhi /media/remotefs` смонтирует удаленную директорию `home` в локальной файловой системе.

180 Улучшайте мониторинг сетевого трафика
`Ntop` доступен в официальных репозиториях большинства дистрибутивов, и он дает вам подробный анализ сетевого трафика через свой сетевой интерфейс, работающий на порте 3000.

181 Просмотр сетевой статистики
Используйте `netstat -s` для просмотра статистики для всех протоколов или `netstat -st` только для протокола TCP.

182 Сохранение web-страницы
Используйте для скачивания web-страницы `wget` — например, `wget -r -np -k http://www.tuxradar.com/content/dearedward-`



`snowden` скачает все изображения и изменит ссылки в файлах HTML и CSS для указания на локальные файлы.

183 Сохранение нескольких файлов
Если вы сохранили в файле ссылки на несколько скачиваний, используйте `cat isos.txt | xargs wget -c` для скачивания их всех.

184 Ограничение скорости передачи данных
Не позволяйте `wget` захватить весь трафик, установив ограничения; например, `wget --limit-rate=2m` ограничит скорость передачи двумя мегабайтами в секунду.

185 Скачивайте файлы по дате изменения
Используйте `curl` с опцией `-z` для скачивания только тех файлов, которые были изменены после определенного времени. Например, `curl -z 29-May-2015 [download-location]`.

186 Загрузка файлов
Вы можете использовать `curl` для соединения с FTP-сервером и загрузки файлов, например: `curl -u [user:pass] -T upload.txt ftp://ftp.example.com`.

187 Найдите определения
`Curl` умеет добывать определение слова на сервере директории. Перечислите все слова по `curl dict://dict.org/show:db` и сделайте запрос с помощью `curl dict://dict.org/d:shell:foldoc`; в результате вы найдете определение слова "shell" в словаре `Foldoc`.

188 Простая фильтрация Сети
Чтобы ваш компьютер не получил доступа к сайту, введите его URL в `/etc/hosts`, например, `127.0.0.1 www.addictivewebsite.com`.

189 Создайте зеркала сайтов
Используйте графический инструмент `WebHTTrack`, имеющийся в официальных репозиториях большинства дистрибутивов, для создания зеркал целых сайтов и автоматического изменения ссылок.

190 Регулируйте скорость соединения
Для управления скоростью приема и передачи можно использовать `Trickle`, легковесный генератор скорости соединения области пользователя. Он также умеет регулировать скорость для менеджеров пакетов, таким образом: `trickle -d200 apt-get install`.

191 Мониторинг скорости соединения
Для мониторинга полосы пропускания, используемой отдельными сетевыми приложениями, используйте `nethogs`, небольшой сетевой инструмент, имеющийся в репозиториях большинства дистрибутивов. Вместо того, чтобы разбивать трафик по протоколам, он группирует полосу по процессам. **LXF**

Ntop — универсальный инструмент, расширяемый плагинами.

Горячие клавиши команд Top

192 Shift+m
Сортировка по использованию ОЗУ.

193 k
Отключите задачу из `top`.

194 1
Отслеживайте всю оперативную память из `top`.

195 Shift+w
Сохраняет измененную настройку на постоянной основе.

Горячие клавиши команд less

196 /
Поиск шаблона.

197 n
Ищет следующее совпадение.

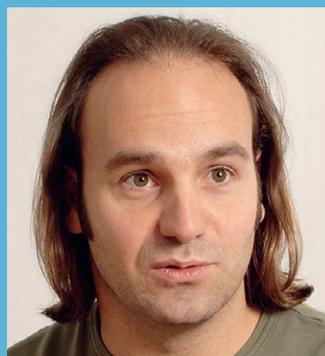
198 Shift+f
Отображает новый контент, добавленный в файл.

199 v
Редактирует файл с помощью настроенного редактора вашей системы.

200 h
Отображает список всех горячих клавиш.

Герои Open Source

За 15 лет *Linux Format* успел пообщаться с первыми лицами мира Open Source, и вот некоторые фрагменты...



**Ричард Столлмен
[Richard Stallman]**

Создатель GNU
Linux Format №145,
июнь 2011 г.

Linux Format: Большин-
шинство наших читате-

лей — убежденные сторонники свободного ПО...

Ричард М. Столлмен: А что они понимают под «свободным»?

LXF: Ну, когда они с нами общаются, многие используют термин «свободное ПО» — а некоторые говорят «открытый код»...

PMС: А, видите, тут есть разница. Открытый код относится к другим идеям — к другой философии. А разница фундаментальная, она лежит на уровне ценностей. Это не разногласие в мелочах; это разногласие в основе. Наша цель — свободное общество, где пользователи обладают свободой. Организаторы и лидеры открытого кода говорят, что их цель — лучшее качество кода. Разойтись дальше просто некуда: мы говорим

о свободе и общественной солидарности, а они говорят о качестве.

LXF: Однако разве это не способ увести людей от одной системы и привести к другой?

PMС: Я вас не понимаю — это разные вещи.

LXF: Ну, допустим, ментальность компании, создающей проприетарное ПО, изменить в сторону GPL и свободного ПО непросто. А если свети к идее открытого кода через объяснение преимуществ его качества — идея будет воспринята и станет привычной, и можно будет ее расширить...

PMС: Это не одно и то же, и первое обычно не приводит ко второму. Фактически, когда широко распространяется философия открытого кода — что и произошло — она склонна отвращать умы от идей свободного ПО. И имеет тенденцию скрывать наше существование. Большинство статей о системе GNU даже не называют ее системой GNU или свободным ПО. Они описывают ее так, словно это открытый код, и создают впечатление, что мы — ее разработчики — согласны с идеями

открытого кода, о которых читатели уже слышали и никогда не догадаются, за что мы на самом деле ратуем.

LXF: Услышав «свободное ПО», некоторые представляют себе мелкие шпионские программки в Windows.

PMС: Мне потребовалось время на осознание колоссальной важности этой разницы. В 1993 году, когда я сделал объявление [о GNU], я не разделял эти концепции. Прошло несколько лет, прежде чем я их разделил. И так, еще раз, в *Манифесте GNU*, появившемся в 1985 году, все еще смешиваются два значения слова “free”. Тогда я и осознал необходимость подчеркнуть, что имеется в виду свободное в смысле свободы, а не бесплатное в смысле цены. Думайте о свободе слова, а не о бесплатном пиве. Конечно, было бы лучше, осознай я это раньше. Хотя тогда я сказал бы в точности следующее: разночтения присутствуют по той причине, что в английском языке нет слова, однозначно выражающего именно то, что я хочу сказать. Единственное слово в значении «свободный», то есть имеющий свободу — это “free”,

именно поэтому мы говорим “free/libre”, потому что таким образом мы можем уточнить смысл.

Я заметил в вашем журнале [LXF143] утверждение о LibreOffice, которое является важной иллюстрацией. Sun приобрел StarOffice и выпустил его как свободное ПО под названием OpenOffice.org. Но люди из Sun, которые это сделали, не были сторонниками идей свободного ПО в политическом смысле. По сути, они были сторонниками открытого кода. И их целью было сделать их программу качественной и успешной — а не предоставить пользователям свободу. Это их целью не являлось; хотя их код был свободным ПО и уважал свободу пользователя, но они об этом не думали в таком смысле. И они сделали список расширений, внося в него проприетарные расширения.

И что я мог тут сделать? Мы предложили: давайте сделаем свой список расширений. LibreOffice использует наш список расширений — они его приняли от нас. Это решило проблему, а причина, по которой они это сделали, в том, что люди, создающие данную версию программы — активисты свободного ПО: им важна свобода. Они принимают решения ради свободы. Это показывает, что те, кто не думает о свободе или не ценит свободу, порой делают что-то не из желания содействовать делу свободы.

LXF: Кругом полно этих чисто свободных дистрибутивов GNU/Linux, вроде Trisquel и gNewSense, и многие из них сдают позиции — это очень разрозненные проекты. Есть ли место для официального GNU? GNU/Linux от GNU?

РМС: Я полагаю, было бы неплохо, если бы такие проекты начали работать совместно. Но я не хочу затевать еще один дистрибутив GNU — это станет пощечиной всем тем, кто сейчас работает над этими дистрибутивами, и я не хочу принимать чью-то сторону среди этих людей, отдавать кому-то предпочтение. Это было бы довольно неуместно.

LXF: Многие из наших читателей хотят знать, чем именно вы занимаетесь, для примера. На вашем сайте есть фотографии, где вы работаете с ThinkPad, но сейчас вы его не рекомендуете.

РМС: Я не пользуюсь ThinkPad — это фотографии годичной давности. Сейчас у меня вот это устройство от Lemote — Yeeloong — можете называть его «remote», с китайским акцентом! [точнее, японским, — прим. ред.] Я выбрал это устройство, потому что оно свободно целиком, вплоть до BIOS. У него процессор типа MIPS, в китайской версии. В любом случае, суть в том, что он решает проблему.

LXF: Возвращаясь к более глобальным вещам, что бы вы назвали главной угрозой СПО в 2011?

РМС: Их несколько. Юридические запреты, такие, например, как софтверные патенты в некоторых странах с неумной политикой. И еще есть законы, явно выступающие в качестве цензоров свободного ПО, например, Digital Millennium Copyright Act в США, контролирующий свободное ПО, способное сломать цифровые наручники. В Европейском Союзе тоже есть подобные законы. И США, и ЕС пытаются протолкнуть свои мерзопакостные законы



LXF: Намного больше людей сейчас используют смартфоны и планшеты в качестве компьютерных платформ, со всеми их магазинами приложений...

РМС: Смартфон — это компьютер: он не создан с помощью компьютера; он создан, чтобы быть компьютером. Так что всё, что мы говорим о компьютерах — что программы, с которыми вы работаете, должны быть свободными, и вам надо настаивать на этом — применимо в равной мере и к смартфонам. И к планшетам тоже. Что сказать по поводу этих магазинов приложений? Ну, во-первых, магазины приложений Apple и Microsoft запрещают свободное ПО и разрешают только несвободные программы. И это доказывает их вредность.

А Android — иной случай. Исходный код Android — свободный, в том виде, как его выпустил Google, но они используют лицензию, не являю-

О ЗНАЧЕНИЯХ СЛОВА “FREE”

«Имеется в виду свободное, а не бесплатное. Думайте о свободе слова, а не о бесплатном пиве.»

в другие страны, через договоры, которые этим странам предлагается подписать. Так что это — вредные правительства.

А еще есть препоны, чинимые производителями, сотрудничающими с Microsoft. Есть масса примеров оборудования для ПК, работающего только с Windows. И, как правило, спецификации этого оборудования недоступны — безусловно, неэтично продавать продукт, отказываясь сообщить, как он работает. Это недопустимо. Еще одно препятствие — тенденция продавать компьютеры вместе с Windows. Я бы рекомендовал запретить и эту практику. И еще у некоторых компаний есть тенденция безвозмездно — или почти безвозмездно — передавать в дар школам свои программы, не относящиеся к СПО. Это делают и Microsoft, и Apple — и я читал, что Фонд Гейтса [Gates Foundation] тоже этим занимается. Идея благотворительности Билла Гейтса — посадить школьников на Windows и зашибить еще больше денег. По-моему, это уже не благотворительность.

LXF: Один из наших читателей задал вопрос: возможен ли мир, где есть только свободное ПО? Это по-прежнему является главной целью?

РМС: Да, полагаю, цель именно такова. Это моя цель. Может быть, совсем избавиться от пережитков несвободного ПО нельзя. Ведь, в конце концов, за 200 лет аболиционизма мы так и не искоренили рабство. Есть места, где люди, по сути, рабы. Я читал, что иностранные рабочие в Великобритании, по сути, являются рабами, поскольку если они вздумают жаловаться, их просто депортируют. Так что полностью устранить все формы злоупотребления трудно, но я уверен, что общество, где проприетарное ПО является исключением, вполне возможно, если мы все вместе будем за него бороться.

щуюся копиелфтом, кроме Linux, который идет под GPL v2. И в результате лицензия не защищает пользователя от запертости или тивоизации [Tivoization — практика создания аппаратно-программных систем, у которых ПО имеет копиелфт-лицензию, но аппаратура не дает запускать модифицированную версию ПО, — прим. пер.] — то есть от практики превращения свободной программы в несвободную, запрещающую пользователю установить и использовать собственную версию. Очень многие разновидности смартфонов с Android блокируют установку иной версии программ.

LXF: Например, мои родители — это как раз случай, когда надо попытаться найти правильный подход...

РМС: Я использую аналогию с рецептами. Это хорошая аналогия, потому что программа во многом похожа на рецепт. И то, и другое — определенная последовательность действий для получения желаемого результата. И если вы посмотрите, как повара используют рецепты, то увидите, что на практике они пользуются теми же четырьмя свободами. Повара готовят по рецептам свободно, они изучают и изменяют их, когда хотят, они распространяют копии; и если они вносят свои изменения, то могут поделиться измененной версией. И так, представьте себе, что бизнесмены и государство решили навязать проприетарные рецепты.

Предположим, государство заявило: с завтрашнего дня, если вы скопируете или измените свой рецепт, мы посадим вас в тюрьму и объявим пиратом. Представьте себе, как обозлятся повара. Многие из тех, кто ничего не знает о программировании, поймут это. Государство не пыталось поступить так с рецептами — зато попыталось сделать с программами. »



Линус Торвалдс [Linus Torvalds]

Создатель Linux
Linux Format №163,
ноябрь 2012 г.

Linux Format: Рабочий стол Linux терпит крах из-за слишком большого выбора? [Смешок.]

Линус Торвалдс: Не думаю, что у рабочего стола дела идут хорошо, и на то есть технические причины. Возможно, вы слышали мои тирады о том, что рабочий стол в известной степени движется не туда; но главная причина в том, что обычные люди не хотят устанавливать операционную систему. У вас не будет рабочего стола, пока не будет предустановки, а этого нет.

Бывают случаи, когда, зная, куда смотреть, вы получите предустановленный Linux, если приобретете Dell. Но в реальности, предустановки никто не сделал.

LXF: А если бы команды KDE и Gnome не приложили столько усилий к созданию провальных первых версий своих рабочих столов?

ЛТ: Я знаю людей, которые решили отказаться от рабочего стола Linux, хоть они и технари, просто потому что их достали Gnome и KDE, так что негативный момент был. Но даже если бы такого не было, не думаю, чтобы нормальные... бабушки, люди, которые на самом деле не очень любят компьютеры, перешли бы на настольный Linux. Меня совершенно не радует то, что сделали Gnome и KDE, но если брать картину в целом, я думаю, это очень мелкая деталь.

LXF: Почему вы не используете торговую марку Linux для создания среды Linux по умолчанию?

ЛТ: Мне это не интересно. Никогда не хотел ничего делать сверх технической стороны. Я совершенно счастлив пожаловаться, потому что это очищает, и я совершенно счастлив поспорить в Интернет, потому что мое любимое занятие — именно спор, а вовсе не программирование. И в то же время, торговая марка — это то, чего мне меньше всего хочется иметь, потому что это большая головная боль. Это становится проблемой с самого начала, когда у нас появляется владелец всей торговой марки, и торговые марки — это до нелепости плохо. И, по сути, юридическая ситуация с торговыми марками заставляет корпорации дурковать, раз их юристы полагают, что не дуркуя, они утратят контроль над торговой маркой; а это, естественно, чушь, но юристам и платят за создание геммороя. Так что мне совершенно не хочется использовать торговую марку.

LXF: Мы читали, что вы по большей части проводите время, подтверждая изменения ядра.

Как вам удается сохранять энтузиазм, когда речь уже не идет о кодировании?

ЛТ: Большинство моих коммитов технически занимают пару секунд. У меня больше времени уходит на чтение и копирование и вставку адреса *Git*, и на то, чтобы прочитать, что там происходит...

LXF: И этого достаточно?

ЛТ: Этого достаточно для большей части кода. Меня куда больше волнует... ну, я хочу сказать, что по-настоящему меня расстраивает, когда кто-то делает глупость... код, конечно, важен, но то, чем я занимаюсь сейчас, уже не код, а организация производственного процесса для других. И когда кто-то делает широкомасштабную дурь, то я действительно волнуюсь, а под волнением я подразумеваю, что начинаю крепко ругать людей.

LXF: Что произойдет, когда Google выкинет свой большой Android, как он сделал в начале этого года после отделения, вы будете начеку?

ЛТ: Я полагал, что споров будет больше. Мы это обсуждали до того, как все произошло, и я хочу сказать, что многие проблемы Google с Android были не столько проблемами Google, сколько проблемами разработчиков ядра. Android на деле очень хорош, он работает, и ни у кого по сути не было альтернативного рабочего кода, приемлемого для Google, потому что они решали проблемы, которые никто за них не решил бы.

LXF: На Google оказывали давление, чтобы они пересобрали разрабатываемое ими ядро Android?

ЛТ: Разработчикам ядра в Google был ненавистен факт наличия собственных патчей. Им прежде всего не нравится работать вне ядра, и еще они этого терпеть не могли, потому что это лишняя работа.

LXF: Выходит, ответственность за Google должны принять на себя отдельные подсистемы?

ЛТ: Программы Google работают, большинство из них, и что нам, например [*Линус тянется к телефону, на который мы записываем интервью*]... «О, это iPhone... вон отсюда! Мне с вами говорить не о чем».

Среди прочих, у Android есть проблема с пробуждением системы, когда с позиций системы им надо, чтобы она продолжала спать; но при этом им приходится быть очень осторожными с приложением, которое вот-вот выведет систему из спящего режима.

Она не очень ладит с некоторыми другими частями кода управления энергопотреблением, и не очень-то ложится в теорию. И она влияет на многие драйверы. Это просто такая цепная реакция получения интерфейса, нужного Google, который они сделали не так, как в стандартном ядре, и потом это оказало свое воздействие на создателей драйверов, которые используют интерфейс, которого и духу не было в стандартном ядре.

Вообще-то никто не возражает против указаний, что им делать. Возражения были против самого разделения, породившего проблемы без всякой на то реальной причины, кроме незначительных разногласий. И тогда мы решили: «Да и плевать, может, нам и не все нравится из того, что делает Google и как он это делает», но, с другой стороны, настоящей ненависти тоже не было...

ГОД РАБОЧЕГО СТОЛА

«Главная проблема в том, что обычные люди не хотят устанавливать ОС.»





Марк Шаттлворт [Mark Shuttleworth]

Основатель Ubuntu
Linux Format № 71,
октябрь 2005 г.

Linux Format: Что при-
вело вас к тому, чтобы

начать проект Ubuntu?

Марк Шаттлворт: Многое. Во-первых, острое желание дать что-то обществу. Я был невероятно удачлив на пике популярности Интернет-компаний [Прим.: — В 1999 году он продал свою компанию “Thawte” фирме “VeriSign” за \$575 миллионов], и причиной такой удачливости был тот факт, что в нужный момент я был увлечен открытым программным обеспечением. Я был в состоянии построить Thawte на Linux, и на MySQL, и на Apache. У меня была очень мощная благотворительная программа — отчасти, я даже испытывал некую потребность избавиться от всего, что я приобрел. Я был одержим сделать это на своем веку, в наиболее подходящий для этого момент времени. Таким образом, я взялся за дело.

Для меня это означает, что благотворительность можно направить в стоящее предприятие. И пускай это инвестирование сыграет небольшую роль, но это хоть как-то повлияет на будущее состояние индустрии программного обеспечения. Я не могу сказать, как эта отрасль будет выглядеть в будущем — потому что на данный момент слишком много спрятано в некоем «тумане войны».

LXF: Но ведь существует уже достаточно много различных дистрибутивов Linux. Не имеет ли смысл просто направить один из них в сторону ваших взглядов?

МШ: Только один из них, который, как я полагал, будет действительно непревзойденным, не был управляемым! Всем известно, что мы делаем не что иное, как продукт, который я долгое время обдумывал. Я полагал стать лидером проекта Debian, но осознал, что есть другой путь, чтобы достичь того же эффекта — создать что-то новое, идеально соответствующее моей концепции. Сделать эти идеи полностью свободными и доступными, и позволить другим людям брать из этого все, что они пожелают — до последнего бита.

LXF: Да, он действительно популярен, казалось, он появился просто ниоткуда...

МШ: Ну, если сравнить это с другими рискованными проектами, то мы просто не могли пойти на компромисс с некоторыми техническими требованиями. И я дал команде очень свободное управление, таким образом, чтобы она могла бы технически двигаться в том направлении, в котором хочет.

В этом плане, построение Ubuntu на базе Debian пришлось очень кстати. Нам повезло, что с самого начала все сложилось так благополучно. Другая группа, которая, как мне кажется, считает открытые проекты действительно привлекательными, прямо противоположна первой. Это люди, которые знают о компьютерах совсем немного и не хотят знать ничего сложного.

LXF: Некоторые говорят, что Ubuntu достиг столь большого успеха только за счет Debian. Вы думаете, это справедливая критика? Имеет ли это какое-то значение?

МШ: Да, я согласен, что это имеет значение, если люди так думают, потому что Debian необходим мне для успеха, а также для того, чтобы люди видели, какие конструктивные изменения мы предлагаем. Так что, это меня очень волнует, когда бы и где бы это ни было сказано. В то же время, когда я начинал свою работу над этим проектом, я отлично понимал, что сообщество отнесется к моей затее скептически, так как такая реакция там в порядке вещей.

Одна из причин, почему я решил этого больше не делать с Debian — я твердо верил в то, что нет ничего, что не могла бы сделать команда Open Source или сообщество, разве что сделать невозможное. Не существует такой горы, на которую бы не смогла взобраться группа Open Source, но в то же время они не могут быть на пике всех гор одновременно. Вся красота Open Source заключается в том, что мы можем посылать команды взбираться на разные горы. Мы можем одержать победу сразу в двух битвах. Таким образом, если Ubuntu и теряет что-то от Debian, то это происходит только для того, чтобы приобрести что-нибудь в другом месте.

LXF: Он помещается на один диск...

МШ: Это просто потрясающе уменьшение. Чтобы добиться этого, мы принесли в жертву огромное количество пакетов. Если вас сильно тревожит всё то, что не попало на диск, мы, к сожалению, не сможем вам ничем помочь. Но, с другой стороны, за счет этого мы выиграли в других аспектах. Мы находимся в XXI веке, мы должны идти в ногу со временем и осознавать, каким образом работают распределенные организации. И это действительно тяжело: очень трудно основать и руководить совершенно распределенной компанией. Мы должны понимать, можно ли создать компанию, основываясь только лишь на взглядах Open Source. Я не знаю ответа, но я полагаю, что

это действительно интересно и стоит того, чтобы хотя бы попробовать.

LXF: Как вы считаете, нужно ли Linux сражаться за компьютеры пользователей, чтобы это стало достижением?

МШ: Linux абсолютно готов к использованию на некоторых пользовательских компьютерах. Меня несколько не смущают попытки убедить кого-то в том, что Linux готов к этому. На самом деле, я стараюсь сменить тему обсуждения в сторону того, чтобы выяснить, для каких именно компьютеров он предназначен. В зависимости от размеров вашей организации, он будет подходить вам в разной степени. Для меня очень трудно, например, предложить отцу трех детей перейти на Linux, так как у него есть только один компьютер. »



ПОЧЕМУ DEBIAN — НЕ DUBUNTU

«Тот, что я полагал неподражаемым, оказался неуправляемым.»



Грег-Кроа Хартман (Greg Kroah-Hartman)

Ведущий разработчик стабильной ветки ядра Linux
Linux Format № 81,
июль 2006 г.

Linux Format: Вы упомянули, что другие ОС используют драйвера для Linux. Syllable, IBM K42...

Грег-Кроа Хартман: Hurd...

LXF: Да, и Hurd! Вы считаете, что это — в духе идеалов открытого кода, или, по-вашему, в дальней перспективе это плохо?

ГКХ: Нет, меня удивляет, что мы не делимся всем и полностью. Ребята из IBM K42 не хотят писать драйверов — они горят желанием работать над тем, чем они там занимаются в экспериментальном ядре. А драйвера писать никто не любит. Некоторые любят, но большинство — те, которые занимаются исследованиями — не любят, и все-таки тоже хотят работать, не заботясь о драйверах.

LXF: Не потому ли никто не любит писать драйвера, что это сложно? Сложно отлаживать?

ГКХ: Не думаю. Мне это нравится, вот в чем дело. Это не похоже ни на что другое; традиционно драйверы привыкли считать чем-то низкопробным, скверным, что спихивают на пришедших в фирму новичков. Обнадёживает то, что за долгие годы Линус собрал команду неплохих парней, изменивших этот подход, и наши драйвера славятся высокой стабильностью, и все знают, что мы делаем действительно хорошие вещи. Сетевые решения у нас очень, очень хорошие; SCSI тоже очень хорошее; USB вообще отличное — мы поддерживаем большинство новых устройств быстрее любой другой ОС. Поддержку USB 2.0 мы сделали раньше всех. И всякие другие непростые штуки получаем раньше, чем любая другая ОС. Например, Bluetooth.

LXF: Вот на эту штуку мне пришлось установить Windows [тычет в ноутбук]. Это непросто.

ГКХ: Да, мы получаем поддержку оборудования быстрее. Все разработчики устройств используют Linux для выпуска оборудования. IA-64 был создан на Linux, x86-64 был создан на Linux. Парни, занимающиеся оборудованием, любят Linux, они это умеют. У них есть исходный код, и они могут выяснить, что не так с их «железом»... Большую работу выполнили парни из PowerPC: взяли и издали документ, как перевести Linux на гигантские мультипроцессорные PowerPC без firmware и без BIOS. Им не надо было ждать разработчиков BIOS, парни могли сразу приступить к работе с оборудованием.

LXF: Вы сказали на Kernel Summit 2004, что затронули треть ядра. Я выяснял — 1,2 миллиона строк написано и 850 000 удалено, просто невероятно. Похоже на колоссальное переписывание.

ГКХ: Эти числа надо брать с щепоткой соли. Они — механические. Бывает, что добавляются и замещаются в точности те же строки —

в основном, конечно, нет... Добавляются новые драйвера, пересматривается API ядра, происходят улучшения.

LXF: Вы очень тверды насчет бинарных драйверов. Не могли бы вы объяснить, почему бинарные драйвера — скажем, драйвер Nvidia — незаконны?

ГКХ: Сам по себе драйвер Nvidia не является противозаконным. Это очень просто, поговорите с юристом. Я не юрист. GPL дает четкое определение компоновке.

LXF: Смешение кода GPL с кодом не-GPL?

ГКХ: Да, когда они компонуются — потому что это необходимо сделать при загрузке модуля — когда вы связываете код с ядром, вы получаете единый образ, который подпадает под GPL. И это не нейтральная территория.

LXF: И как же драйвера это обходят?

ГКХ: Ну, если никто тебя не видит, можно и что-нибудь незаконное проверить.

LXF: Да, но вы сказали, что драйвер Nvidia не незаконный.

ГКХ: Потому что в нем ничего незаконного. Вы сами, как пользователь, все компилируете и компонуete. И не можете передать этот скомпилированный объект еще кому-то, не нарушив GPL.

LXF: Это облегчило вам жизнь?

ГКХ: Нет, нисколько не облегчило. Бинарные драйвера превращают нашу жизнь в ад. Пользователи сообщают нам о своих проблемах с ядром, и если у них там стоит бинарный драйвер, мы не можем узнать, в чем дело — возможно, он записал что-то где-то поверх ядра и из-за этого — бах! [сло-малось], или что-то плохое случилось, а нам этого не понять. Ну и если вы сломали ядро, мы сообщаем: у вас бинарный драйвер, мы не можем оказать вам поддержку... решайте вашу проблему сами. Люди знают: из сообщения об отказе ядра должно быть ясно, что они не работают с бинарными драйверами, иначе никакой поддержки им не будет.

LXF: [О тяжбе с SCO] И даже пары недель не ушло на сомнения, не надо ли быстренько пересмотреть исходный код?

ГКХ: Нет. Linux — лучше всех документированная крупная кодовая база. Там всегда точно известно, откуда что взялось. Всегда можно проследить историю всех этих публичных изменений, до самого истока. Не бывает такого, чтобы мы не знали, откуда что взялось, мы всегда это знаем, это же происходит открыто. Можно по-другому повернуть вопрос: вот эти закрытые ОС, откуда они берут свой код? Откуда нам знать, что они не взяли наш? Я не говорю, что они так и сделали, но вы ж понимаете. А за наши программы я не боюсь. **LXF**

КОЕ-ЧТО НИКОГДА НЕ ИЗМЕНИТСЯ

«Нет, это не облегчило мне жизнь. Бинарные драйвера превращают нашу жизнь в ад.»



ГНУ/Линуксцентр: Долгая дорога свободы



Издатель *Linux Format* Павел Фролов вспоминает о пройденном пути.

ГНУ/Линуксцентр появился в сентябре 2000 г. как интернет-магазин, где продавались диски с СПО и книги, посвященные администрированию Linux и других свободных продуктов. Очень быстро стало понятно, что, несмотря на популярность этой темы, в России оказалось не так много производителей СПО. Поэтому мой партнер Стас Медведев предложил перейти от дистрибуции и реселлинга чужих дистрибутивов к производству и издательству дистрибутивов GNU/Linux и другого свободного ПО собственного изготовления под маркой ГНУ/Линуксцентра.

Мы приобрели оборудование для тиражирования компакт-дисков и начали самостоятельное производство дисков, в первую очередь, с ОС GNU/Linux, и занимались этим несколько лет. Затем, осознав, что дальнейшему продвижению СПО в России мешает отсутствие обучающей и справочной литературы на полках книжных магазинов, мы начали сотрудничество с издательством BHV, в рамках которого совместно выпустили около десятка книг, посвященных ОС GNU/Linux и СПО, и среди них оказалась пара бестселлеров.

Примерно через пять лет деятельности нам стало очевидно, что у пользователей есть потребность в создании журнала для СПО-сообщества, и мы запустили журнал *Linux Format*. У нас не было свободных средств на выпуск журнала, поэтому мы обратились к пользователям и предложили им авансом приобрести подписку на журнал, который в действительности еще не существовал: это был один из первых успешных примеров краудфандинга в России. Я очень благодарен людям, которые тогда поверили в нас и подписались на еще не изданный *Linux Format*.

Затем мы начали готовить *Linux Format* к выпуску, собравшись с нашими коллегами, Денисом Игнатовым и Родионом Водейко. Мы заперлись на квартире у Родиона и около недели не выходили на свет Божий. Это, конечно, осложнило жизнь всей команды: лично я за это время похудел на несколько килограммов — даже поесть было некогда. Так мы выпустили первый номер *Linux*

Format в сентябре 2005 г., и с тех пор журнал выходит ежемесячно вот уже 10 лет.

В 2007 г., через 2 года после выпуска первого номера *Linux Format*, ко мне пришел Валентин Сеницын, на тот момент главный редактор журнала, и рассказал об очень перспективном, на его взгляд, проекте свободного аппаратного обеспечения — Arduino. Восторженные отклики «Хочу такую штуку!» от наших читателей после первых публикаций об Arduino в *Linux Format* привели к тому, что мы стали первым дистрибьютором итальянского производителя этих плат — компании Smart Projects, привезли первые партии Arduino в Россию и обратили пристальное внимание на продвижение данного продукта. Это дало свои плоды, и сейчас Arduino очень популярен в России — мне известно как минимум с десятком дистрибьюторов и пять предприятий, которые производят клоны Arduino в России.

Следующим этапом на нашем длинном пути стало налаживание отношений с государством. В 2007 г. на всю страну прогремело дело Поносова, когда директор школы в Перми чуть было не сел

платформы. Также мы разработали сайт Министерства юстиции РФ на базе свободного продукта *Drupal*. Причем был разработан сайт не только федерального министерства, но и 83 сайта территориальных органов. Это был очень большой и сложный проект, и я очень горжусь, что нам выпала честь его сделать.

Сейчас мир очень серьезно изменился по сравнению с тем, что было 10–15 лет назад. То, с чего начинал ГНУ/Линуксцентр, уже неактуально: мало кому нужны дистрибутивы на дисках, немногие читают печатные книги. Всё стремительно переходит в цифровую форму, поэтому журнал *Linux Format* активно развивается в электронном виде, а ГНУ/Линуксцентр переключается с продаж дистрибутивов на дисках на продажи дистрибутивов на флешках, различного оборудования под свободными лицензиями.

Государство в условиях сложной внешнеполитической обстановки стало серьезнее относиться к вопросам информационной безопасности и независимости, что автоматически выводит на первый план свободное программное и аппаратное обеспечение. В частности, 18 сентября этого года в Госдуме прошло заседание комиссии по стратегическим информационным системам, где очень подробно изучались вопросы применения СПО для построения государственных информационных систем, и закончилось это обсуждение внесением инициативы по принятию закона об обязательном внедрении СПО в государственные структуры РФ.

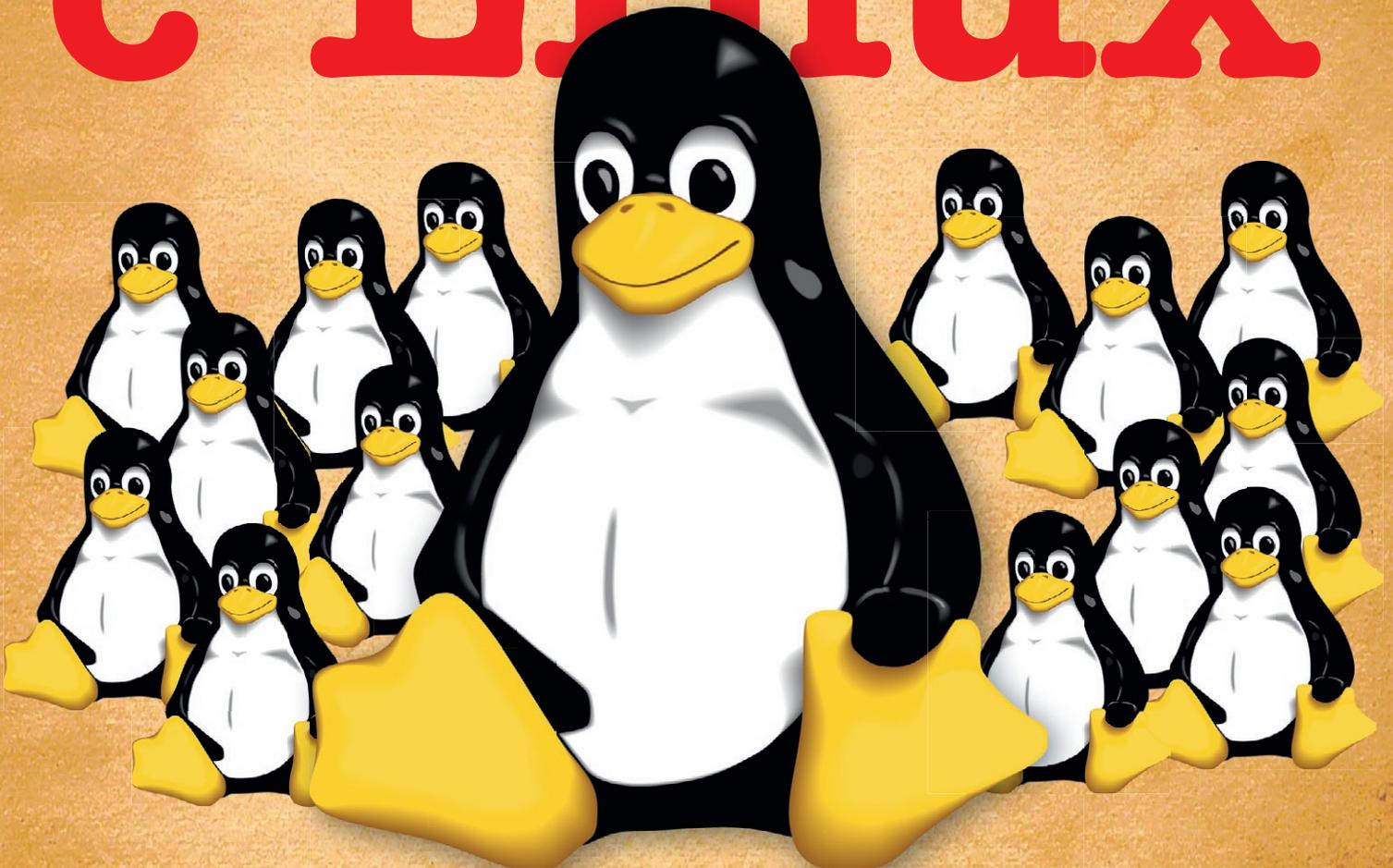
Я присутствовал и выступал там. Был приятно удивлен тем, насколько хорошо представители Государственной Думы информированы о специфике СПО и его преимуществах для использования в госорганах. Очень хочется верить в то, что государство сделает подарок на юбилей ГНУ/Линуксцентра и журнала *Linux Format* и действительно примет закон об обязательном использовании СПО в органах государственной власти — по примеру такого же закона, который уже много лет как принят в Венесуэле и недавно — в Великобритании. Наша долгая дорога должна привести к правильной цели. **LXF**

«Наша долгая дорога должна привести к правильной цели.»

в тюрьму из-за обнаруженного в школе пиратского Windows. Мы бойко включились в массовую кампанию по ликбезу, чтобы объяснить людям, как такие случаи предотвращать. Назрело решение предложить перевести государственные учреждения на СПО, начиная от школ и заканчивая организациями государственной власти. Результатами нашей деятельности в этой области стали целый набор аналитических отчетов и даже нормативных документов, подготовленных при нашем непосредственном участии, которые мы заботливо собрали на сайте www.linuxformat.ru/foss-russia.

В 2012 и в 2013 гг. коллективом ГНУ/Линуксцентра и редакцией *Linux Format* была создана концепция НПП — национальной программной

15 лет с Linux



Ударимся в воспоминания вместе с Джонни Бидвеллом, который выясняет, как же изменился Linux за время существования журнала.

Шло холодное хмурое утро мая 2000 г. Зима уже должна была закончиться, но в Британии этого не было заметно. И тогда Читатель Икс в поисках укрытия от ледяного дождя и от мрачной толпы заглянул в местный газетный киоск. Икс надеялся найти какое-нибудь мотивирующее и поучительное чтение, однако у него создавалось впечатление, что полки забиты низкопробными романами и игровыми и прочими журналами, которые совершенно не интересовали Икса. И тогда он увидел его, ярко-оранжевый луч света

во мраке: «Присоединяйся к революции!» — звала нашего восхищенного читателя надпись на обложке. Пораженный тем, что ожесточенная возня с терминалом, которую почему-то считали не хоб-

диск с целым дистрибутивом Linux (Definite Linux)! Чтобы скачать его через телефонное соединение, ушел бы месяц. А через четыре недели должен был появиться еще один, и не CD, а DVD.

Жизнь Икса радикально изменилась, и хотя Definite Linux определенно оказался не самым долговечным, а за последующие 15 лет журналу неоднократно пришлось переходить из одних рук в другие, меняя хозяев, он остался бастионом качествен-

«Свет во мраке: „Присоединяйся к революции!“ — звала читателя обложка.»

ных публикаций [Ред.: — Пока до него не дорвался Джонни], которые информируют, развлекают и доставляют истинное удовольствие.

ных публикаций [Ред.: — Пока до него не дорвался Джонни], которые информируют, развлекают и доставляют истинное удовольствие.

На тот момент, когда Икс испытал про-светление в газетном киоске, Linux уже было девять лет. Некоторые дистрибутивы уже вполне устоялись, и одним из самых первых был Softlanding Linux System (SLS), он появился в мае 1992 г. В отличие от своих современников, SLS предлагал не только ядро и несколько инструментов GNU для подготовки файловой системы: вместо этого он поставлялся с сетевым пакетом и сервером отображения X. Его сочли весьма амбициозным, но полным ошибок, и попытки отладить его достигли кульминации в релизе Slackware в 1993 г. В том же году, опять же в качестве реакции на приносимые SLS огорчения, появился Debian. В следующем году появился Red Hat Commercial Linux, который породил множество популярных в 1990-е дистрибутивов, включая Mandrake, Yellow Dog и Definite Linux. KDE вышел в 1998 г., а в 1999 г. появился Gnome. Gnome был создан отчасти из-за того, что KDE использовал набор инструментов Qt, который на тот момент не подлежал свободному лицензированию. К маю 2000 г. самыми популярными дистрибутивами были Debian 2.1, Red Hat 6.1, Linux-Mandrake 7.0 (так он тогда именовался), Slackware 7.0 и SUSE Linux 6.3. О некоторых из них рассказывалось в самом первом Сравнении LXF, и вы можете прочитать о них в эксклюзивном оцифрованном выпуске LXF1 на LXF DVD этого месяца.

Что такое опыт пользователя?

Если вы недавно обратились к Linux, и вам пришлось присоединиться к борьбе с гадкими файлами настройки, не желающими работать драйверами или прочими неприятностями, подумайте о первых пользователях, чьи сообщения об ошибках и нелюбимые высказывания проложили дорогу к современному настольному Linux. До относительно недавнего времени можно было испортить отображение, предоставив X неверную информацию по времени. У вас были проблемы с Grub? Попробуйте-ка справиться с ними с помощью первых версий Lilo.

В те первые дни даже всего лишь заставить работать мышью было отнюдь не просто: от пользователя требовалась всевозможная ручная калибровка. Red Hat выпустил инструмент под названием *Xconfigurator*, который обеспечивал



» Тема Human была попыткой сделать Ubuntu Linux дружелюбнее: ведь каждому известно, что коричневый — это красиво, особенно если ты бородач.

текстовый интерфейс к меню для настройки X-сервера. И он считался просто даром Божиим, несмотря на то, что всего-навсего генерировал файл *XF86Config*, который в противном случае вам приходилось писать самостоятельно. И вот, пока пользователи Windows плакали, что Windows ME тормозит и отключает реальный режим DOS, среднестатистический пользователь Linux плясал от радости, если процесс установки доходил до завершения. И даже если вы доходили до этого момента, надеяться на то, что ОС нормально загрузится, было необоснованным оптимизмом. Определения оборудования вообще не существовало, и из нескольких

было событием исключительной важности для пользователей настольных ПК, благодаря унифицированной работе с PCI, ISA, PC Card и устройствами PnP, а также поддержке ACPI. Пузырь дот-комов должен был вот-вот взорваться, однако шумиха и ажиотаж вокруг него означали, что множество компьютерных энтузиастов получили широкополосное соединение дома, а некоторые даже позволили себе роскошь обладания более чем одним компьютером. Это снимало часть основных барьеров на пути к Linux: теперь скачать его становилось намного проще; доступ к обновленной документации был прост; и когда в Linux пропадало соединение с Интернетом (или система не загружалась), можно было использовать другую машину для поиска решения проблемы. Однако в целом взаимодействие с пользователем по-прежнему не отличалось дружелюбием. Хотя некоторые программы установки начали использовать графические возможности, по большей части они создавали больше проблем, чем стоило. От пользователя ожидалось знание всех деталей деления диска на разделы и способности выяснить, какие нужны пакеты, по часто весьма сжато описанию.

Примерно в октябре 2001 г. появилась Windows XP, и хотя ее считали значительным шагом вперед по сравнению с ее предшественником, многие

«Даже заставить работать мышью было непросто: требовалась ручная калибровка.»

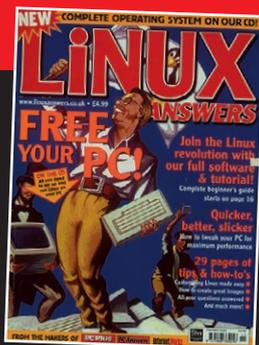
драйверов, написанных под Linux, большинство не отличалось качеством. Однако первопроходцы не сдавались — многие из них предпочитали стиль работы DOS, который к началу второго тысячелетия оказался на задворках. Пользователи Windows начали выделять свои файлы — выразителем этого движения стал «My Computer».

В январе 2001 вышло ядро Kernel 2.4, и с ним, среди всего прочего, появилась поддержка USB и потрясающих новых процессоров Pentium IV. Это

Шкала времени

Предыстория — Linux Answers

В конце 1999 г. Future plc выпустила один номер журнала, который появился на волне успех ныне закрытых PC Answers и PC Plus [Ред.: — О воспоминания!]. Он есть на LXF DVD. Все, что мы можем сказать — это что он был достаточно успешен, чтобы привести к появлению ежемесячного журнала...



Май 2000 — Linux Format №1

Название меняется в соответствии с самыми успешными журналами от Future: *Amiga Format* и *PC Format*. *Linux Format* вышел под редакцией Ника Вейча [Nick Veitch], который обрел славу в *Amiga Format* и отточил писательский талант в *PC Plus*. Журнал вышел с CD и мгновенно стал хитом.



пользователи обнаружили, что их машины не дотягивают до ее требований. Ведь она требовала 64 МБ ОЗУ и целых 1,5 Гб места на диске. Вспомните, что лишь недавно BIOS обрели способность работать с большими дисками (существовали разные ограничения в зависимости от BIOS: обычно пределом были 2,1, 4,2 и 8,4 Гб). Очень многим было не установить ее на свое оборудование, а многие из тех, кто сумел выполнить минимальные требования, обнаружили, что после установки обычного пантеона офисных программ и библиотек runtime производительность очень быстро падает. Это стало толчком для следующего небольшого исхода в Linux, и контингент пользователей со старым оборудованием продолжает составлять важную часть пользовательской базы Linux (которая ругает нас, если мы не включаем 32-битные дистрибутивы). До 2006 г. все Mac'и имели процессоры PowerPC, и многие из них (так же, как и первые Intel Macs), давно лишившиеся обновлений от Apple, теперь тоже работают на Linux.

В 2002 г. появился рабочий стол Gnome 2 — и стал настолько значимым, что многие до сих пор (из-за ностальгии, в порядке атавизма или принципиального неприятия современных альтернатив) изо всех сил стараются его воссоздать. Он был намерен стать простым, легким в настройке

и интуитивным, и нельзя не признать, что всех поставленных целей он достиг.

О, мы прелестны

Одним из главных решающих факторов была приверженность принципам Gnome Human Interface Guidelines, которые определили ряд основных принципов для дизайнеров приложений. Это означало, что рабочий стол был устойчивым не только внутренне, но и относительно всех приложений GTK, которые могли для него написать.

«Gnome 2: настолько значимый, что многие до сих пор стараются его воссоздать.»

Вышел также и KDE 3, который несколько напоминал Windows — он был более похож косметически и чуть более требователен по отношению к ресурсам, чем Gnome. Пользователям и дистрибутивам пришлось выбирать между ними. SUSE Linux (предшественник openSUSE) всегда стремился быть независимым от рабочего стола, однако большинство его пользователей предпочли KDE. Приняв это во внимание, хотя это и произошло только после 2009 г., он изменил свою направленность, и сейчас это ведущий дистрибутив на базе KDE.

В конце 2002 г. Йона 'DVD' Йохансена [Jon Johansen] обвинили за выпуск в 1999 г. программы DeCSS в обход Content Scrambling System (CSS), используемой в коммерческих DVD. Эта программа позволила пользователям Linux воспроизводить DVD — радость, которой они до этого были лишены, ибо программы DVD требовали лицензионного ключа от DVD Copy Control Agency, одного из источников в разбирательстве. Позднее выяснилось, что CSS можно взломать куда проще, и Йохансен был оправдан. К тому времени iPod'ы и пиратство

начали повсеместное распространение файлов MP3. Им постоянно сопутствовали — и до сих пор сопутствуют — проблемы с патентами, связанные с тем, что ряд претендентов предъявляет права собственности на разные части лежащего в основе алгоритма. В результате многие дистрибутивы шли без обремененных патентами мультимедиа-кодеков. Однако закон довольно расплывчат, и правообладатели не спешили предъявлять иски за реализации этих кодеков в FOSS. Большинство дистрибутивов отличаются осторожностью и оставляют их установку на усмотрение пользователя, хотя Ubuntu предлагает пользователям лицензионные (но проприетарные) кодеки Fluendo при установке. К счастью, срок многих патентов MP3 истек, и у многих он истечет к 2017 г., но это на самом деле не так уж важно, поскольку теперь у нас есть множество открытых форматов и кодеков (OGG, FLAC, VPx и x264). Технически использование *libdvdcss* (это современный и куда более эффективный способ взлома CSS, применяемый большинством медиа-плееров в Linux) для просмотра DVD по-прежнему является нарушением DMCA, однако это практикуется только в некоторых [Ред.: —Экономически отсталых] странах, и на сегодняшний день никто не опротестовывает его использование.

В 2003 г. Мюнхен объявил о переводе всей своей инфраструктуры с Windows NT на Linux. Помимо экономических причин, жители Баварии объяснили, что основным импульсом этого перехода стала возможность освободиться от запретов производителей. Стив Балмер [Steve Ballmer] лично посещал мэра, но даже его шарм и красноречие (а также, вероятно, и обещание приличной скидки) не переубедили революционеров. Проект был завершен десять лет спустя, когда порядка 15000 машин перешли на индивидуальный дистрибутив

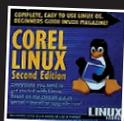


➤ В проекте LiMux красуется Tux с эмблемой Мюнхена, Мюнхенским Дитятей [Münchner Kindl]. Переход оказался совершенно безболезненным и, по оценкам, позволил сэкономить около €11 млн.

Шкала времени

Ноябрь 2000 — LXF007

На CD попал Corel Linux — дистрибутив, основанный на Debian. Может, ОС и не была успешной, но это было исключительно легкое введение в Linux, и оно указало путь разработчикам дистрибутивов.



Апрель 2001 — LXF013

Начало нового века потребовало новых носителей, поэтому для *Linux Format* наконец-то началась эра DVD! На первом LXF DVD вас дожидались SuSE Linux 7.1 и Red Hat 7.0.



Май 2001 — LXF014

Сначала DVD, а потом и совершенно новый вид LXF! Первый редизайн журнала закрепил место для таких фаворитов, как HotPicks, Сравнения и Ответы. В новостях Linux была Международная космическая станция, и ожидался AMD64.



Видеодрайверы и недовольство ими

К 2003 г. ATI (теперь это часть AMD) и Nvidia выпустили проприетарные драйверы для укрепления возможностей 3D на своем новом оборудовании (в 2005 г. флагманскими картами были X1800 и серия 6800 от Nvidia). Были и драйверы с открытым кодом, однако их производительность была невысока.

ATI были более настроены на выпуск спецификаций устройств, чем их оппоненты, и в результате драйверы с открытым кодом разрабатывались быстрее. Nvidia через свой драйвер nv выпустила только весьма туманный исходный код, который лишь расстроил и озадачил разработчиков. Двоичные драйверы, как оказалось, были весьма проблемными, даже при наличии таких удобных инструментов

управления, как Ubuntu Jockey. Репозитории сильно отставали от последних релизов, подстегивая пользователей скачивать пакеты прямо с AMD или Nvidia. Они отличались крайне плохой работой (мы их до сих пор не любим) и наносили серьезный ущерб имеющейся системе драйверов. Поскольку они существовали вне области действия менеджера пакетов, при любом обновлении ядра модуль драйвера приходилось перекомпилировать. В ином случае при следующей перезагрузке графика отсутствовала вообще, что, естественно, сильно огорчало некоторых пользователей.

Эта ситуация улучшилась благодаря DKMS, однако проблемы с графикой продолжают доводить некоторых пользователей до скрежета зубного.

Во многих случаях история все та же: драйверы с открытым кодом слишком медленные, а двоичные могут нанести ущерб.

В ответ на слабую производительность и отсутствие поддержки 3D в драйвере nv в 2006 г. было сообщено о проекте pouveau. Это была колоссальная попытка чистого обратного инжиниринга, который частично использовал данные краудсорсинга: участники должны были скачать программу REpouveau, которая вела реестр, создавала графики и затем делала мгновенный снимок области реестра для анализа разработчиком. Только в 2012 г. вышел стабильный релиз pouveau, хотя в некоторые дистрибутивы он попал на три года раньше, поскольку даже на стадии наличия ошибок оказался лучше nv.

Linux. В 2014 г. пронесся жуткий слух, что город должен снова вернуться на Windows, но он оказался ложным. По оценкам, этот переход позволил Мюнхену сэкономить около €11 млн.

О ядро! Мое ядро!

[аллюзия автора на стихотворение Уолта Уитмена «О капитан! Мой Капитан!» — прим. пер.] После двух лет разработки в 2003 г. вышло Kernel 2.6. Оно в корне отличалось от 2.4, предлагая улучшения в планировщике, улучшенную поддержку систем мультипроцессора (включая гиперпоточность, поддержку NPTL и NUMA), более быстрый ввод/вывод и огромный объем дополнительной поддержки оборудования. Мы также увидели Physical Address Extension (PAE), чтобы машины могли обращаться к 64-ГБ ОЗУ, даже на 32-битной архитектуре. Появилась также достойная подсистема Advanced Linux Sound Architecture (ALSA), которая давала (почти) готовую функциональность популярных звуковых карт и поддержку нескольких устройств, аппаратное микширование, дуплексный режим и MIDI. Самой далеко идущей новой функцией была подсистема управления старым оборудованием, devfs, на смену которой пришла udev. Ее не существовало до 2.6.13 (ноябрь 2003 г.), когда директория /dev перестала быть списком (множеством) статических узлов и сделалась динамическим отражением устройств, реально соединенных с системой. Подсистема udev также работает с загрузкой прошивки и событиями пространства пользователя, и стала огромным вкладом в удобство работы для



» EeePC Linux от Asus был основан на Xandros и IceWM, но начинающим пользователям он не понравился, а профессионалы просто его заменили.

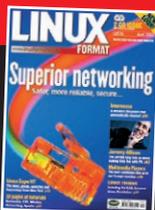
пользователей настольных ПК. Хотя по-прежнему приходилось использовать мистические силы вроде HAL и ivman, чтобы автоматически смонтировать устройство USB с должными разрешениями.

Linux (уже портированный на 64-битные процессоры pentium 4) поддерживал набор команд Itanium IA64, когда она вышла в 2001 г. Однако этой архитектуре не суждено было добиться успеха, и в итоге Intel перешел на более консервативную архитектуру AMD64 (или x86-64), которая (и нам при-

ятно напомнить об этом нашим дорогим читателям) существует с 2003 г. Благодаря ПО с открытым кодом, пользователи Linux смогли сразу же использовать 64-битные ПК, а вот пользователям Windows пришлось ждать до 2005 г., когда появился x64 релиз XP. Разнообразные проприетарные приложения (наиболее примечательными являются Steam и его игры) работают в 32-битном режиме, который обеспечивает некоторую мотивацию для дистрибутивов поддерживать 32-битные релизы, од-»

Апрель 2002 — LXF026

Второй новый дизайн для журнала за столько лет! В этом выпуске также было очень популярное интервью с инженером Samba, Джереми Эллисоном [Jeremy Allison].



Май 2002 — LXF027

В этом выпуске были опубликованы долгожданные результаты голосования читателей Linux Format Awards 2001. Победили Mozilla и Apache, а лучшим дистрибутивом был избран Mandrake.



Февраль 2003 — LXF037

Вероятно, впервые мы задали вопрос: станет ли этот год годом Linux на рабочем столе? Цитата: «Я ожидаю, что 2003 год станет годом прорыва». Мы рассказали о Windows 3.0, Unreal 2003, и нам по-прежнему нравились IceWM, KDE и WMaker.



нако наступит день, когда поддерживать их будет уже нерационально, и в конечном итоге их ожидает судьба 386, которая больше не поддерживается в Linux с 2013 г.

Введите архетип

Выход в 2004 г. Ubuntu 4.10 (Warty Warthog) стал, без сомнения, главным событием для настольного Linux. Истратив кучу денег, полученную от создания и продажи Thawte, Марк Шаттлворт [Mark Shuttleworth] создал Canonical Inc. Целью была продажа серверных продуктов и поддержки, и в то же время — создание настольного Linux «для людей». Используя Debian (который к тому времени доказал свою состоятельность) в качестве базы, в Canonical добавили настройки драйверов, очень коричневую тему Gnome 2 и крайне амбициозный цикл релизов каждые полгода. Мы также стали свидетелями появления <http://ubuntuforums.org>, где полные благих намерений, но располагающие весьма скудной информацией члены сообщества могли размещать свои «решения» разных проблем Ubuntu.

В 2004 г. появился аудиосервер под названием *Polyaudio*, созданный доселе неизвестным разработчиком по имени Леннарт Пёттеринг [Lennart Poettering] со товарищи. На тот момент рабочий стол использовал аудиосервер для устранения не-

Революция Raspberry Pi

В 2012 г. вышел Raspberry Pi. В чем-то вдохновленный успехом BBC Micro (отсюда и монограмма названий моделей) в начале 1980-х, Raspberry Pi поставил задачу принести практические информационные технологии в школьные классы и загрузить электронную промышленность Великобритании. Этот дешевый компьютер размером с кредитную карту уже продан в количестве более 5 миллионов единиц.

И хотя многие из них сейчас дают поле деятельности юным кодерам, солидная часть стала частью разнообразных проектов людей весьма зрелого возраста: тех, кому уже за 30, и кто вырос на BBC, Spectrum, C64, и кто заново переживает восторги кодирования и творчества. GPIO-выходы Pi позволяют разработать самые разные дополнения,

достатков в системе микширования ALSA: Gnome использовал Enlightened Sound Daemon (ESD), а KDE — аналоговый синтезатор Realtime (aRts). *Polyaudio* был создан в качестве замены ESD, с намного более передовыми функциями, такими как регулировка громкости для каждого приложения и сквозной тракт передачи. В 2006 г., ссыла-

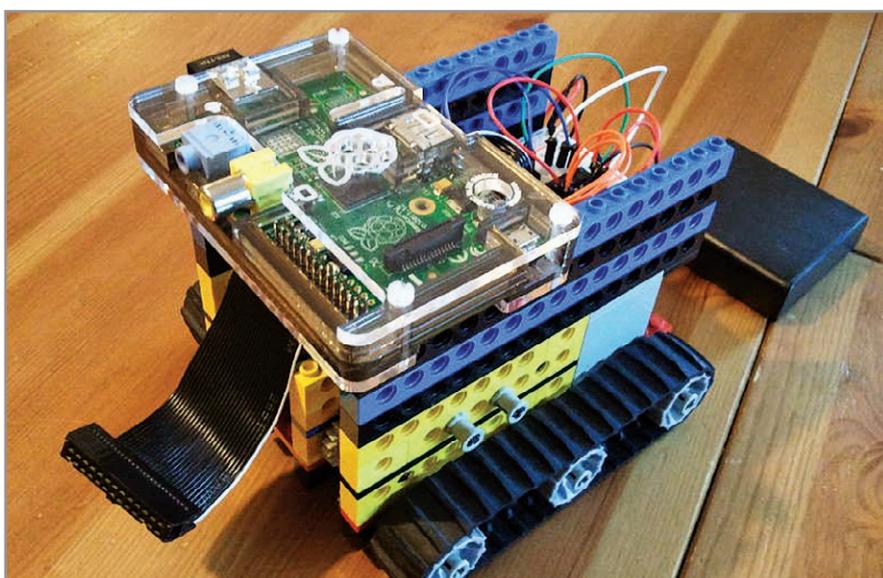
дея Pi основой чего угодно, от роботов до систем полива с дистанционным управлением.

Общепринятым языком проектов Pi является Python, который, как и Basic, прост в изучении. Однако, в отличие от Basic, он стабилен и расширяем, и его не надо изгонять из сознания при переходе на более продвинутые языки. Поддержка 3D-графики в Pi впечатляет, однако в плане CPU он более ограничен. Исходные Pi прилагали героические усилия, пытались работать настольными ПК, пусть и со скромным дистрибутивом Raspbian (хотя недавняя работа над браузером *Epiphany* улучшила ситуацию). В 2015 г. Pi получил перезагрузку, обзаведясь четырехъядерным процессором и добавив ОЗУ, так что теперь это воистину компьютер общего назначения, ценой по-прежнему всего £25.

на критические замечания о том, что вряд ли кому-то нужны полипы, проект переименовали в *PulseAudio* (на самом деле он получил свое название в честь морского обитателя, а не медицинской проблемы).

Благодаря новому названию и возросшему спросу на аудиосистему, сравнимому со спросом на OSX или недавно вышедшую (и вызвавшую волну негатива) Windows Vista, *PulseAudio* подвергся солидной разработке и стал котироваться как кандидат на включение во множество дистрибутивов. Традиционно первым оказался Fedora, включивший его по умолчанию в версию 8, вышедшую в конце 2007 г. За ней последовал Ubuntu в версии 8.04, хотя его реализация вызвала немало критики, обусловившей злобное отношение к *Pulse*. В какой-то момент Пёттеринг даже назвал свое детище «программой, которая ломает вам аудио». Но в конечном итоге Ubuntu (и другие дистрибутивы) решили проблемы с реализацией, на что ушло немало времени, и сейчас он в основном по умолчанию работает.

До появления доступных планшетов и смартфонов вершиной портативных технологий были нетбуки. И первым был Asus EeePC 701. Из-за низких спецификаций оборудования (у него был процессор 700 МГц, дисплей 800×480 и 512 МБ ОЗУ) установка на него Windows даже не рассматривалась. Вместо этого он шел с индивидуальной версией Xandros Linux, который был функциональным, но ему не хватало лоска. В целом большинству



➤ Raspberry Pi вдохновил целое поколение создателей. В случае восстания роботов мы сможем обвинить в этом Foundation (и загнать Леса Паундера в штрафбат).

Шкала времени

Декабрь 2004 — LXF060
Первый обзор Ubuntu 4.10 парнем по имени Джоно Бэзон [Jono Bacon], по нашему мнению, весьма скандальный; как ни странно, Ubuntu ему понравился. Нет, на диске его не было, зато был Mandrake 10.1!



Январь 2005 — LXF061
LXF запускает сравнение Лучший Дистрибутив [Best Distro], и Mandrake с легкостью побеждает, а бедняга Ubuntu занимает 9-е место. Некий Грэм Моррисон [Graham Morrison] становится штатным автором, и мы впервые включаем Ubuntu в LXF DVD вместе с Fedora Core 3.

Октябрь 2006 — LXF084
Происходит последний редизайн LXF, и по большей части журнал выглядит так же, как и сегодня — с выборочной сменой разделов — и LXF DVD также переместился внутрь журнала.



Июнь 2008 — LXF106
Вундеркинд Пол Хадсон [Paul Hudson] стал редактором/Главным командующим, и мы уделали все внимание Asus Eee PC; у Джонни по-прежнему свой, на котором работает Arch Linux...

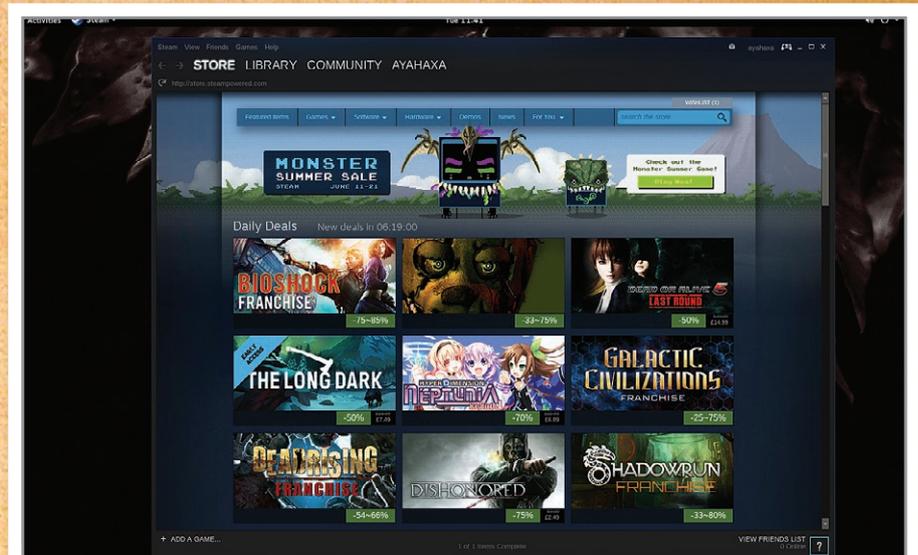


он не нравился, но нетбуки продолжали оставаться отличной платформой для опытных пользователей Linux. По мере появления новых нетбуков (многие использовали более подходящие чипы Intel Atom), они стали предлагать Windows XP (лет через семь после его выхода), а потом кривую Windows 7 Starter Edition. Позднее Asus перестал проявлять такой энтузиазм по отношению к Linux: объединившись с Microsoft, они даже запустили кампанию «С Windows лучше», целью которой было отвлечь людей от приобретения ноутбуков на Linux. Эта грязная кампания использовала фразы типа «основные проблемы с совместимостью» и «незнакомая среда», чтобы отпугнуть людей.

Цена прогресса

2010 г. некоторые помнят как год утраты связи Ubuntu с реальностью. До того момента дистрибутив всё хорошел, обретая всё больше пользователей и все большую стабильность. Он был олицетворением мечты (погибшей или несбыточной, в зависимости от того, у кого вы спросите) о настольном Linux. Однако с релиза 10.10 все пошло наперекосяк. Отныне *Ubuntu Software Center* включал платные приложения (первым стал лицензионный DVD плеер от Fluendo), а ремикс Netbook использовал новую среду рабочего стола под названием Unity. А в релизе 11.04 она стала основной оболочкой для основного релиза. Ubuntu долго не соглашался с новым рабочим столом Gnome 3, который на момент заморозки функций Ubuntu не считался достаточно стабильным для включения в релиз, а Gnome 2 уже стал реликтом. Так что в некотором смысле у Ubuntu не было выбора: но перемен никто не любит, и пользователи быстро начали жаловаться на новые рабочие столы. Ubuntu не отказался от Unity, и на сегодняшний день он значительно улучшен, но в релизе 12.10 произошел спад, когда пользователи обратили внимание на «предложения» от Amazon, вводя запросы в линзы поиска.

В Gnome 3 тоже имеются противоречия — его критикуют аж по трем направлениям: во-первых, многие предпочитают старый способ работы Gnome 2, а он совершенно явно изменен. Во-вторых, крутые эффекты рабочего стола требуют наличия мощной видеокарты (и рабочих драйверов). Там был режим отката к прежнему, но он сильно вредил удобству работы. И, наконец, оказалось, что он разработан для использования на мобильных те-



► Благодаря *Steam on Linux*, геймеры Tux наконец-то получили в свое распоряжение тысячи игр, и авторы LXF теперь могут внимательно изучать предложения Летней распродажи, притворяясь при этом, что работают.

лефонах или планшетах, однако мобильный Linux (не считая Android) до сих пор так и не запустился, так зачем навязывать пользователям подобный образ мысли? Многие все же полагают, что если забыть старые привычки, выучить некие хитрые сочетания клавиш быстрого запуска (и установить *Gnome Tweak Tool*), то работа в Gnome 3 может оказаться не менее, если не более, эффективной, чем

части приходилось через *Wine*, которая существует с середины 1990-х. Все изменилось, когда в 2013 г. Valve выпустили свой клиент *Steam* для Linux. Ныне есть более 1000 игр, доступных для Linux, и они все время портируются, так что их становится больше. Естественно, многие высококачественные порты используют либо уровень *Wine*, либо надстройку типа *eOp*, но все равно наблюдается солидная пропорция инди-релизов, работающих по умолчанию. Valve даже создали OpenGL-версию игры про зомби *Left 4 Dead 2*, которая превзошла релиз DirectX/Windows. Пользователи Linux на текущий момент составляют всего около 1% пользовательской базы

«Последние несколько лет жизни Linux наполнены достижениями и драмами.»

работы с его предшественником. Пользователи KDE взирали на всё это несколько свысока, поскольку уже пережили ужасы модернизации рабочего стола (хотя и не столь радикальные, как у Gnome), когда в 2008 г. вышел KDE 4. Здесь мы также должны упомянуть *Systemd*, однако вряд ли мы много добавим ко всему, что уже сказано: старая система *init* трещала по швам, появилась новая. Она устраивала не всех, но мы все равно ее используем; на Леннарта Пёттеринга Интернет клеветает.

В Linux всегда существовал нишевый интерес к играм, однако играть линуксоидам по большей

Steam, но это может измениться, если воплотится план Valve по завоеванию гостиных [Ред.: — Почему не рабочих столов?!] через компьютеры Steam, на которых работает Steam OS на базе Debian.

Последние несколько лет жизни Linux были наполнены достижениями и драмами, включая баг Heartbleed, частичное разрешение долгосрочного судебного разбирательства SCO-IBM и смену отношения на куда менее негативное со стороны извечного соперника — Microsoft. Но нам здесь, к сожалению, просто не хватит места на все эти увлекательные рассказы. LXF

Июль 2009 — LXF120

Мы отметили Ubuntu 10.04, поместив его на обложку и вновь взяв интервью у Марка Шаттлворта. От этого типа с его Salopical нам никак не удается отделаться!



Август 2010 — LXF134

Что это за Android такой, и каким образом он рассчитывает конкурировать с iPhone? Мы объяснили, каким именно, плюс поместили Mint 9 и Fedora 13 на наш LXF DVD.



Апрель 2012 — LXF156

Мы сообщили об этой штуке под названием Raspberry Pi еще в LXF147, но наконец-то мир смог приобрести этот крошечный ПК, и мир его полюбил. А как же было иначе?



Январь 2014 — LXF179...

В редакции *Linux Format* обновляется новая редколлегия, а старая команда уходит на другую работу. LXF179 — самый продаваемый выпуск этого года, а LXF181 становится бестселлером в Великобритании почти на два года! Спасибо, что помогаете LXF оставаться бестселлером!



LINUX FORMAT

Первый редактор русской версии журнала Валентин Сеницын — о том, как это было.

Э тот неловкий момент, когда твой текст снова появляется на страницах LXF после пятилетнего перерыва... Кажется, все эти нескончаемые расшифровки интервью и репортажи, ответы на вопросы и оглавления были только вчера.

Примерно с такими ощущениями я воспринял предложение редакции LXF написать материал по случаю десятилетия русской версии журнала. «Как 10 лет? Пять отмечали буквально в прошлом октябре!» Однако беспристрастный календарь «как бы намекал», что очередной юбилей уже на пороге.

Впрочем, кое-кто может возразить, что юбилейная шкала для ИТ-журнала должна быть другой: 2, 4, 8, 16... В этом есть рациональное зерно, и поскольку LXF всегда прислушивался к мнению своих читателей, даже если оно не совпадало с мнением редакции, ответим словами Владимира Семеновича: «Ну ладно, хорошо, не юбилей, а просто две нормальных пятилетки».

Так с чем мы подошли к не-юбилею? Сегодняшняя дата, как бы ее ни называли, двойная: британская версия LXF, которая изначально опережала русскую на 70 номеров, отмечает свой двухсотый выпуск. И если словосочетание «наши британские коллеги» в 2005 и в 2015 годах подразумевает совсем разных людей, то над российской редакцией LXF время, такое ощущение, имеет куда меньшую власть.

Ты помнишь, как все начиналось?..

Старый анекдот гласит, что великие дела начинаются не с «Да! Мы сможем! Мы сделаем это и пойдем дальше!» Великие дела начинаются с «Ну ладно, Бог с тобой, давай попробуем...» И в этом

смысле у LXF изначально были все задатки для того, чтобы стать великим.

Занятно, что поводом к появлению LXF в России стал еще один юбилей — пятилетие Линуксцентра (www.linuxcenter.ru). Тогда он был еще без ставшей ныне привычной приставки «ГНУ», но уже пользовался заметной популярностью в Рунете. Выход в оффлайновые СМИ выглядел вполне закономерным развитием событий. Так что, одним вечером (хочется написать «зимним», но поскольку полной уверенности во времени года за давностью лет нет, опустим детали) мы сели и составили список зарубежных Linux-изданий, которые показались нам особенно интересными.

На тот момент в РФ и странах бывшего СНГ не было недостатка в ИТ-журналах для администраторов (включая, разумеется, и администраторов-линуксоидов). С простыми пользователями и разработчиками дела обстояли похуже. Исходя из этого, мы сформулировали концепцию и составили «шорт-лист» претендентов, среди которых сразу же выделился *Linux Format*. Журнал оказался самобытным, в меру простым, в меру сложным, нешаблонным — где вы еще видели «Что за штука?», например? В общем, выбор был сделан.

Нельзя сказать, что становление LXF в России было таким уж простым. Неожиданности сыпались, как из рога изобилия: то в части тиража страницы окажутся отображенными зеркально, то диски не читаются. Оказалось, что до некоторых частей нашей необъятной родины журнал может идти буквально месяцами, так что ПО на DVD успевают капитально устареть — а ведь с широкополосным Интернетом дела тогда обстояли гораздо хуже. В итоге русская версия стала выходить с двусторонним DVD — на второй, «нашей», стороне размещались зеркала репозитория и архивов с программами или дополнительные отечественные дистрибутивы. Во многих городах журнал из источника информации о Linux превратился в источник Linux — люди брали его друг у друга, копировали DVD и писали нам, что они хотели бы видеть в следующем номере.

Проблему со скоростью доставки удалось несколько сгладить, введя (в первых рядах в РФ!) электронную версию. Конечно, на продажах это сказалось не слишком позитивно — к нашему прискорбию, хоть и не неожиданно, многие переводили (и переводят!) слово “free” исключительно как «бесплатный». Тем ценнее для нас было понимание читателей, отказавшихся выложить свежий номер на торрент-трекер. А чтобы трекеры не простаивали без дела, мы решили выпускать материалы старых номеров LXF под открытой лицензией. Казалось бы, естественный, даже очевидный шаг в мире свободного ПО; однако многие западные Linux-издания придут к нему только спустя 6–7 лет.

Через некоторое время стало понятно, что мало предоставить контент (открыть журналы). Нужно еще сделать его доступным, организовать поиск и установить связи материалов друг с другом.

➤ Таким был первый номер русской версии LXF. Вопрос к знатокам истории: о каком «самом популярном дистрибутиве» речь?



10 ЛЕТ!



Так на свет появилась вики, здравствующая и поныне на wiki.linuxformat.ru. В последние годы темп «викификации» несколько снизился — ведь она, в русле свободного ПО, выполняется силами добровольцев. Кстати, если вы хотите стать одним из них — посмотрите на адрес редакции.

В ногу с Linux

«Гвоздем» первого выпуска русскоязычной версии **LXF** был выход Debian 3.1 Sarge, случившийся, как нетрудно догадаться, незадолго до этого. Старожилы наверняка помнят обложку с крылатым поросенком, обыгрывающую английскую поговорку “when pigs fly” — примерный аналог нашего «Ну наконец-то!». А если вы не застали те славные времена или подзабыли картинку за давностью лет — взгляните на нее здесь (внизу слева).

Мир Linux и свободного ПО в 2005 году был совсем не таким, как сейчас. Ubuntu, на засилие которого на страницах **LXF** периодически жалуются читатели, был тогда молодым и не слишком известным дистрибутивом «с первой страницы DistroWatch». То, что он когда-нибудь будет бить рекорды популярности, казалось такой же фантастикой, как и отказ от Gnome или коричневой гаммы (впрочем, предпосылки, как минимум к первому, уже были). Да и не удивительно: до выхода Gnome 3 оставалось почти шесть лет, и сообщество бурно обсуждало порядок кнопок, стартовое меню сверху экрана и «нечеловеческий» диалог открытия файлов. KDE 4 еще тоже не было. Стандартом де-факто (если опять же судить по популярности) на рабочем столе была снятая ныне с поддержки Windows XP. Вновь прибывшие в стан пользователей Linux тосковали по «неразмытым» шрифтам от Microsoft, и особым шиком считалось пересобрать *FreeType* таким образом, чтобы гарнитуры Microsoft Core Fonts отображались «как в Windows». Разумеется, **LXF** про это писал. *NetworkMa-*

nager еще тоже только-только появился, работающая в Linux беспроводная сетевая карта была все-таки редкостью, но кое-что удавалось запустить через *NdisWrapper*. Вы присылали нам

письма, в которых спрашивали, как заставить все это работать, и мы в очередной раз ломали голову над тем, где взять «железку», чтобы проверить рецепт. Сейчас для подключения к беспроводной сети обычно достаточно выбрать ее SSID в выпадающем меню — хочется верить, что в этом есть и толика наших заслуг.

Шло время, Linux менялся, и **LXF** тоже не стоял на месте. Пожалуй, самое заметное нововведение произошло в 2006 году — у журнала появился новый дизайн (см. вверху справа). Разумеется, мы работали не только над формой, но и над содержанием — потихоньку увеличивали количество оригинальных материалов, запускали новые рубрики. Помните, к примеру, «Школу **LXF**», «Игрострой» или «Историю успеха»? Несмотря на кризисный 2008 (09, 10, далее по вкусу) год, мы неизменно старались, чтобы в **LXF** каждый находил для себя что-то интересное. Стройность концепции, сформулированной на заре становления российской редакции **LXF**, при этом, конечно, страдала, но у единственного печатного издания о свободном ПО, пожалуй, не было другого пути. “Chip

«Время показало, что потраченные усилия не пропали зря.»

Special Linux” закрылся примерно тогда, когда появился **LXF** (надеюсь, что эти события не были связаны), и до сих пор сколько-нибудь значимые попытки выйти на российский рынок с журналом о Linux не предпринимались. И не только в России — наш журнал пережил (к сожалению) и греческую, и итальянскую версию **LXF**. Пользуясь случаем, хочу сказать спасибо всем постоянным читателям, благодаря поддержке которых это стало возможным.

Ближе к пятилетию **LXF** в России стало понятно, что просто писать о свободном ПО недостаточно. Нужно оказывать поддержку успешным проектам, по мере сил влиять на то, чтобы при помощи Linux и СПО можно было решать все новые и новые задачи. Это понимание воплотилось в конкурсе «Лучший свободный проект России». Впервые он прошел в 2009 году, а его победителями стали такие известные и поныне здравствующие проекты, как *Cuneiform*, *Nginx* и *Midnight*

Commander (в очередной раз воскрешенный как раз отечественными программистами). Сказать по правде, не обошлось и без накладок-шероховатостей, но время показало, что потраченные усилия не пропали зря.

До новых встреч!

Десять лет — солидный срок и в мире ИТ, и для ИТ-прессы. Чего только не было за эти 3650 плюс-минус несколько дней: рождались и умирали (или, скажем оптимистичнее, заменялись новыми) инициативы по продвижению Linux и свободного ПО в госсекторе, проходила ежегодная конференция Linux Land (сайт до сих пор доступен — www.linuxland.ru/), серии статей, публиковавшихся на страницах **LXF**, превращались в книги — причем как в России, так и за рубежом. Оглядываясь назад, можно с уверенностью сказать: со своей основной миссией — продвижением Linux и свободного ПО — журнал справился. Посмотрим, что будет еще через 10 лет. **LXF**



В октябре 2006 года журнал обзавелся новым дизайном, с некоторыми вариациями используемым и по сей день.

ДЕСЯТЬ ЛЕТ СПУСТЯ

Основываясь на десятилетнем опыте, Алексей Федорчук размышляет о закономерностях развития Open Source.



В колонке на стр. 13 были выделены три темы, освещенные в первом номере русскоязычного *Linux Format*, которые теперь, по прошествии 10 лет, представляются ключевыми в свете последующего развития «desktop-ного» Linux'a. Напомню, что это были:

- » «устаканивание» ситуации вокруг дистрибутива Mandriva;
- » стабилизация дистрибутива Ubuntu; и
- » выход версии 3.1 дистрибутива Debian.

Теперь же мы посмотрим, как эти ключевые темы выглядят на сегодняшний день.

Тема Mandriva

Начнем с Mandriva, поскольку сделать это проще всего: фирма Mandriva, вместе с одноименным дистрибутивом, прекратила свое существование. Да, осталась ее «общественная» реализация — OpenMandriva, жива и пара форков — Mageia и ROSA, но будущее всех трех видится туманным. Есть еще два ранних клона — AltLinux и PCLinuxOS, которые благополучно существуют (и прекращать этого занятия не собираются). Но их связь

с прародителем (тогда еще Mandrake) — преданья старины глубокой. А в истории самого первого «юзерофильного» дистрибутива, видимо, поставлена окончательная точка.

Собственно, история с Mandriva и послужила толчком для сочинения данного материала. Не так давно о ее завершении шла речь в колонке **LXF197**, при подготовке которой, читая объяснения руководства фирмы в лице Жан-Мануэля Крозе [Jean-Manuel Croset] о причинах банкротства фирмы, я испытал ощущение «дежавю». И действительно, перечисленные им причины почти дословно повторяли те, что приводил Гаэль Дюваль [Gaël Duval], объясняя в интервью для журнала *Linux Format* (см. стр. 66–71 **LXF70**), откуда пошли трудности развития дистрибутива — тогда, в 2005 году, они казались временными.

Но, как все знают, нет ничего более постоянного, чем временные трудности. И десять лет спустя оказалось, что крах фирмы предопределили те же два фактора, о которых говорил Гаэль: трепетное отношение французского КЗОТ'а к интересам трудящихся и нежелание акционеров и прочих инвесторов расставаться с баблом. Первый можно действительно списать на «особенности

национальной охоты». А вот второй имеет непосредственное отношение к теме данного материала.

Во второй половине «нулевых» Гаэль рассказывает, что, когда «...не удалось привлечь дополнительных инвестиций... мы попытались объяснить ситуацию сообществу. Мы сказали, что если они хотят помочь Mandriva, то могут сделать это, приобретая наши продукты, становясь членами MandrakeClub...»

И это сработало:

«...многие люди стали членами Клуба и оказали нам финансовую поддержку. Мы поняли, что стоит хотя бы попытаться найти какой-нибудь выход».

Десять лет спустя тот же номер не прошел. Внешних инвесторов, выручивших Mandriva в 2010 году, во время очередного «кризиса», на сей раз не обнаружилось. Не проявило патриотизма и правительство, как это было однажды, еще в нулевых годах. Но и собственные акционеры отказались восполнить «недостачу». Как не возгорелись энтузиазмом и члены MandrakeClub. Кредит доверия со стороны последних, видимо, был утрачен полностью. Не в последнюю очередь — и вследствие хамского, иначе не скажешь, увольнения Дюваля из им же созданной фирмы...

Тема Ubuntu

Если вся история Mandriva после увольнения Дюваля (2006 г.) — цепь блистающих провалов, чередующихся с редкими зияющими высотами, то развитие Ubuntu во второй половине «нулевых» можно сравнить только с триумфальным шествием Советской власти. Заняв в 2005 г. первую ступень на пьедестале почета Distrowatch'a, этот дистрибутив удерживал ее шесть лет.

О том, что отражает рейтинг Distrowatch'a, разговор будет особый. А пока вернемся к нашей Ubuntu. Число ее пользователей (как применителей, так и тех, кто пытался стать ее потребителями) росло, словно ком со снежной горы. О реальной численности обеих популяций судить трудно за отсутствием объективных

данных. Но, судя по количеству посвященных ей сетевых ресурсов, она действительно превысила популярность всех остальных дистрибутивов Linux'a, вместе взятых.

Кроме того, Ubuntu по ряду причин оказалась прекрасной базой для создания собственных вариаций на эту тему, например, национально-ориентированных: редкая провинция Испании не имела собственного, базирующегося на Ubuntu и предназначенного для делопроизводства дистрибутива в той или иной провинции этой страны. И это — не считая множества клонов с более иными, нежели в родительской системе, рабочими столами по умолчанию. В результате чего очень быстро оказалось, что число дистрибутивов — производных от Ubuntu быстро превысило количество потомков не только родительского Debian'a, но, по

жалуй, и всех дистрибутивов Linux вообще, вместе взятых.

Казалось бы, наш мир семимильными шагами движется к светлому будущему — каким оно виделось линуксоидам всех стран и народов прошлого

тысячелетия: когда Linux в лице Ubuntu утвердится на каждом персональном компьютере — от рабочей станции программиста до гламурного ноутбука домохозяйки. Тем более что в широких народных массах начало утверждаться мнение о том, что Linux — это и есть Ubuntu (а Ubuntu, обратным порядком, Linux). Но тут последовала серия тревожных звонков.

Первый из них прозвонил в момент выхода релиза 11.04, в котором впервые вместо привычного Gnome 2 в качестве рабочей среды по умолчанию предстала Unity. Которая вызвала реакцию, близкую к шоковой, среди всех приверженцев традиционных десктопных ценностей. Реакцию тем более сильную, что еще более шокирующий Gnome 3 не успел тогда укорениться в большинстве распространенных дистрибутивов, кроме породившей его Fedora.

Впрочем, суммарный эффект от внедрения обеих гипермодернистских сред оказался неожиданным (хотя, на самом деле, и предсказуемым): резко активизировалась разработка рабочих

«Казалось, Linux в лице Ubuntu утвердится на каждом ПК.»

Таблица 1. Дистрибутивы Linux: «десятка сильнейших» по версии Distrowatch.com

Год	1	2	3	4	5	6	7	8	9	10
2002	Mandrake	Red Hat	Gentoo	Debian	Sorcerer	SuSE	Slackware	Lycoris	Lindows	Xandros
2003	Mandrake	Red Hat	Knoppix	Gentoo	Debian	Yoper	SuSE	Slackware	Lycoris	Xandros
2004	Mandrake	Fedora	Knoppix	SuSE	Debian	MEPIS	Gentoo	Slackware	PCLinuxOS	Damn Small
2005	Ubuntu	Mandriva	SuSE	Fedora	MEPIS	Debian	Knoppix	Damn Small	Gentoo	Slackware
2006	Ubuntu	openSUSE	Fedora	MEPIS	Mandriva	Damn Small	Debian	PCLinuxOS	Slackware	Gentoo
2007	Ubuntu	PCLinuxOS	openSUSE	Fedora	Sabayon	Mint	Debian	MEPIS	Mandriva	Damn Small
2008	Ubuntu	openSUSE	Mint	Fedora	PCLinuxOS	Debian	Mandriva	DreamLinux	Sabayon	Damn Small
2009	Ubuntu	Fedora	Mint	openSUSE	Debian	Mandriva	Puppy	PCLinuxOS	Sabayon	Arch
2010	Ubuntu	Fedora	Mint	openSUSE	Debian	PCLinuxOS	Mandriva	Sabayon	Arch	Puppy
2011	Mint	Ubuntu	Fedora	Debian	openSUSE	Arch	PCLinuxOS	CentOS	Puppy	Mandriva
2012	Mint	Mageia	Ubuntu	Fedora	openSUSE	Debian	Arch	PCLinuxOS	CentOS	Puppy
2013	Mint	Ubuntu	Debian	Mageia	Fedora	openSUSE	PCLinuxOS	Manjaro	Arch	Puppy
2014	Mint	Ubuntu	Debian	openSUSE	Fedora	Mageia	Arch	Elementary	CentOS	Zorin
2015*	Mint	Debian	Ubuntu	openSUSE	Fedora	Mageia	CentOS	Manjaro	LXLE	Arch

* за последние 6 месяцев

Кто у нас глючит

Расхожее мнение о «глючности» Ubuntu по сравнению с Debian'ом имеет под собой столько же оснований, как и все подобные изустные утверждения «аналитиков». Строго говоря, в плане стабильности с релизами Debian'a можно сравнить только

LTS-релизы Ubuntu, в отношении которых, начиная с версии 8.04, никаких нареканий нет. Релизы же «промежуточные» предназначены для обкатки новшеств, и их следует рассматривать как аналог Debian Testing.

столов, производных от Gnome, но придерживающихся традиций. В их числе — Mate, прямой форк Gnome 2, и Cinnamon, которая первоначально представляла собой попытку построения среды, основанной на технологиях Gnome 3, но «стилизованной под старину». Причем последняя достаточно быстро стала совершенно самостоятельной средой, постепенно утратившей почти все следы своего «третьегономовского» происхождения.

Это привело и к активизации разработки дистрибутива Linux Mint. Первоначально, во «второгномовские» времена, он представлял собой клон Ubuntu, несколько причесанный и приглаженный. И сначала пытался следовать прежней, «второгномовской», линии, оказавшись прибежищем для все тех же ревнителей традиционных ценностей. Что в том же самом 2011 году выводит Mint на первое место в рейтинге Distrowatch'a, которое он не уступает и по сей день.

Следующий звонок, а точнее, уже колокол — ориентация Ubuntu на единую кодовую базу для настольных и мобильных устройств. Что нашло свое отражение в обещаниях внедрить на десктопах в самое ближайшее время Mir и Unity 8. Обещания, которые не выполнены и по сей день (и не будут выполнены в ближайшей перспективе), но которые создавали среди применителей Ubuntu ощущение неопределенности грядущего: создавалось впечатление, что разработчики фирмы Canonical просто «забили» на десктопное направление своей деятельности. При том, что и в направлении мобильном каких-либо фантастических прорывов не наблюдается...

И тут начинается отток от Ubuntu уже не пользователей, а майнтейнеров систем, бывших до сего времени ее более или менее последовательными клонами. Самым значимым событием в этом ряду была активизация разработки Linux Mint Debian Edi-

tion (LMDE), основанной, как нетрудно догадаться, на пакетной базе соответствующего дистрибутива. Но и «национально-ориентированные» системы уже не столь явно подчеркивают своего Ubuntu'йского происхождения. Однако это — уже содержание следующей темы. А пока —

Интермедия о Distrowatch'e

На протяжении всего этого материала встречаются (и будут встречаться) ссылки на рейтинг Distrowatch'a, что требует некоторых оговорок. Автор не относится к этому рейтингу с каким-то особенным трепетом. Его ни в коем случае нельзя считать показателем числа пользователей состоящих в списке дистрибутивов. Более того, он не отражает даже их реальной популярности. Однако, как говаривал товарищ Сталин, других объективных показателей у нас нет — не считать же таковыми некие мифические сведения о числе скачиваний ISO-образов или результаты опросов на форумах. И потому давайте посмотрим, что же этот рейтинг выражает.

Для начала вспомним, что позиция дистрибутива в рейтинге определяется числом посещений его страницы на Distrowatch'e же (для каждой страницы засчитывается один заход с одного IP в сутки). А кто больше всего шляется по сети в поисках наилучшего дистрибутива? Правильно, те, кто еще не определился с его выбором. Ну и просто люди с хронически здоровым любопытством. Так что рейтинг этот отражает, в первую очередь, насколько тот или иной дистрибутив «на слуху». И подвержен сезонным колебаниям, связанным с выходом новых версий. А также — самой обычной моде. Что легко увидеть из таблицы «первой десятки» Distrowatch'a за все годы его существования.

Надо заметить, что на Distrowatch'e существует еще и своего рода «гамбургский рейтинг» — Top Ten Distributions, основанный не на случайных или сезонных колебаниях заходов, а на «широте известности в узких кругах», причем в течение ряда лет. В данный момент их набор и ранжирование такое:

- | | |
|--------------------|--------------------|
| 1 Linux Mint | 7 Arch Linux |
| 2 Ubuntu | 8 CentOS |
| 3 Debian GNU/Linux | 9 PCLinuxOS |
| 4 Mageia | 10 Slackware Linux |
| 5 Fedora | 11 FreeBSD |
| 6 openSUSE | |

Таблица 2. Ubuntu и Debian: сравнительная «свежесть» пакетов

Пакеты	Rafaela	Betsy
Libre Office	4.4.3-rc2-0ubuntu1~trusty1	4.3.3-2+deb8u1
Firefox	39.0+build5-0ubuntu0.14.04.1	38.0.1~linuxmint1+betsy
Pidgin	2.10.9-0ubuntu3.2	2.10.11-1
Zsh	5.0.2-3ubuntu6	5.0.7-5
Geany	1.23.1+dfsg-1	1.24.1+dfsg-1
gthumb	3:3.3.1.is.3.2.7-0ubuntu1	3:3.3.1-2+b2
shutter	0.90.1-0ubuntu1.1	0.92-0.1
Calibre	1.25.0+dfsg-1build1	2.5.0+dfsg-1
Inkscape	0.91.0+47~ubuntu14.04.1	1.4.4+dfsg1-2+b1
Scribus	1.4.2.dfsg+r18267-1ubuntu2	1.4.4+dfsg1-2+b1
Krita	2.8.5-0ubuntu6~ubuntu14.04~ppa1	2.8.5+dfsg-1+b2
musescore	1.3+dfsg-1	1.3+dfsg1-0.1

Можно видеть, что на самом-то деле список этот насчитывает 11 позиций, ибо в него в качестве почетного дистрибутива Linux'a включена FreeBSD. И предлагается этот список в качестве руководства по выбору дистрибутива для тех начинающих (или даже потенциальных) применителей, которые, тем не менее, уже определились со своими возможностями и потребностями.

Тема Debian

Через год после своего появления Ubuntu превзошла Debian многократно и во всех отношениях. Что, однако, прародителю отнюдь не повредило. Наоборот — интенсивное развитие потомка стимулировало и разработчиков Debian'a. Во-первых, с тех пор в этом дистрибутиве достаточно четко выдерживался двухлетний релиз-цикл. Во-вторых, был, наконец, закончен графический инсталлятор этого дистрибутива. А в-третьих и главных — оживился интерес широких народных масс к Debian'у как к настольной системе, ибо до сих пор она имела репутацию в основном системы серверной.

Не снизилось, насколько можно судить по косвенным данным, и число применителей Debian'a. Напротив — ранние версии Ubuntu были, мягко говоря, не свободны от некоторых багов (что поминается этому дистрибутиву по сей день — и без всяких на то оснований). И миграция столкнувшихся с ними применителей на Debian была вполне закономерной.

Однако настоящий подъем популярности Debian'a начался после того, как разработчиков начало заносить в разные стороны, подобно нотариусам из «Периколы». Ряд клонов Ubuntu начал понемногу перебираться на пакетную базу Debian'a. Активизировалась разработка систем, основанных на Debian Testing и Sid. И, наконец, LMDE, о которой уже упоминалось, перестала быть «бедной родственницей» проекта Linux Mint, что в финале завершилось выходом весной этого года LMDE 2, известной под псевдонимом Betsy.

Интермедия о «свежести»

Принято считать, что Debian — дистрибутив очень консервативный и включает в себя версии пакетов времен Скобелева и присоединения Коканда. Тогда как Ubuntu, напротив, полагают дистрибутивом фронтирным, содержащим самые современные релизы тех же пакетов. Однако проверка показала, что эта легенда столь же мифична, как и мнение о «глучности» Ubuntu.

Проверка производилась для релизов Linux Mint 17.2 Rafaela и LMDE 2 Betsy, базовая часть которых целиком заимствуется из Ubuntu 14.04 LTS trusty и Debian 8.1 Jessie, соответственно. Разумеется, сравнивать версии всех пакетов — задача непосильная (да и не нужная). Поэтому в качестве объектов сравнения были выбраны представители трех групп:

- » пакеты, в той или иной мере используемые очень многими применителями (*LibreOffice, Firefox, Pidgin*);
- » пакеты, используемые лично мной, за версиями которых я слежу (*Zsh, Geany, gthumb, shutter, Calibre*);
- » пакеты, требуемые для всякого рода верстальных задач.

Список последних взят из комментария одного из читателей <http://alv.me> и кажется вполне представительным. Результаты его приведены в таблице 2. За буквально двумя исключениями, пакеты в Betsy представлены версиями, идентичными таковым в Rafaela или более новыми. Особенно разительным (для меня) показался пример с *Calibre*: это один из тех редких в наши дни случаев, когда смена номера (от 1.X до 2.X) сопровождалась существенным расширением функциональности.

Предвижу ехидное возражение: Rafaela основана на пакетной базе Ubuntu 14.04 LTS, вышедшей почти полтора года назад,

Betsy же — на базе Debian 8 Jessie, релиз которой состоялся 26.04.2015. Поскольку базовые компоненты обоих родительских дистрибутивов обычно «мажорно» не обновляются, естественно, что версии пакетов в LM на текущий момент будут более старыми. А вот если взять пакеты из PPA — там-то и будет самый «свежак».

Возражение с негодованием отмечается. Во-первых, в основе Rafaela лежит релиз Ubuntu 14.04 со всеми коррекциями (коих за время его жизни было три). А во-вторых, в распоряжении LMDE имеются и официальные репозитории Debian-веток, отличных от Stable, которые и правомерно напрямую сравнивать с пакетами из PPA.

Итоги «десятилетки»

Пора подвести итоги: оправдались ли чаяния и ожидания далекого уже ноль-пятого года, года появления русскоязычной версии журнала *Linux Format*? Не буду оригинальным, если скажу, что в этом вопросе есть две стороны — хорошая и плохая. И, по традиции великих вождей белого, красного и прочих цветов, начну с плохой. Главная надежда — на то, что Linux снизойдет на каждую персоналку — не оправдалась. И, похоже, не оправдается уже никогда. Потому что истории с Mandriva и, в меньшей степени, с Ubuntu в очередной раз показали: бизнес на Open Source в традиционном понимании данного термина невозможен. Как невозможен он на науке, образовании, медицине.

До сих пор практически единственный пример удачного бизнеса вокруг Linux'a и Open Source явил миру Red Hat. Однако тут

имели место быть особые условия: фирма Red Hat вступила на это поприще практически первой, и поприщем ее был американский рынок. Злые языки добавляют к этому, что еще и председателем совета директоров

в ней случайно оказался бывший генерал армии США — а генералы, как нам известно, «бывшими» бывают очень редко. Но мы ведь не верим в теории заговоров, верно?

Вне зависимости от теорий и заговоров, повторить успех Red Hat'a не смогли ни SUSE (как самостоятельно, так и в составе Novell, а затем Attachmate Group), ни Caldera с Corel'ом, ни Canonical (по крайней мере, пока). Компанию же Mandriva этот путь привел к печальному исходу.

Совсем другое дело — использование открытых разработок в коммерческих проектах и государственных программах, от «умных часов» и фитнес-браслетов до космоса и атомного ядра. Но этом занимаются не разработчики операционных систем, не майнтейнеры дистрибутивов и не создатели интегрированных рабочих сред, а совсем другие люди. И в их мире действуют совсем другие законы, нежели законы Computer Science.

Так что в очередной раз повторю банальность, известную со времен как минимум Ньютона с Лейбницем (а то и Аристотеля с Архимедом): единственный способ финансирования Open Source — подтвержденная всей историей науки «экологическая цепочка», состоящая из таких звеньев:

- » дотирование обществом открытых разработок, выполняемых, как правило, «для души» (именно так я в данном контексте перевел бы хрестоматийное Just for Fun);
- » использование их в коммерческих проектах с целью извлечения прибыли и отчуждение части последней на общественные нужды (например, в виде налогов);
- » возврат обществом «части отчужденной части» на дотирование (в том числе и) открытых разработок.

И стабильно успешное (или успешно стабильное) развитие Debian'a и проекта Mint, финансирование которых осуществляется по примерно такой схеме — это та самая хорошая сторона вопроса, которой я закончу данную заметку. **LXF**

«Сравнивать версии всех пакетов — задача непосильная.»

10 ЛЕТ LINUX FORMAT

Евгений Балдин оглядывает свое творческое наследие...

10 лет — это реально века, и вполне можно подвести некоторые итоги. Почти десять лет я ежемесячно писал в *Linux Format* как минимум колонку, число которых, между прочим, сейчас равно 116, то есть минус 4 от 120. Я не профессиональный журналист. Я профессиональный физик. Возможно, это и правильно, так как орфографические ошибки за мной выбирает корректор, а осознать, что современному физико-экспериментатору GNU/Linux как минимум важен, а временами необходим, проще все-таки именно физики. А если это хорошо для физиков, то это, очевидно, хорошо для всех! GNU/Linux для меня — это реально основное рабочее программное окружение уже больше 10 лет.

Кроме десятка-другого одиночных статей в *Linux Format*, я опубликовал серии статей о PostgreSQL, MetaPost и LaTeX, участвовал в написании серий по R, параллельным вычислениям и ПЛИС, взял несколько десятков «электронных интервью» у учителей, преподавателей ВУЗов, бизнесменов и просто хороших людей, продвигавших GNU/Linux на территории нашей необъятной, и даже начал писать «о том, чего под Linux нет», то есть об играх.

Почти все, что было опубликовано в журнале с моим участием, сейчас выложено в открытый доступ под свободной лицензией, так как политика журнала позволяет это сделать через полгода после публикации. За что редакции огромное спасибо! Если что-то еще не выложено, то не выложено оно по причине того, что руки у меня не дошли до этого, а не потому, что мне кто-то ставит палки в колеса в виде копирайта. Две серии в результате такой политики превратились в полноценные книги, и обе эти книги сейчас доступны под свободными лицензиями. Я имею в виду свою книгу о том, как делать книги — «Компьютерная типография LaTeX» (ее, как и ее исходники, можно забрать с моей домашней странички <http://www.inp.nsk.su/~baldin/LaTeX>) и книгу «Наглядная статистика. Используем R!» от группы авторов во главе с А. Б. Шипуновым (код и PDF можно забрать на его страничке: <http://ashipunov.info/shipunov/software/r/r-ru.htm>).

Но что это я о себе да о себе? Есть же именинник! *Linux Format* говорит на русском уже десять лет. Громкий ли этот голос или хриплый, картавый или писклявый — спорить совершенно бессмысленно, так как он есть, и, к сожалению, в окружающей нас действи-



➤ Евгений Балдин позирует. Примерно в 20 км отсюда работает Большой адронный коллайдер, где GNU/Linux используется на полную катушку.

тельности он фактически единственный. *Linux Format* — это эпоха попыток рассказать о мире GNU/Linux по-русски не только через ман-страницы. Это люди, которые считают, что дело свободного программного обеспечения стоит их усилий изложить свой опыт и свое виденье в концентрированном виде на страницах журналах. Это личности, который выслушивают людей, мотивируют и приводят их поток сознания в удобоваримый для восприятия вид. Возможно, не все получается, но мы стараемся! **LXF**

Говорят наши авторы



Александр Толстой

Прекрасно помню номер от октября 2005 года, в котором я был литературным редактором. С тех пор много воды утекло, но я ста-

раюсь время от времени мелькать на страницах **LXF**. Еще бы, ведь это единственный качественный журнал о Linux в России!



Сергей Комков

Низкий поклон за то светлое и радостное, что несет ваш журнал против цифрового зомбирования. Дуду-линукс навсегда!

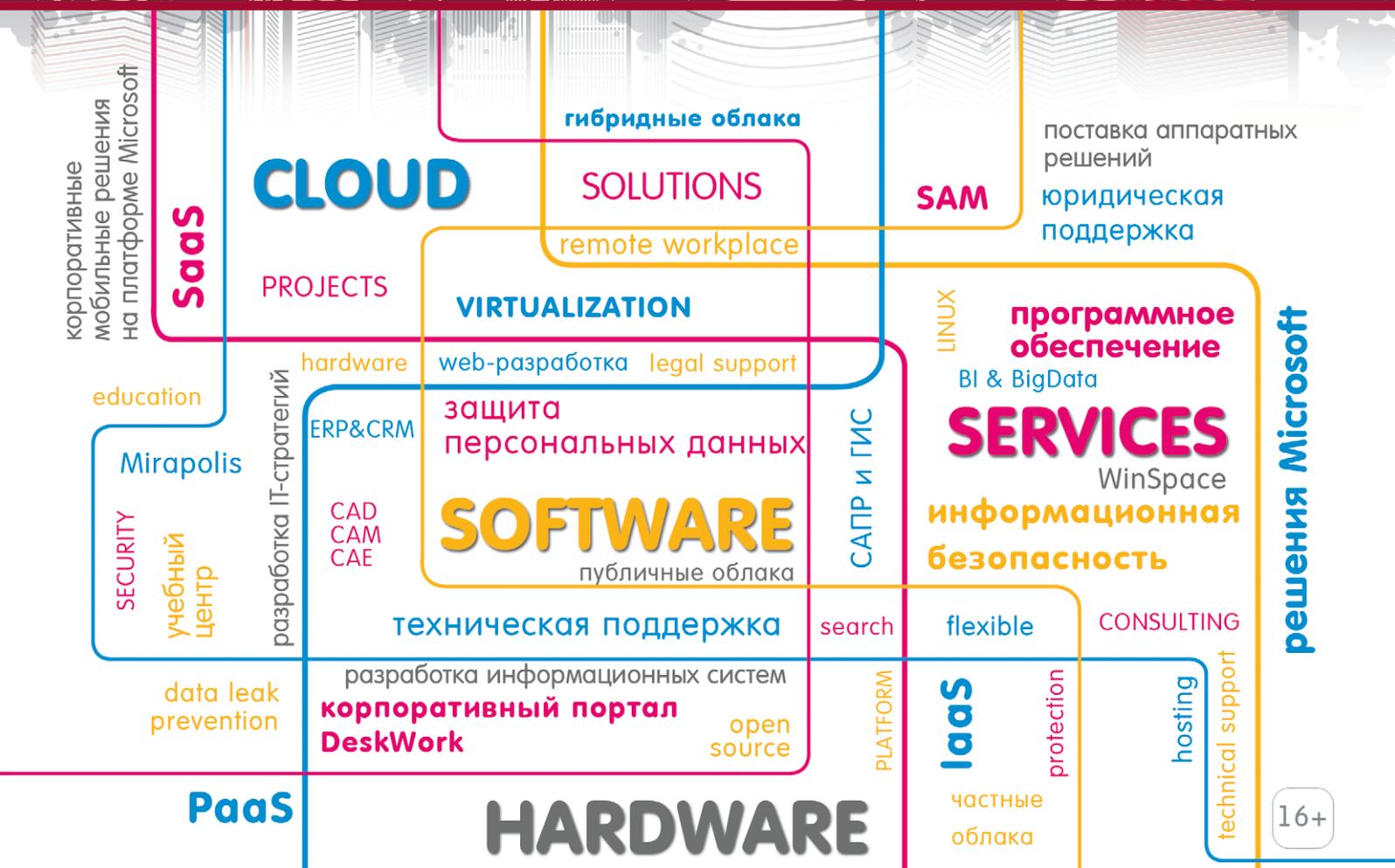
softline®

Cloud Software Hardware Services

20+

Years in IT

IT-архитектура вашего бизнеса



Сборочные системы дистрибутивов

Денис Силаков разбирается, откуда берутся пакеты в репозиториях. Ведь не аист же их приносит?

Большинство современных дистрибутивов распространяют ПО в виде прекомпилированных пакетов, которые представляют собой архив с собранной программой, вспомогательную информацию (в частности, зависимости от других пакетов) и, возможно, скрипты, которые будут исполняться при установке, обновлении или удалении пакета.

Пакеты и репозитории

Пакеты помещаются в репозитории дистрибутива согласно определенным принципам, различающимся от системы к системе. Например, часто выделяют отдельный репозиторий для программ, распространяемых под несвободными лицензиями (“non-OSS” в openSUSE, “non-free” в ROSA и OpenMandriva, “Restricted” в Ubuntu и так далее).

Подробнее о видах репозиториях, сборке пакетов и связанном с этим процессом тонкостях уже рассказывалось в **LXF169**. Там мы выяснили, что за каждым пакетом следит отвечающий за него мейнтейнер, который, в частности, занимается его сборкой. Но за кадром остался немаловажный вопрос: как собранный пакет попадает в репозиторий? Самый очевидный способ для мейнтейнера — это собрать пакет на собственной системе и скопировать его на сервер. Однако у такого подхода есть серьезные недостатки.

Во-первых, на сборку приложений могут оказывать влияние различные настройки системы — переменные среды, флаги компиляции и так далее. В результате в разных системах из одного и того же исходного кода могут получаться разные пакеты. Конечно, можно призывать всех мейнтейнеров использовать одинаковые настройки, но чем больше людей работает над системой, тем больше вероятность, что у кого-то из них настройки случайно сойдутся.

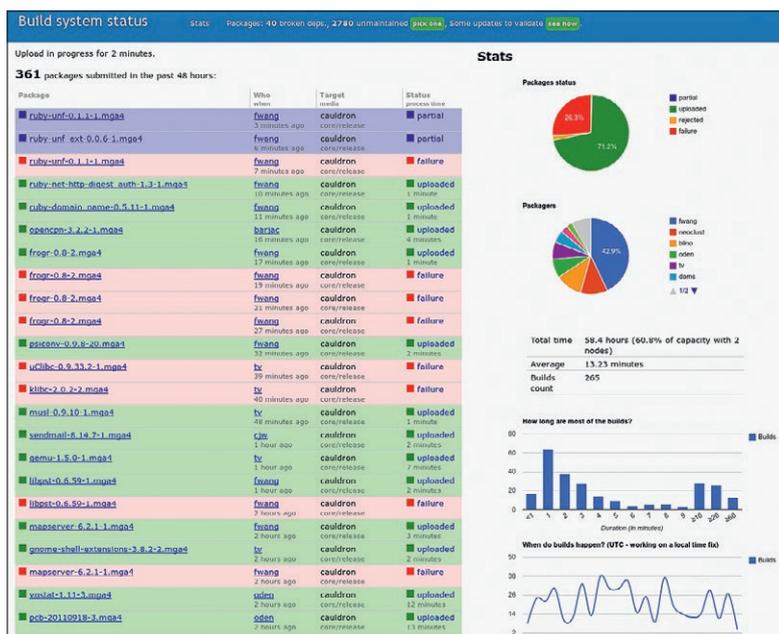
Во-вторых, сам пакет также может оказать нежелательное влияние на систему — вплоть до ее повреждения, особенно если производить сборку с правами суперпользователя. Ведь процесс сборки по факту контролируется скриптом на языке оболочки, но подразумевающим интерактивности — то есть у пользователя никогда не запрашивается подтверждения тех или иных действий. Как результат — небольшая опечатка в скрипте, и вы можете скопировать файлы не внутрь пакета, а куда-нибудь в системные директории. Или еще хуже: вместо удаления ненужных файлов собираемой программы перед ее упаковкой в архив вы можете удалить реальные файлы из своей системы.

В-третьих, при обычной локальной сборке сложно контролировать сборочные зависимости пакета, то есть перечень пакетов, которые должны быть установлены в системе для его успешной сборки. Мейнтейнер может забыть указать какую-нибудь зависимость, но если эта зависимость уже установлена в его системе (например, для сборки других пакетов), то пакет все равно успешно соберется.

Однако если в будущем эта зависимость будет удалена из системы либо пакет решит собрать другой мейнтейнер, в системе которого нужные зависимости отсутствуют, то пакет не соберется. И вместо полезных улучшений в пакете придется потратить некоторое время, чтобы разобраться, почему он собрался раньше, но не собирается сейчас.

Для борьбы с этой проблемой сборку можно производить не в рабочей системе, а в отдельной виртуальной машине или chroot-окружении, где по умолчанию установлен только базовый набор пакетов, а остальные устанавливаются непосредственно перед сборкой на основе перечня сборочных зависимостей пакета. Такой подход позволяет проверить полноту сборочных зависимостей, но требует существенно больше ресурсов и времени — ведь на установку зависимостей может уйти больше времени, чем на все остальное (компиляцию программы, ее помещение в архив и так далее).

Наконец, большинство дистрибутивов поддерживает несколько аппаратных архитектур, и пакеты надо собирать под каждую из них. И по завершении сборки полученные пакеты необходимо поместить на сервер с репозиториями дистрибутива. Все это также требует дополнительного времени и ресурсов (в частности, для «экзотических» аппаратных платформ может потребоваться



➤ Web-интерфейс сборочной системы Mageia позволяет наблюдать за заданиями, но не управлять ими.

виртуальная, а то и реальная машина) и добавляет мейнтейнеру рутинной работы.

В общем, минусов у подготовки пакетов на машине мейнтейнера хватает. Неудивительно, что во многих дистрибутивах (особенно в тех, где над пакетами работает большое количество мейнтейнеров) используется другой подход, о котором мы и расскажем далее.

Системы сборки

Для решения перечисленных выше проблем разработчикам дистрибутивов обычно выделяются отдельные сервера с нужными настройками, на которых и производится сборка. Часто набор этих серверов называют «сборочной фермой». Использование централизованной сборочной среды обеспечивает единое сборочное окружение для всех пакетов и позволяет выполнять сборку сразу для всех аппаратных архитектур, поддерживаемых дистрибутивом. При этом архитектура фермы исключает влияние собираемых пакетов на сборочную среду. В дополнение к собственно сборке, система может производить различные автоматизированные проверки собранных пакетов и смотреть, соответствуют ли они правилам сборки, принятым в дистрибутиве.

В случае успеха система сборки самостоятельно подписывает получившийся пакет ключом дистрибутива и помещает его в соответствующий репозиторий. Факт помещения пакета в репозиторий сразу после сборки очень важен на активных стадиях подготовки новой версии дистрибутива — ведь все пакеты системы сильно взаимосвязаны, и важно, чтобы каждый пакет собирался и работал с актуальной версией своих «соседей».

Помимо сборки отдельных пакетов, сборочные фермы могут позволять собирать ISO-образы дистрибутивов — как работающие в Live-режиме, так и используемые для установки системы на жесткий диск.

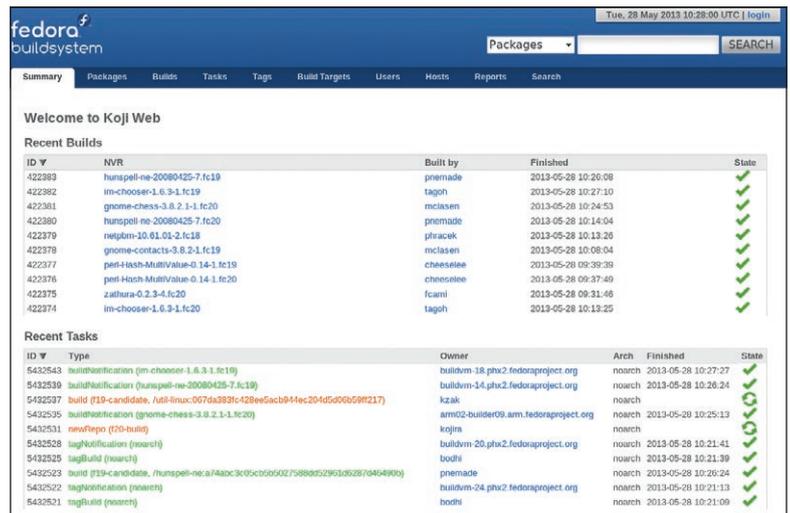
Архитектура сборочных ферм

Сборочных ферм существует достаточно много (хотя и существенно меньше, чем дистрибутивов Linux). Многие из них «заточены» под сборку конкретного дистрибутива и сильно привязаны к его инфраструктуре и серверам. Есть и универсальные решения, которые можно использовать при разработке практически любого представителя семейства Linux. Однако хотя реализации всех этих систем могут различаться очень сильно, их внутренняя архитектура в целом одинакова: за годы развития, общения друг с другом и заимствования удачных идей многие разработчики пришли к схожим взглядам на то, как должна быть устроена сборочная ферма дистрибутива.

Непосредственно сборка пакетов производится в некоторой виртуальной среде, изолирующей сборку от остальной системы. Распространенным вариантом такой среды является chroot, подразумевающий изменение корневого каталога работающей программы (в нашем случае — сборки пакета).

Chroot является легковесным решением, не требующим накладных расходов на поддержку виртуальной машины. Недостатком этого является низкая безопасность — при сильном желании, вредоносная (или просто ошибочно написанная) программа может выйти за пределы chroot-окружения и начать орудовать в реальной системе. Поэтому chroot хорошо подходит для проверки полноты сборочных зависимостей, но если задуматься о безопасности и надежности сборочной системы, то стоит выбрать нечто более серьезное.

Вместо chroot можно использовать полноценную виртуальную машину — например, QEMU или VirtualBox. Выйти за пределы



► В любой момент времени на Koji собирается какой-нибудь пакет для Fedora.

такой машины «случайно» практически невозможно. Встречаются и подходы с двойной изоляцией — когда для каждой сборки запускается отдельная виртуальная машина, внутри которой используется chroot. Такой подход оправдан, если в дистрибутиве виртуальной машины по умолчанию установлено слишком много пакетов, что мешает проверке полноты сборочных зависимостей.

Сборочные системы большинства дистрибутивов тесно интегрированы с системами контроля версий (СКВ), в которых хранится исходный код для сборки пакетов (часто — сразу в виде архивов, взятых из вышележащих программ [upstream]), патчи и инструкции по сборке (спрес-файлы для RPM и управляющие скрипты сборки для Deb). Использование СКВ позволяет отслеживать изменения в пакетах, а также поддерживать сразу несколько версий одного пакета для разных версий дистрибутива.

Основной сценарий взаимодействия мейнтейнера со сборочной системой — это помещение обновленных файлов для сборки в СКВ и отправка задания, в котором указывается, какой пакет надо собрать и в какой репозиторий поместить результат. Взаимодействие может происходить через web-интерфейс, но практически все системы предоставляют возможность удаленного управления заданиями с помощью соответствующих утилит командной строки.

Такая возможность очень полезна, если вам требуется послать на сборку несколько десятков пакетов — написать

скрипт в одну строку в командной строке явно быстрее, чем много раз щелкать мышью в браузере. Более того, многие системы ограничиваются взаимодействием через командную строку, а через Web можно только наблюдать за их работой. Таковы, например, сборочные системы дистрибутивов Debian и Mageia.

Как мы уже отметили выше, многие системы сборки сильно привязаны к конкретному дистрибутиву. Если вы решите сделать похожую систему для своих нужд, то позаимствовать из этих систем вы сможете разве что идеи. А необходимость развертывания таких систем возникает не так редко, как может показаться на первый взгляд. Ведь использовать сборочную ферму можно не только для сборки полноценного дистрибутива, но еще и для сборки пакетов под ту или иную вариацию Linux. А это актуальная задача для многих разработчиков ПО, которые не хотят полагаться на мейнтейнеров дистрибутивов и предпочитают выкладывать готовые пакеты под разные системы непосредственно на своих сайтах.

К счастью, существуют и более универсальные решения. Ниже мы рассмотрим наиболее функциональные сборочные системы, ►

«За годы многие разработчики пришли к общим взглядам.»

The screenshot displays the OBS web interface. At the top, there are navigation tabs: Downloads, Support, Community, and Development. Below this is a 'Welcome to openSUSE Build Service' section with a brief description of the service and links to 'All Projects', 'Search', and 'Status Monitor'. A 'System Status' section features a line graph showing the number of active build jobs from May 22 to May 28. Below the graph, text indicates that 447 of 449 build hosts are busy. A 'Latest Updates' section lists recent updates for packages like python-quantumclient, python-ke...neclient, python-glanceclient, python-cinderclient, openstack-swift, and openstack-nova. The bottom of the page contains 'Locations', 'Help', and 'Contact' sections with various links.

➤ Масштабы сборки пакетов на <http://build.opensuse.org> поражают воображение.

код которых открыт и которые при необходимости могут быть адаптированы к нуждам любого проекта.

Koji

Начнем с Koji — простого представителя сборочных ферм, используемого при сборке Fedora и репозитория EPEL (Extra Packages for Enterprise Linux — дополнительные репозитории для Red Hat Enterprise Linux). Web-интерфейс позволяет производить базовое управление заданиями сборки, а также просматривать всевозможную статистику — список текущих и завершенных задач, загруженность серверов, наиболее активных мейнтейнеров и так далее.

Внутреннее устройство Koji четко соответствует описанной выше архитектуре: все необходимое для сборки хранится в СКВ (хотя допускается загрузка файлов из Интернета непосредственно перед сборкой), мейнтейнеры могут работать с этой СКВ удобным для них способом (с помощью утилит командной строки или инструментов с графическим интерфейсом), а работать с самим сервером Koji удобнее всего из консоли.

Отличительной чертой Koji является возможность ее использования для других проектов. Код этой системы открыт и относительно невелик, а настройка не представляет особой сложности. Впрочем, если вы захотите серьезно расширить систему (например, добавить поддержку дистрибутивов, сильно отличающихся от Fedora и Red Hat), то вам придется изрядно повозиться.

Open Build Service

Open Build Service (OBS) изначально назывался openSUSE Build Service и, как нетрудно догадаться, использовался для сборки openSUSE. В настоящее время OBS поддерживает сборку и для ряда других систем, среди которых как дистрибутивы, использующие формат пакетов RPM (Fedora, RedHat, старые версии Mandriva), так и использующие формат Deb — Ubuntu и сам Debian.

Некоторые проекты используют свои собственные инсталляции OBS (например, отдельно развернутая инфраструктура OBS использовалась для сборки MeeGo, а сейчас применяется при разработке его последователя Tizen). Наиболее крупной инсталляцией по-прежнему является OBS, развернутый на сайте <http://build.opensuse.org>. Основной задачей сервиса является сборка openSUSE, но можно готовить пакеты и для других популярных дистрибутивов.

OBS был первым сервисом, предложившим удобную возможность сборки пакетов под разные варианты Linux. В принципе, реализовать это технически не так уж и сложно. Ведь непосредственно сборка происходит в полностью изолированных виртуальных машинах, и задача сервиса сводится к запуску машины с нужной ОС, помещению в нее всех необходимых файлов и инициации процесса сборки. Безусловно, остается реализовать множество технических тонкостей (поскольку процесс сборки в каждом дистрибутиве надо инициализировать по-разному), но никаких фундаментальных преград поддержке многих дистрибутивов нет.

Неудивительно, что OBS используется не только мейнтейнерами openSUSE, но и разработчиками, желающими собирать свои программы под разные дистрибутивы. Правда, публичный сервис OBS подойдет только для проектов с открытым исходным кодом, однако разработчики коммерческих продуктов могут развернуть отдельную копию OBS на своих машинах.

Еще одним типичным сценарием использования OBS является создание своего собственного варианта того или иного приложения, уже присутствующего в репозиториях openSUSE (например, вы хотите добавить свою заплатку-патч, который упорно не желают принимать мейнтейнеры). Для этого вам надо найти на OBS соответствующий проект и клонировать его в свой репозиторий, после чего можно смело экспериментировать с кодом, собирать пакеты и устанавливать их в свою систему. Можно также отправить свои изменения на рассмотрение разработчикам исходного проекта, так что они смогут в удобной форме просмотреть изменения и, если все понравится, принять их одним щелчком мыши.

Работать с OBS можно как из командной строки, так и через полнофункциональный web-интерфейс. Более того, доступны плагины для использования сборочной системы из Eclipse и Qt Creator.

Automatic Build Farm

Automatic Build Farm (ABF) — разработка отечественной компании POCA, используемая в процессе создания линейки дистрибутивов ROSA (как ROSA Desktop, уходящего корнями в Mandriva, так и ROSA Server, основанного на RedHat Enterprise Linux). Также ABF в настоящее время является основой разработки OpenMandriva.

В отличие от многих аналогов, ABF хранит в СКВ (в роли которой выступает Git) только текстовые файлы, для которых можно не просто хранить разные версии, но и просматривать изменения между ними средствами самой СКВ. Все двоичные файлы (в частности, архивы с исходным кодом) выносятся в отдельное файловое хранилище, а в Git помещается служебный файл **.abf.yml** с перечнем двоичных файлов, относящихся к пакету. Перед сборкой ABF считывает содержимое **.abf.yml** и извлекает необходимые файлы из хранилища. При этом при обновлении двоичных файлов их старые версии из хранилища не удаляются, так что вы в любой момент можете откатиться к старым версиям, просто откатив содержимое файла **.abf.yml**.

Еще одна отличительная черта ABF заключается в том, что это не просто система для сборки пакетов, но полноценная

инфраструктура для ведения разработки приложений. ABF предоставляет программистам Git-репозиторий для хранения исходного кода, легковесный трекер ошибок и запросов, проектную вики и ряд других полезных инструментов для коллективной работы команды программистов.

Такой набор возможностей позволяет вести разработку программ непосредственно в ABF, не используя сторонних сервисов наподобие SourceForge или GitHub. И здесь же, на ABF, собирать пакеты под различные дистрибутивы — благо, ABF позволяет брать исходный код для сборки непосредственно из Git-репозитория вашего проекта. Подобным образом разрабатываются уникальные приложения РОСЫ — такие, как медиа-проигрыватель *ROSA Media Player* и инструмент быстрого просмотра файлов *Klook*.

Как и OBS, система ABF предоставляет полнофункциональный web-интерфейс и утилиты для работы из командной строки. Имеется также клиент для Android, позволяющий не только запускать сборки и наблюдать за ними, но и вносить изменения в код проектов.

Launchpad Build Farm

В завершение: нельзя обойти стороной процесс сборки пакетов для Ubuntu, разработчики которого также совместили процесс разработки ПО с процессом сборки в инструментарию Launchpad. Однако если для OBS и ABF сборка пакетов является первоочередной задачей, то основная цель Launchpad — это управление процессом разработки ПО. Здесь к вашим услугам системы контроля версий и отслеживания ошибок, средства планирования, интеграция с инструментами локализации и многое другое. Все это доступно через приятный и удобный web-интерфейс.

К сожалению, сборке пакетов из исходного кода в Launchpad уделяется не так много внимания, как процессу разработки программ. Набор серверов для сборки пакетов присутствует и именуется Launchpad Build Farm, но взаимодействовать с ним необходимо из командной строки. Как и Debian, в Ubuntu рекомендуется предварительно подготовить и проверить пакет на своей локальной машине, а уже потом загружать пакет с исходным кодом для

Launchpad.net
 Launchpad Home Code Bugs Blueprints Translations Answers

The Launchpad build farm

119 available build machines, 4 disabled and 101 building of a total of 123 registered.

Machine	Status
amd64 i386	
aatxe	Idle
akateko	Idle
allspice	Idle
balSu	Idle
kisset	Idle
klock	Idle
komainu	Idle
lamiak	Idle
orlo	Idle
panlong	Idle
phianna	Idle
roseapple	Idle
tipua	Idle
toyoi	Idle
arm64	
auburn	Building arm64 build of bash 4.3-13ubuntu1 in ubuntu wily PROPOSED
beebe	Building arm64 build of unity-settings-daemon 15.04.1+15.10.20150818-0ubuntu1 ppa-service/ubuntu/landing-016]
magic	Building arm64 build of messaging-app 0.1+15.04.20150818-0ubuntu1 in ubuntu v service/ubuntu/landing-009]
templar	Building arm64 build of pcl 1.7.2-9 in ubuntu wily PROPOSED
twombly	Building arm64 build of unity-control-center 15.04.0+15.10.20150818-0ubuntu1 in service/ubuntu/landing-016]

» Web-интерфейс Launchpad Build Farm приятен, но функциональных возможностей в нем немного.

сборки на ферме. Это относится как к пакетам, идущим в официальные репозитории дистрибутива, так и к пакетам из персональных архивов (Personal Package Archive, PPA), в которые каждый пользователь Launchpad может собирать свои программы.

Через web-интерфейс можно наблюдать за загруженностью сборочных серверов и за тем, какие задания на них выполняются (и даже наблюдать за процессом сборки в «живом» режиме), но возможностей управления сборками через браузер, да и просто интеграции с остальными компонентами Launchpad (хотя бы с системой контроля версий) явно недостаточно.

Одной из возможных причин такой ситуации является то, что Launchpad, хотя и разрабатывается компанией Canonical, может использоваться для ведения проектов, никак не связанных с Ubuntu — то есть можно рассматривать Launchpad как аналог SourceForge, GitHub и им подобных. Функциональность же сборки пакетов ограничена Ubuntu и производными и используется преимущественно мейнтейнерами дистрибутивов, а не разработчиками программ.

Заключение

Итак, при подготовке дистрибутивов Linux мейнтейнерам обычно помогают сборочные системы, автоматизирующие рутинные задачи и избавляющие людей от множества забот. Эти же системы могут применяться пользователями, обладающими некоторыми навыками программирования, для сборки своих собственных программ или своих вариаций стандартных приложений под тот или иной дистрибутив. Некоторые представители сборочных ферм могут быть развернуты локально, что позволяет использовать их в других проектах — либо для сборки дистрибутивов, либо для сборки конкретных приложений.

Схема организации таких систем уже хорошо проработана, и принципы их функционирования во многом схожи. Впрочем, прогресс не останавливается, и развитие сборочных ферм идет в сторону наращивания функциональности — будь то поддержка нескольких целевых платформ или возможность вести разработку там же, где собираются пакеты. Приятно отметить, что и отечественные разработки, во-первых, существуют, а во-вторых — выглядят очень достойно на общем фоне. LXF

Automated Build Farm
 A distributed environment to build distributions, supporting all steps from managing source code to creating ISO images.

Project Management
 Every git repository in ABF is accompanied by tools necessary to manage a project, be it a public project or a private one. more...

Source Code
 ABF is full-functional gh-hosting with convenient Web interface. more...

Package Building
 ABF is not just a hosting for source code, but a system to build and publish packages. Build your projects for large variety of distributions and hardware platforms in your personal repository! more...

» Стартовая страничка ABF подчеркивает, что сборка пакетов — не единственное назначение этой системы.



По советам м-ра Брауна

Джолцион Браун

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

Бокал за Линуса

Моя первая попытка овладеть GNU/Linux была не слишком успешной. В середине-конце 1990-х я официально «обучался на работе», приглядывая за маленькой офисной сетью из Windows и Novell NetWare. Все это выглядело бедновато по сравнению с рабочими станциями UNIX, которыми я пользовался в университете, но они стоили слишком дорого и были недоступны. Когда я занимался поиском вариантов для почтового сервера, я вдруг услышал разговор про нечто под названием 'Debian'.

Я был поражен. Я правда могу бесплатно загрузить UNIX-подобную ОС и посмотреть на код? Я приступил к делу и попробовал загрузить Debian 1.3 (насколько я помню) с помощью интернет-доступа, который я буквально выиграл в лотерею. Не стоит и говорить, что это заняло целую вечность, связь несколько раз обрывалась, и в конце концов я плюнул и сдался. Фи! Я снова заглянул в этот дивный новый мир, только когда получил некий активный журнал того времени с дистрибутивом Linux на диске.

Потом я отправился в дебри корпоративных IT-инфраструктур и наблюдал, как ростки Linux медленно пробиваются в дата-центры вопреки явному противодействию (помню, как мой тогдашний коллега восхищенно показывал мне новый журнал о Linux, который он только что купил — его 200-й номер вы сейчас читаете). Конечно, ныне Linux используется повсеместно и стал де-факто стандартом для всего «облачного».

Не будет преувеличением сказать, что маленький проект Линуса в конце концов определил мою карьеру (и карьеру многих других) на годы вперед — и с меня бокал-другой пива (а то и три).
jolyon.brown@gmail.com

› SourceForge долго поддерживал открытое ПО, но сталкивается с конкуренцией на жестком рынке репозитория программ.

Эзотерическое системное администрирование из таинственных закоулков серверной.



SourceForge под огнем

А также: конец близок — Apple открывает Swift, а Microsoft, наконец, добавляет OpenSSH в Powershell.

SourceForge, почтенный репозиторий программ, попал под огонь критики от крупных проектов после того, как было обнаружено, что в загрузочных пакетах программ появились рекламные вставки. Эти вставки позволяют пользователям дополнительно загрузить коммерческие пакеты, но, как сообщалось, в некоторых из включенных пакетов обнаружен вредоносный код.

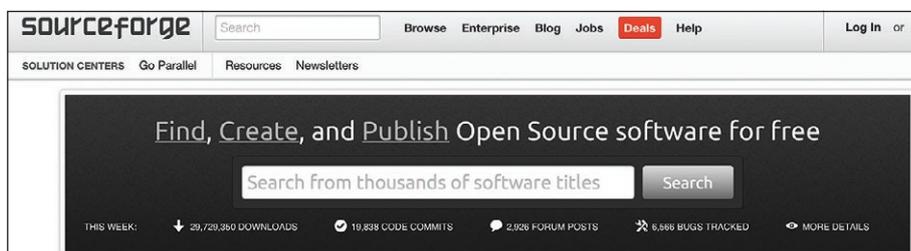
Политика сайта состоит в свертывании учетных записей и проектов, которые больше активно не поддерживаются — например, GIMP, который забросил свой сайт — и использовании рекламных вставок для получения прибыли. Рекламная платформа под названием DevShare ранее устанавливалась по выбору пользователя, но потом стала устанавливаться автоматически. В ответ на шум SourceForge выпустила заявление (<http://bit.ly/3rdPartyOptInOnly>), в котором говорилось, что «в ответ на беспокойство... мы в SourceForge хотели бы отметить, что перестали включать рекламные материалы от третьих лиц в неподдерживаемые проекты».

В еще одной выявленной уязвимости OpenSSL под кодовым названием 'Logjam' ничего явно вредоносного нет. Она похожа на ранее выявленную

атаку FREAK [«Советы», стр. 62 LXF197] и столь же опасна. Она позволяет атакующему снизить степень защиты между двумя конечными точками и проанализировать трафик. Но не волнуйтесь, заплатки для решения этой проблемы уже выпущены.

Apple неожиданно объявила, что позднее в этом году откроет свой новейший язык Swift, а программы на нем будут запускаться в Linux, а также на платформах Apple OS X и iOS. Этот шаг должен способствовать принятию нового языка разработчиками.

Microsoft также анонсировала, что OpenSSH наконец будет поддерживаться в PowerShell (что я предполагал в LXF199). В своем анонсе команда, ответственная за поддержку, не без подтекста сообщила следующее: «...это уже третий раз, когда команда PowerShell пытается включить поддержку SSH. Первые попытки предпринимались для PowerShell V1 и V2 и были отклонены. С учетом изменений в руководстве и культуре компании мы решили попробовать еще раз, и на этот раз компания очень поддерживает нас, так как мы смогли показать явную и серьезную ценность поддержки OpenSSH для клиентов». К сожалению, я вряд ли смогу приписать себе эту разработку.



Amazon Web Service и Ansible

Любимая конфигурационная утилита м-ра Брауна поможет вам разобраться с облачной платформой Amazon с командной строки.

В последнем отчете Gartner об облачной инфраструктуре как сервисе платформа Amazon Web Services (AWS) была отмечена добавлением в «Магический квадрант» (что на языке Gartner означает причисление к разряду лучших). Хотя OpenStack остается наиболее открытой из облачных «операционных систем», AWS — это платформа, о которой важно знать. Хотя лично я предпочитаю более открытые экосистемы, очень вероятно, что вам приходилось или придется поработать с AWS для развертывания на ней экземпляров Linux и для запуска всех приложений. И я думаю, что стоит рассказать о ней, хотя бы на каком-то базовом уровне. Конечно, при возможности хотелось бы делать это с командной строки и сохранить все, что я сделаю, в системе управления версиями. Я обращаюсь к *Ansible*, моей любимой на данный момент утилите управления конфигурацией (о ней я кратко писал в **LXF197** и **LXF196**).

Итак, что же мы будем развертывать? Недавно меня попросили проиллюстрировать идею для клиента, который хотел проверить, сможет ли он перенести свои сайты на WordPress на Amazon (с более традиционного хостинг-провайдера). Клиент хотел получить большую гибкость и легкую масштабируемость, когда один из его сайтов осадил клиенты после рекламы в социальных сетях, а также утилиту управления конфигурацией для выполнения всех этих действий. В этом и в следующем месяце я попробую в общих чертах описать, что я сделал и как я это сделал. В этом номере я опишу основы, а в следующем обращаюсь к более продвинутым возможностям, которые AWS предлагает для таких сценариев. Я также постепенно заменяю обращения к консоли управления EC2 с веб-интерфейсом на обращения к *Ansible*, насколько это возможно.

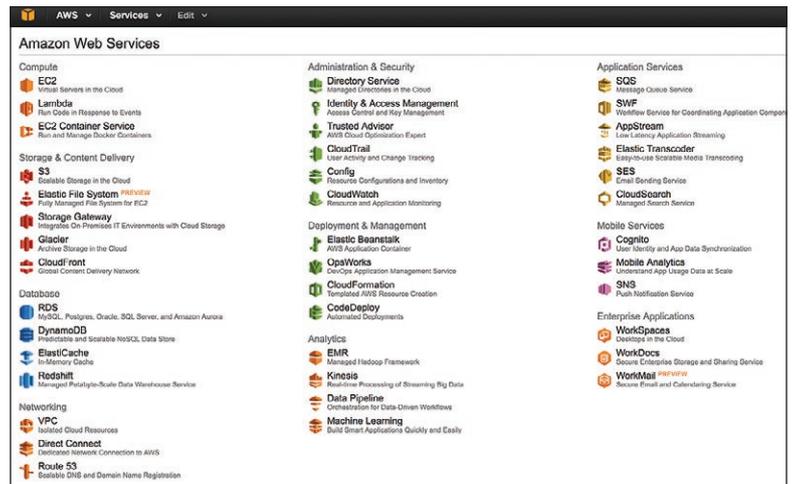
Для следующих примеров есть несколько предварительных требований. Необходима учетная запись AWS (она бесплатна; зайдите на <http://aws.amazon.com> и зарегистрируйтесь); также потребуется локально установить *Ansible*, в чем вам поможет старый добрый *Git*. Я использую чистый дистрибутив Ubuntu 14.04 (с *Ansible 1.8.2*). Также пришлось установить пакет *python-boto* (командой `sudo apt-get install python-boto`), это интерфейс Python для AWS. Для новичков, желающих войти в курс дела, на сайте AWS есть масса полезных руководств и видеороликов. Признаюсь, я был слегка ошарашен текущим количеством продуктов AWS — оно показано

на скриншоте (справа). Лучший способ познакомиться с системой — как всегда, окунуться в нее и попробовать что-нибудь сделать. Поэтому без дальнейших предисловий начинаем.

Есть у меня и ключ, и секрет...

Под именем моей учетной записи в AWS есть выпадающее меню с пунктом *Security Credentials* [Учетные данные безопасности]. Для выполнения любых действий в AWS мне нужно сгенерировать пару ключей — ключ доступа и секретный ключ доступа. Для лучшего разграничения прав доступа Amazon рекомендует настраивать несколько учетных записей IAM (Identity and Access Management), каждая со своими ключами доступа. Это имеет смысл, когда несколько человек пользуются общей корпоративной учетной записью с отдельными сферами ответственности. Так как я собираюсь только проиллюстрировать идею, я ограничусь первоначальным набором ключей, который загрузил в виде файла.

Для использования *Ansible* также нужна пара связанных файлов: `ec2.py` и `ec2.ini`. Их можно загрузить с <https://raw.githubusercontent.com/ansible/ansible/devel/plugins/inventory/ec2.py> и <https://raw.>



» AWS обвиняли в том, что он сложноват для новичков. Прямо и не знаю, откуда взялось это мнение...

Все аббревиатуры AWS из трех букв!

Первый опыт с Amazon Web Service бывает трудноватым, во многом из-за избытка применяемых сокращений. Несколько мы упомянем в этой статье, и вам могут пригодиться следующие термины:

- » **VPC (Virtual Private Cloud — виртуальное частное облако)** Эластичная сеть, заполняемая инфраструктурой, платформой и серверами приложений, которые разделяют общую безопасность и среду подключений.
- » **EC2 (Elastic Compute Cloud — эластичное вычислительное облако)** Web-сервис, позволяющий запускать экземпляры серверов Linux/UNIX и Windows и управлять ими в дата-центрах Amazon.

- » **AMI (Amazon Machine Image — образ компьютера Amazon)** Зашифрованный образ компьютера. Образы компьютеров Amazon фактически представляют собой шаблон корневого диска компьютера. Они содержат ОС и несколько основных программ.
- » **PV (Paravirtual — паравиртуальный)** Тип паравиртуальных экземпляров, работающих на оборудовании без явной поддержки виртуализации, которые не могут использовать специализированные аппаратные расширения, типа улучшенной сети или обработки графическим сопроцессором.
- » **HVM (Hardware Virtual Machine — аппаратная виртуальная машина).** Аппаратные виртуальные

машины (VM) имеют полностью виртуализованный набор аппаратных устройств, позволяющий запускать ОС прямо поверх VM, без модификаций, как если бы она запускалась на голом железе. Amazon рекомендует их для наилучшей производительности, особенно в сочетании с последним поколением вычислительных экземпляров.

- » **T2, M4, M3, C4, C3, R3, G2** Все доступные типы вычислительных экземпляров: например, T2, M4 и M3 — вычислительные экземпляры общего назначения. Amazon предлагает различные типы аппаратных устройств и VM. Список можно найти здесь: <http://aws.amazon.com/ec2/instance-types>.

githubusercontent.com/ansible/ansible/devel/plugins/inventory/ec2.ini соответственно.

Скрипт `ec2.py` используется для динамического перечня хостов — так как в облачном окружении хосты могут появляться и исчезать, традиционного статического перечня, который применяется в других конфигурациях *Ansible*, не вполне достаточно. Файл `ec2.ini` содержит конфигурацию скрипта Python. В нем есть несколько параметров (например, используемый по умолчанию регион Amazon), но пока я оставляю их без изменений.

Загрузив `ec2.py` и `ec2.ini` и скопировав их в `/etc/ansible`, я установил для них следующие права доступа:

```
$ sudo chmod 755 /etc/ansible/ec2.py.
$ sudo chmod 644 /etc/ansible/ec2.ini
$ sudo chown root:root /etc/ansible/ec2*
```

Теперь мне надо создать ключевую пару для входа во все вычислительные экземпляры, которые я буду запускать. Это обычная пара из открытого и закрытого ключей SSH — не то же самое, что ключ доступа и секретный ключ доступа AWS, которые я загрузил. При желании я мог бы загрузить существующую пару ключей, созданную известной утилитой `ssh-keygen`, но для нашего урока мне столь же просто будет создать новую ключевую пару с помощью консоли управления EC2 в браузере. После указания имени ключей (Ixfkeys — очень оригинально) ключи будут автоматически загружены в формате PEM.

Затем я создаю новый каталог для репозитория *Ansible* и инициализирую его как репозиторий git:

```
$ mkdir aws-example; cd aws-example
$ git init
```

Уверен, что есть люди, способные запоминать идентификаторы образов компьютеров Amazon. Надеюсь, мне не доведется их встретить.

При работе с AWS через *Ansible* я обнаружил, что большую часть времени запускаю задачи для своей локальной рабочей станции. Это отличается от обычной модели *Ansible*, в которой они запускаются для удаленных компьютеров. В модулях, которыми я пользуюсь ниже, выполняются вызовы к различным функциям API AWS, чтобы затем выполнить необходимые действия от моего имени. В среде предприятия советую зарезервировать экземпляр EC2 только для запуска задач управления в самой сети AWS. При этом придется подумать о деньгах: запуск отдельного экземпляра только для этой задачи при небольшой системе — перевод денег, но в очень крупных может сойти за ошибку округления. Разумеется, прелесть облака в том, что экземпляры можно быстро останавливать и запускать.

Пожалуйста, прочтите руководство

У Amazon есть прекрасный список технической документации по AWS, на которую стоит взглянуть. Она охватывает как частные, так и общие вопросы, например, безопасность (<http://aws.amazon.com/whitepapers>). Выясняя, как запустить *WordPress* на AWS, я заинтересовался несколькими документами. У меня есть два варианта: запустить набор стандартных узлов EC2 или воспользоваться системой Elastic Beanstalk PaaS (Platform as a Service — платформа как сервис) для настройки высокодоступного сервиса более сложным образом. Среди основных модулей *Ansible* многие поддерживают EC2, тогда как модули для Elastic Beanstalk обычно являются модулями сторонних разработчиков, размещенными на GitHub. Ничего плохого тут, естественно, нет, но для простоты я пока ограничусь основными модулями. Сайты у моего клиента были довольно простыми, так что EC2 изначально был разумным выбором. Сначала я собираюсь выстроить основу, а затем обратиться к автоматическому масштабированию и т.д.

Вернемся к моему репозиторию Git. Проект *Ansible* содержит руководство по наилучшим практикам (https://docs.ansible.com/playbooks_best_practices.html) с эффективной структурой каталогов для установки *Ansible*. Конечно, благодаря гибкости *Ansible* можно воспользоваться и собственной схемой каталогов, но я увидел, как эффективна эта схема в работе, и взглянуть на эту страницу в любом случае стоит. Я начал заполнять свою систему, создав несколько основных каталогов:

```
$ mkdir roles keys group_vars host_vars tools
```

Чтобы *Ansible* мог работать с AWS и скриптом `ec2.py`, надо определить несколько переменных окружения для своего сеанса. Для простоты я создал файл `env.sh` в каталоге `tools` и добавил в него следующие строки:

```
export AWS_ACCESS_KEY_ID='<my amazon key>'
export AWS_SECRET_ACCESS_KEY='<my amazon secret key>'
export ANSIBLE_HOSTS=/etc/ansible/ec2.py
export EC2_INI_PATH=/etc/ansible/ec2.ini
```

EC2 Instance type	Software	EC2	Total
t2.micro	\$0.00/hr	\$0.014/hr	\$0.014/hr
t2.small	\$0.00/hr	\$0.028/hr	\$0.028/hr
t2.medium	\$0.00/hr	\$0.056/hr	\$0.056/hr
m3.large	\$0.00/hr	\$3.10/hr	\$3.10/hr
m3.xlarge	\$0.00/hr	\$6.20/hr	\$6.20/hr
m3.2xlarge	\$0.00/hr	\$12.40/hr	\$12.40/hr
r3.2xlarge	\$0.00/hr	\$3.75/hr	\$3.75/hr
g2.2xlarge	\$0.00/hr	\$0.702/hr	\$0.702/hr
g2.xlarge	\$0.00/hr	\$1.404/hr	\$1.404/hr
m3.medium	\$0.00/hr	\$0.073/hr	\$0.073/hr
m3.large	\$0.00/hr	\$0.146/hr	\$0.146/hr
m3.xlarge	\$0.00/hr	\$0.292/hr	\$0.292/hr
m3.2xlarge	\$0.00/hr	\$0.585/hr	\$0.585/hr
q2.xlarge	\$0.00/hr	\$0.938/hr	\$0.938/hr
q2.2xlarge	\$0.00/hr	\$1.876/hr	\$1.876/hr
q2.xlarge	\$0.00/hr	\$3.751/hr	\$3.751/hr
q2.2xlarge	\$0.00/hr	\$7.502/hr	\$7.502/hr
c3.large	\$0.00/hr	\$0.12/hr	\$0.12/hr
c3.xlarge	\$0.00/hr	\$0.239/hr	\$0.239/hr
c3.2xlarge	\$0.00/hr	\$0.478/hr	\$0.478/hr
c3.4xlarge	\$0.00/hr	\$0.956/hr	\$0.956/hr

Экономика AWS

Одна из часто называемых причин отказа от облачной инфраструктуры связана с почасовой оплатой использования виртуализированного «железа» и программ. Беглый взгляд на модели ценообразования Amazon создаст у обычного читателя впечатление, что их разрабатывали те же люди, которые создали старую систему счетов за газ и электричество — настолько они темны для непосвященных. Работа сисадмина частично состоит в том, чтобы заниматься ежемесячными счетами Amazon или взаимодействовать с другими сотрудниками, которые

отслеживают эти счета. Тестовые системы, которые мы разберем в этой и в следующей статье, должны легко окупиться бесплатным 750-часовым использованием экземпляра `t2.micro` (но по окончании работы с ним уничтожайте все не нужные вам данные). Хотя экземпляры с оплатой за использование обычно характеризуются наибольшей гибкостью, они могут быть не дешевле других решений, пусть и за вычетом капитальных затрат на приобретение «железа» и оплаты счетов за электричество. Amazon предлагает и другие рыночные модели, которые тоже стоит исследовать.

» **Экземпляры с плавающей оплатой** Это ресурсы, с которыми пользователь сам решает, сколько ему будет по карману заплатить. С приливами и отливами цена колеблется и может опуститься до заданного пользователем порога, и в этот момент ресурсы становятся доступны пользователю. Эта модель удобна для задач, у которых нет дедлайнов или которые можно прерывать.

» **Зарезервированные экземпляры** Это именно то, на что оно похоже — ресурсы, хранимые заданный срок (от года до трех). Amazon дает на них существенные скидки.

› Так как этот блог скорее всего будет читать только моя мама, бесплатной подписки Amazon должно быть более чем достаточно.



Два параметра ключей — это, очевидно (ну, надеюсь, что очевидно), те два значения, которые я сгенерировал сразу после первого входа в консоль EC2. После добавления их в скрипт я смогу добавить их в свое окружение, просто выполнив команду `./tools/env.sh`. Запуск встроенной команды `env` подтверждает, что ключи стали знакомы моему сеансу. Затем я копирую закрытый ключ SSH, сгенерированный для меня Amazon, в каталог `keys`, переименовав его в `lxkeys` и выполнив команду `chmod 400` для файла, чтобы про- честь его смог только я.

(Но тут я столкнулся с ошибкой при создании тестовой системы. Мне надо было создать в каталоге проекта файл `inventory`. В этом файле должно быть всего две следующие строки — без него у меня появились бы ошибки при последующих запусках *Ansible*.)

```
[base]
localhost ansible_connection=local
```

Теперь решим, что мы будем запускать. В Amazon есть огромный рынок программных пакетов для использования в AWS (<https://aws.amazon.com/marketplace>), от известных дистрибутивов Linux до более смутных промышленных систем. Я набрал в поисковой строке "Wordpress" и нашел несколько вариантов. Я решил попробовать образ "Wordpress powered by Bitnami HVM" (так как у него был хороший рейтинг и он был бесплатен для использования). Выбрав его и согласившись на «покупку» (процесс напоминает покупку книг для Kindle), я взглянул на параметры для запуска вручную (на рисунке на стр. 64). В них мне нужен идентификатор AMI, уникальный для этой программы в конкретном дата-центре Amazon, где я собираюсь запускать свою программу (в моем случае это дата-центр в Ирландии, также известный как eu-west-1). Он понадобится для использования в моем скрипте *Ansible*. Через несколько минут Amazon напишет мне, что мой образ готов к использованию, и я готов к следующему шагу.

Включим и развернем... блог

Теперь мне надо записать идентификатор своего виртуально- го частного облака. По сути это выделенный мне фрагмент сети Amazon. Его можно увидеть справа на панели управления EC2, но у него есть и собственное выпадающее меню (чтобы его найти, выберите "Services"). Исследовав это, я также смог выделить под- сети, доступные мне в этом виртуальном частном облаке (по умол- чанию мне выделили три подсети). Я выбрал одну из них для фай- ла, который потом создал для своего репозитория *Ansible*, назвав его `site.yml`.

```
---
# LXF AWS example
- hosts: localhost
  connection: local
  gather_facts: False
  vars:
```

```
region: eu-west-1
tasks:
- name: Provision an instance
ec2:
  key_name: lxkeys
  instance_type: t2.micro
  image: "ami-51345f26"
  wait: true
  count: 1
  region: eu-west-1
  vpc_subnet_id: subnet-baf628df
  assign_public_ip: yes
```

В этом очень простом файле содержится вся информация, не- обходимая AWS для запуска экземпляра EC2. В нем я выбрал дата- центр, объем экземпляра (t2.micro), образ, который хочу загрузить (взят со страницы образов компьютеров Amazon), а также указал, что мне нужен публичный IP-адрес. После этого, убедившись, что переменные окружения заданы правильно, я выполнил команду:

```
$ ansible-playbook site.yml --private-key=keys/lxkeys -i inventory
```

Запуск *Ansible* завершается удачно:

```
localhost : ok=1 changed=1 unreachable=0 failed=0
```

Но что произошло на самом деле? Я воспользовался скрип- том `ec2` для вывода списка экземпляров, которые теперь есть на облаке AWS.

```
$ /etc/ansible/ec2.py --list
```

Эта команда вывела массу данных, включая IP-адрес экземп- ляра, который теперь запустился, его публичное DNS-имя, тип и дру- гую разнообразную информацию:

```
"ec2": [
  "54.154.141.142",
  ],
```

Но если открыть публичное выделенное DNS-имя, браузер только и сообщает, что время ожидания истекло. Еще одно изме- нение, которое мне надо сделать, вызвано тем, что по умолчанию AWS блокирует весь внешний трафик. Воспользовавшись кон- солью EC2 и выбрав `Security Groups` [Группы безопасности] сле- ва, я вижу свою группу по умолчанию. Нажатие кнопки `Action` [Действие] в верхней части экрана позволяет мне изменить прави- ла для входящего трафика [`edit inbound rules`] и разрешить HTTP- трафик из любого входящего источника. После этого снова от- кройте публичное DNS-имя, и вы должны увидеть чистый блог *WordPress*, готовый к размещению записей.

В этом ретроспективном номере **LXF** закончить статью, думаю, следует тем, чтобы отметить, как легко сейчас развернуть рабо- чую систему Linux в любой точке мира, воспользовавшись парой строк текста и браузером. В следующем месяце мы попробуем из- бавиться от браузера совсем, и рассмотрим ряд продвинутых воз- можностей AWS. **LXF**

LINUX FORMAT

Подписывайтесь и читайте Linux Format на iPad или iPhone!

Доступно
в AppStore



А если у вас Android, подпишитесь на электронную версию журнала Linux Format на сайте shop.linuxformat.ru!

Наши эксперты помогут вам с любым приложением Linux!



ЕВГЕНИЙ БАЛДИН
Подтвердивший
свою квалификацию
физик.

Десять лет

— Ты не тот, кого я знала
десять лет назад.
— Дело не во времени, детка.
Дело в пробеге.
*Индиана Джонс. В поисках
утраченного ковчега*

Не, честно, почти десять лет
назад я начал писать в *Linux
Format*. Не в самый первый
номер, так как я не знал, что он го-
товится к выходу, но в четвертый!
И с тех пор, ну почти, так как пятый
номер я пропустил, дорогие мои чи-
татели, я в каждом номере журнала
оставляю следы своих мыслей. Хо-
рошо или плохо — оставляю решать
редактору. Я стараюсь.

И сегодня, как десять лет назад,
я по-прежнему убежден, что свобод-
ное программное обеспечение в об-
щем и GNU/Linux в частности — это
реально удобно и полезно всем и ка-
ждому. В этом факте ничего за де-
сять лет не поменялось. Эта область
человеческой деятельности активно
развивается и загибаться отнюдь
не собирается, ибо бессмертие сво-
бодных программ или как минимум
идей, воплощенных в них, заключено
в их доступном всем и каждому коде.

Очень хорошо, что тексты, из ко-
торых почти целиком и полностью
состоит журнал, помогают этой сво-
боде, и я реально благодарен тем
людям, которые почти незаметно для
читателя стоят за выпусками *Linux
Format*. Уже десять лет...

PS Было объявлено о выпуске *exwm*
или *Emacs (!) X Window Manager*.

Я думаю, что это достойный ответ
evilwm, позволяющего управлять
окнами в стиле *vi*.

E.m.Baldin@inp.nsk.su

В этом месяце вы научитесь...



Работать в Debian 68

Самодовольство убунтоидов раздражало
Джонни Бидвелла, и им в пик он перешел
на солидный, серьезный и очень стабильный
дистрибутив.



Защищать девайсы 72

Ничто так не подвержено кражам, как ноут-
буки и смартфоны — ох уж эта их мобильная
природа, вздыхает **Нейл Ботвик** и разрабаты-
вает меры предосторожности.



Латать ядро 74

Покушаемся на священный запрет! С поднач-
ки **Джонни Бидвелла** попробуем вносить ис-
правления, не останавливая работу системы.



Внедрять IPv6 78

Вот раздадут еще каких-то полсотни милли-
онов адресов, тут IPv4 и конец... **Джонни Бид-
велл** рекомендует озаботиться этим заранее.

АКАДЕМИЯ КОДИНГА



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

Процесс пошел 82

Д-р **Крис Браун** принимается за процессы и систем-
ные вызовы, а также перенаправление данных в кана-
лы. Сам создал процесс — сам его и убил.

Составьте слово 86

Ради праздника, **Пол Хадсон** написал игру в угадыва-
ние слов. Освойте под его чутким руководством ос-
новы языка программирования PHP.

Debian 8: Пора стартовать!

Джонни Бидвелл учит, как почувствовать превосходство над друзьями-убунтоидами с помощью Debian — почтеннейшего из всех Linux'ов.



Наш эксперт

Джонни Бидвелл когда-то был достойным и уважаемым членом общества, и перед ним открывались блестящие перспективы, пока он не подсел на Linux.



```
jonni@debian: ~
File Edit View Search Terminal Help
GNU nano 2.2.6 File: /etc/apt/sources.list
#
# deb cdrom:[Debian GNU/Linux 8.0.0 _Jessie_ - Official amd64 CD]
# deb cdrom:[Debian GNU/Linux 8.0.0 _Jessie_ - Official amd64 CD]
deb http://ftp.uk.debian.org/debian/ jessie main
deb-src http://ftp.uk.debian.org/debian/ jessie main
deb http://security.debian.org/ jessie/updates main
deb-src http://security.debian.org/ jessie/updates main
# jessie-updates, previously known as 'volatile'
deb http://ftp.uk.debian.org/debian/ jessie-updates main
deb-src http://ftp.uk.debian.org/debian/ jessie-updates main
```

» Для жителей доброй старой Англии (она же — Великобритания) обычный файл `sources.list` выглядит примерно так.

программа установки создана с упором на максимальную совместимость, и если ваше оборудование не является совершеннейшей экзотикой, вам не придется использовать текстовый режим. Если вы устанавливаете Debian вместе с Windows или любой другой операционной системой, опция разбивки диска на разделы вручную позволит вам установить нужный размер любого раздела. Возможно, вы решите разбить диск на разделы заранее, особенно если вам нужна, или желаемая, или у вас уже налицо сложная структура разделов.

Разбиение на разделы

Выделять раздел Windows из Linux (например, при посредстве *GParted*) — идея не из лучших, поскольку загрузчик [bootloader] Windows относится к подобным вещам нервнически и обычно отказывается грузиться впоследствии. Поэтому либо уж не двигайте разделы Windows, либо делайте это только из Windows. Также, если вы изменяете размер подобных вещей в *GParted*, то убедитесь, что опция `Align to` [Выравнивание] настроена на `None` [Нет], поскольку в противном случае раздел будет перемещен в ближайший цилиндр или MiB, что приведет к плачевным последствиям. Благоразумно будет проверить, грузится ли Windows после внесения изменений в разделы, еще до того, как вы усугубите ситуацию другими ОС. Стоит отметить, что Debian 8.0 притом является исключительно подходящей ОС для VPS [виртуальный приватный сервер]. Любой приличный провайдер уже предоставляет образы Debian 8, и те, кому не нужны особо мощные ресурсы, могут получить его всего за \$5/месяц. Естественно, вам незачем устанавливать на подобную машину рабочий стол, но как только провайдер

В прошлом месяце мы рассказывали о Debian 8 [см. Обзор, стр. 14 LXF199], самом последнем стабильном релизе одного из самых достопочтенных дистрибутивов в Солнечной системе. Многие из наших читателей хорошо знакомы с Debian, а многие, без сомнения, жаждут обучиться. Поэтому далее вам предлагается руководство по выполнению повседневных задач в самом маститом из Linux'ов. Главным образом мы займемся основами, но коснемся и проблем, с которыми могут столкнуться патриархи Debian, в частности, при переходе на *Systemd*. Пользователей, пробующих Debian впервые, иногда несколько разочаровывает то, что пакеты сознательно привязаны к стабильным версиям релизов. Некоторым подавая новинки, и мы увидим, что есть возможности внедрить их в вашу инсталляцию Debian. Однако это не всегда хорошая идея: старый — не значит бесполезный, и потому вместе с более современным дистрибутивом стоит держать по соседству стабильную версию Debian, пусть даже в качестве аварийного варианта, на случай, если вы нарветесь на ошибку.

Если вы сумеете отключить механизм безопасной загрузки [secure boot], то процесс установки достаточно прост. Графическая

Скорая помощь

Если в вашей домашней сети работает несколько машин с Debian, то вы можете сэкономить трафик с помощью программы *Apt-Cacher-NG* из репозитория. Она настроит одну машину в качестве кэширующего прокси, который будет работать локальным репозиторием.

Освойтесь с программой

Есть ряд случаев, когда использовать версию пакета поновее, чем предлагаемая репозиториями Debian, более желательно: например, когда у вас браузер или клиент электронной почты, которые в каждом релизе предлагают интересные новые функции. Обратные порты [backports] — это переработки пакетов из менее стабильных каналов. Например, Mozilla поддерживает архив пакетов *IceWeasel*, которые можно включить следующим образом. От имени root создайте файл `/etc/apt/sources.list.d/mozilla.list` со следующей строкой:

```
deb http://mozilla.debian.net/ jessie-backports
iceweasel-release
```

Вы можете заменить `release` на `beta` или `aurora`, в зависимости от того, насколько близко к передовой вы хотите очутиться. Этот архив подписан, так

что вам придется добавить архивный ключ Mozilla к связке *APT*. Для этого:

```
$ wget http://mozilla.debian.net/archive.asc
$ sudo apt-key add archive.asc
```

Наиболее параноидальные среди вас захотят удостовериться скачанный ключ. У него есть отпечаток `06C4AE2A`, так что вперед, удостоверьтесь. Добавив ключ, вы можете обновить архив пакетов (на этой стадии вы увидите несколько ссылок на <http://mozilla.debian.net>) и затем обновить *IceWeasel*:

```
$ sudo apt-get update
$ sudo apt-get install iceweasel
```

Помните, что версия *IceWeasel*, поставляемая в Debian, является частью Mozilla Extended Service Release, так что бреши в системе безопасности браузера латаются незамедлительно.

Использование более новой версии из backports не обязательно обеспечит дополнительную безопасность, но зато увеличит номера версий и, возможно, число ошибок.

Имеется также официальный репозиторий Debian backports, так что вы можете устанавливать пакеты, перекомпилированные из Testing и Unstable. Вы можете добавлять их с помощью следующей строки в `/etc/apt/sources.list`:

```
deb http://http.debian.net/debian jessiebackports main
```

Пакеты из этого репозитория отмечены как низкоприоритетные (100), чтобы они не мешали пакетам из stable. Чтобы их установить, явно укажите репозиторий backports, с помощью

```
$ sudo apt-get install -t jessie-backports install
"package"
```

VPS ее развернет, вы сможете войти в нее через SSH и применить мудрые советы нашего урока.

То, что Debian Stable уклоняется от самых последних версий программ, не означает, что он вообще боится обновлений пакетов. Обновления происходят с той же частотой, что и в любом другом дистрибутиве; единственная разница в том, что обновления обычно не включают особо резких изменений версий — только отладки и обновления безопасности. После запуска 8.0 появился технический релиз 8.1, который содержит ряд подобных отладок. Однако совершенно незачем производить переустановку или еще что-либо столь же радикальное: все эти отладки имеются в репозиториях. Обновлять пакеты жизненно необходимо, но и очень просто. Список репозитория Debian хранится в файле `/etc/apt/sources.list`. Вы можете добавить репозитории в этот файл или в новый файл в директории `/etc/apt/sources.list.d`. Если вы делали установку с CD, то ваш файл `sources.list` будет содержать строку, начинающуюся с `deb cdrom:`. Это означает, что менеджер пакетов *APT* в качестве источника может использовать CD. Если у вас медленное соединение с Интернетом, после установки вы можете выгадать на этом некоторое время, но когда вы синхронизируетесь с репозиториями, большинство этих пакетов окажутся устаревшими. Так что прокомментируйте строку `cdrom`.

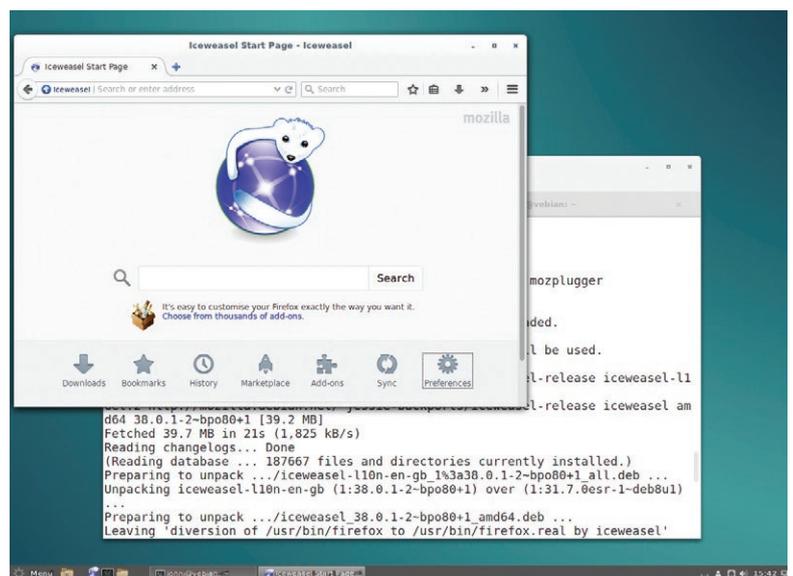
Поддержание актуальности

Каждая строка файла `sources.list` указывает определенный репозиторий. Строка начинается с `deb` или `deb-src`, что определяет, размещаются ли там бинарники или исходные пакеты. Второе поле — URL, третье — дистрибутив, которым будет Jessie, если вы просто сделали новую установку. Здесь также можно указать ветвь — стабильную или тестовую. Текущий стабильный релиз — Jessie, но через пару лет стабильным станет Stretch (ныне тестовый). Замена имени `jessie` на имя канала, например, `stable`, через `sources.list` не возымеет мгновенного действия, но когда Stretch станет стабильным, вы получите обновление автоматически. Если оставить все как есть, то вы будете отслеживать релизы Jessie до конца его жизни (обычно — через пять лет после выхода релиза). Остальная часть строки репозитория указывает на компоненты, используемые в репозитории. В процессе установки выбирается только основной компонент, `main`. К репозиториям можно добавить `non-free`, чтобы включить проприетарные программы (такие, как плагин Flash — но не используйте его!) — или проприетарные драйверы).

Есть также полупочтенные компоненты, принадлежащие `contrib`. Эти пакеты сами по себе соответствуют правилам Debian Free Software Guidelines (DFSG), но у них есть не соответствующие этим указаниям зависимости. Вы увидите, что некоторые строки в `sources.list` ссылаются на дистрибутив `jessie-updates`. Сюда включены обновления, которые стоит установить до следующего релиза (когда они станут частью настоящего Jessie). Можно еще добавить имя дистрибутива `proposed-updates`, чтобы получить обновления, запланированные для очередного знакового релиза, но помните, что они будут не полностью протестированы.

Периодически, и когда бы вы ни обновляли `/etc/apt/sources.list`, вам надо будет запускать (от имени root) `# apt-get update` для обновления локального списка доступных пакетов. Вам следует использовать команду `su` (вместе с паролем root, который вы указали при установке) для получения доступа root изначально, поскольку Debian не настраивает для вас `sudo`. Если вы привыкли к Ubuntu, то `sudo`, вероятно, очень близкая вашему сердцу команда. Настроить ее в Debian несложно — просто запустите от имени root

»



» Логотип *Firefox* нельзя перераспространять, и Debian представляет браузер под именем *IceWeasel*. С помощью нашего замечательного урока мы обновили его до версии 38.

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

редактирование файла с помощью *nano* (можно обойтись редактором по умолчанию, *vi*, коли вам так хочется, но он не слишком дружелюбен к новичкам):

```
# EDITOR=nano visudo
```

Программа *visudo* редактирует и затем проверяет синтаксис файла */etc/sudoers*, чтобы невольные опечатки не лишили доступа к *sudo* сразу всех. Добавьте следующую строку, заменив *user* на свое имя пользователя:

```
user ALL=(ALL:ALL) ALL
```

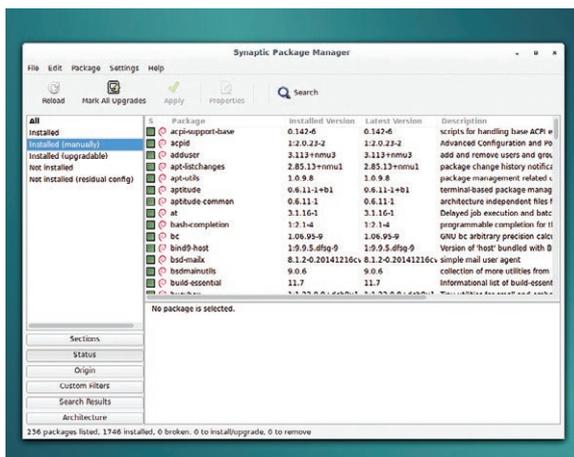
Теперь выйдите с помощью *Ctrl+c* и нажмите *u* для сохранения. Сохранится временный файл для проверки в *visudo*, а если он будет одобрен, то перепишет */etc/sudoers*. Отныне вам не понадобится команда *su*, и вы сможете обновлять свою систему точно так же, как это делал бы пользователь *Ubuntu*:

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

Иногда *APT* будет сообщать, что некоторые пакеты не обновляются; причина — некий конфликт зависимостей. Это может произойти, например, если вы отслеживаете стабильный канал в */etc/apt/sources.list* в момент выхода нового релиза. Решением будет использовать умную программу решения проблем зависимостей, вызываемую командой

```
$ sudo apt-get dist-upgrade
```

Можно воспользоваться *apt-cache* для поиска в локальном кэше имеющихся пакетов. Например, вот как найти в описаниях пакетов соответствия 'chromium':



➤ *Synaptic* очень удобен для просмотра установленных пакетов или поиска новых. Опытные пользователи, однако, игнорируют подобную графическую ерунду.

```
$ apt-cache search chromium
```

Однако управление пакетами Debian способно на большее. Пакет *APT* работает как интерфейс для инструмента *dpkg*, разделяя все это сложное разрешение зависимостей и что-там-у-вас-есть. Информация о состоянии установленных или иных пакетов, в данном случае, браузера *IceWeasel*, просматривается с помощью

```
$ dpkg -l iceweasel
```

Последняя строка ее вывода будет выглядеть примерно так:

```
ii iceweasel 31.7.0esr-1~d amd64 Web browser...
```

если только вы не изгнали *IceWeasel* (часть установки по умолчанию) из своей системы. Код из двух букв слева относится к состоянию пакетов — первая *i* означает «желательно, чтобы пакет был установлен», а вторая *i* сообщает нам, каково его настоящее состояние. Если пакет находится в нестабильном состоянии, то *dpkg* покажет третью букву, *R*; она означает, что требуется переустановка. Иногда исправления можно сделать автоматически: команда

```
$ sudo apt-get install -f
```

попытается сделать ваше древо пакетов стабильным.

Удаление пакетов

Пакеты можно удалять (*apt-get remove*) или вычищать (*apt-get purge*); последнее означает, что все файлы настройки тоже будут удалены. Пакеты, которые были вычищены полностью, имеют статус, сообщаемый *dpkg -l* как *rr*, а удаленные пакеты, от которых уцелели файлы настройки, получают статус *rc*. По мере замены пакетов некоторые зависимости осиротеют — следовательно, станут не нужны. *APT* предложит вам использовать

```
$ sudo apt-get autoremove
```

чтобы их убрать, и вам следует согласиться. Также, директория кэша пакетов, */var/cache/apt*, со временем будет переполняться. Неплохой привычкой будет периодически чистить ее с помощью

```
$ sudo apt-get autoclean
```

Помимо низкоуровневой команды *dpkg*, есть высокоуровневая программа *Aptitude*. Хотя для повседневных задач установки и удаления вы можете использовать ее вместо *apt-get*, желательно всё же избегать их смешения, поскольку они сообщают о своих действиях в разных местах. Одна из самых привлекательных функций *Aptitude* — способность увидеть, почему установлен определенный пакет; например, мы возжелали узнать, с чего это понадобилось установить сервер *Wayland*:

```
$ aptitude why xwayland
```

Дело в том, что у нас рабочий стол *Cinnamon*, и он зависит от пакета *gnome-session-bin* (*Cinnamon* по-прежнему использует множество компонентов *Gnome* за сценой), а там-то и кроется зависимость

Отход от Systemd

Если только вы не свалились с луны, то знаете, что некоторым не особо нравится переход почти всех основных дистрибутивов Linux на *Systemd*. Помимо страха, неуверенности и сомнений, есть и обоснованная критика *Systemd*, и кое-кто был бы рад от него избавиться.

Однако помните, что хотя *Systemd* сейчас можно удалить и заменить старым добрым *SysVinit*, после релиза *Jessie* поддерживать старую систему никто не планирует. Помните также, что от *Systemd* зависит *Gnome* (и, следовательно, *Cinnamon*), и вам, возможно, придется решать, какое из ваших желаний

сильнее: избавиться от *Systemd* или наслаждаться этими рабочими столами.

Способ удалить *Systemd* наиболее чисто — использовать в начале процесса установки строку *preseed*. Просто нажмите *Tab* в опции меню *Graphical Install* и добавьте следующее в команду *boot*:

```
preseed/late_command="in-target apt-get install -y sysvinit-core"
```

Если вы уже выполнили установку, процесс удаления тоже не самый сложный. Сначала установите все пакеты *SysV*:

```
$ sudo apt-get install sysvinit-core sysvinit-sysvinit-utils
```

Теперь перезагрузитесь, чтобы *Systemd* больше не было в памяти, и затем вычистите его, возможно, во время воскресения шалфея:

```
$ sudo apt-get remove --purge --auto-remove Systemd
```

Вы можете засекать через *apt*, что *Systemd* не пытается проскользнуть обратно. От имени *root* отредактируйте файл */etc/apt/preferences.d/system*, придав ему следующий контент:

```
Package: systemd
Pin: origin
```

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

Wayland. Рабочий стол Wayland мы в Debian вживе увидим не вдруг, да, если на то пошлёт, и в любом другом дистрибутиве.

Если у вас возникли проблемы при установке пакета, то в *Aptitude* есть и команда `why-not`. Там (к счастью) нет эквивалента *Ubuntu Software Center*, но *Synaptic* позаботится обо всех требованиях по графическому управлению пакетами.

Стандартная установка Debian создана так, чтобы включить всего как можно больше. Иными словами, есть некоторые части, без которых вы, вероятно, можете обойтись, в зависимости от того, как вы планируете использовать свой компьютер. Ненужные пакеты будут занимать место на диске и замедлять запуск, однако не спешите все удалять. Например, мы пришли к выводу, что нам не нужен почтовый сервер *exim*; но некоторые демоны используют его для отправки предупреждений на локальный почтовый буфер. Тогда безопаснее будет деактивировать его:

```
$ sudo systemctl disable exim4
```

Урезаем сервисы startup

Подобным же образом мы прихлопнули сервисы *modemmanager* и *speech-dispatcher*. Это высвободило из времени загрузки порядка пяти секунд (что показывает команда `systemd-analyze`). Debian только что внедрил *Systemd* в качестве своей системы инициализации; если вы намерены его отключить, следуйте нашим инструкциям (см. врезку *Отход от Systemd*, внизу стр. 70). Но в обозримом будущем проект Debian будет поддерживать его в режиме совместимости. Это означает, что старые скрипты *SysV* будут по-прежнему работать, и *Systemd* даже будет поддерживать ссылки в `/etc/rc?.d/`. То есть если мы включим SSH-сервер с помощью

```
$ sudo systemctl enable ssh
```

— как по волшебству, появится ссылка, например:

```
$ ls -l /etc/rc3.d/S02ssh
```

```
lrwxrwxrwx 1 root root ... S02ssh -> ../init.d/ssh
```

Если вы отключите сервис, то эта стартовая ссылка, как и другие стартовые ссылки в других папках уровня выполнения [runlevel], исчезнет.

Скрипты *Systemd*, пожалуй, чуть опрятнее, чем их эквиваленты у *SysV*. Они (что тревожит) отформатированы, как файлы Windows INI. Вы найдете файлы *Systemd* в директории `/lib/systemd/system`. Поскольку *Systemd* довольно умно работает с зависимостями, файлы *Systemd* нередко намного проще скриптов *SysV*. (Сравните, например, файл сервиса SSH с его коллегой `/etc/init.d/ssh` — файл *Systemd* содержит 15 строк против 154). Когда вы включаете такие сервисы с помощью `systemctl`, ссылки создаются в директории `/etc/systemd/system`.

Проприетарные драйверы

Хотя мы всячески приветствуем использование свободных драйверов и драйверов с открытым кодом везде, где возможно, все же остается ряд видеокарт (в частности, Nvidia), которые намного лучше работают с проприетарными драйверами. Первый шаг — добавить к репозиторию *jessie* компоненты `contrib` и `non-free` (первая строка содержимого `/etc/apt/sources.list`). Далее надо будет добавить правильную версию пакета *linux-headers*. Если вы недавно обновили *apt-get* и с тех пор не перезагружались, сделать это стоит, поскольку драйверы должны компилироваться относительно ядра, которое будет работать после следующей перезагрузки, а не относительно работающего в данный момент. Если в обновлении не упоминались *linuximage* или *linux-headers*, можете спокойно продолжать без перезагрузки.

Найдите нужные заголовочные пакеты с помощью

```
$ sudo apt-get install linux-headers-$(uname -r)
```

Jessie поддерживает две ветви проприетарных драйверов Nvidia, 340 и 304, и только серию 14.9 для драйвера Catalyst. Если вам они кажутся архаичными, что вполне вероятно, тогда стоит использовать пакеты с сайта производителя; однако это отнюдь



► Рабочий стол Cinnamon доступен из программы установки — всего лишь в версии 2.2, но смотрится и ощущается хорошо.

не соответствует образу жизни Debian, так что будьте осторожны. В порядке альтернативы, можете подумать о применении пакетов из канала *Experimental*. Однако это процесс необратимый, и в основном лежит за границами нашего урока.

Мы поговорим о возможности выбора из менее стабильных каналов (данный процесс именуется ‘*apt pinning*’) позднее. А сейчас предположим, что вас вполне устраивают имеющиеся в Debian пакеты драйверов.

Для карт Nvidia и драйвера релиза 340 (это последний релиз, где поддерживаются процессоры GeForce 8 и 9) установите пакеты следующим образом:

```
$ sudo apt-get install nvidia-kernel-dkms
```

Dynamic Kernel Module Support (динамическая поддержка модулей ядра, DKMS) — система автоматической recompilации модулей при любом обновлении ядра. Без нее те, кто не следит постоянно за обновлениями пакетов и не переустанавливает эти пакеты вручную, будут постоянно испытывать проблемы с графикой. Если вас больше устроит драйвер более старой серии 304 (который поддерживает по большей части те же карты, что и 340), используйте вместо этого пакет *nvidia-legacy-304xx-kernel-dkms*. *X.org* не определит драйвер *nvidia*, и вам придется создать минимальный файл *xorg.conf*, запустив

```
$ sudo nano /etc/X11/xorg.conf.d/20-nvidia
```

для редактирования нового файла, что является более элегантным вариантом. Придайте ему следующий контент:

```
Section "Device"
```

```
Identifier "My GPU"
```

```
Driver "nvidia"
```

```
EndSection
```

Теперь перезапуститесь, и вы должны ощутить мороз по коже, напоминающий вам о том, что ваше ядро отныне заражено чем-то левым.

Помните, что драйвер Catalyst несовместим с рабочим столом Gnome, так что тут надо оставить свободный драйвер. В ином случае, для установки драйвера Catalyst используйте

```
$ sudo apt-get install fglrx-driver
```

что автоматически найдет всю мелочевку DKMS. Затем от имени `root` создайте файл `/etc/X11/xorg.conf.d/20-fglrx.conf` со следующим контентом:

```
Section "Device"
```

```
Identifier "My GPU"
```

```
Driver "fglrx"
```

```
EndSection
```

На чем мы и заканчиваем наш урок по Debian. Не забудьте взглянуть в *wiki Debian* (<https://wiki.debian.org>), которое станет путеводителем в мире ваших приключений в Debian. LXF



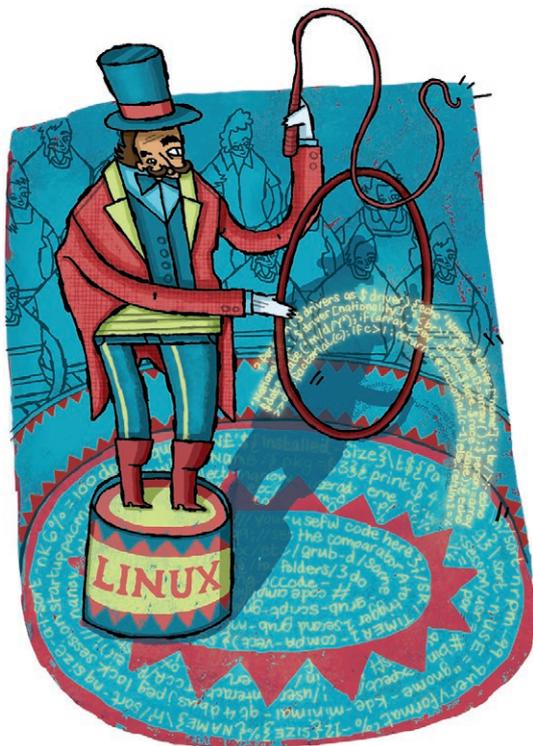
Защита девайсов: Важные советы

Если вам не уберечь свой ноутбук или смартфон от воров, **Нейл Ботвик** покажет вам, как обезопасить их содержание.



Наш эксперт

У **Нейла Ботвика** колоссальный опыт загрузки, ведь у него в каждой комнате по компьютеру, а вот с перезагрузкой похуже, поскольку он перешел на Linux.



Мы часто говорим о безопасности, имея в виду управление паролями или шифрование данных, однако портативные устройства чреваты еще одной потенциальной проблемой — это угроза их похищения. Легко сказать, что вы должны «быть осторожны», чтобы ваши портативные устройства не стырили, но что можно сделать для минимизации ущерба, если ваш ноутбук или смартфон всё же украден? Мы рассмотрим некоторые способы обезопасить ваши данные, если они

попали не в те руки, не добавляя при этом непрактично большое количество уровней безопасности, которые вы сами просто поленились использовать.

Сильные пароли — всегда хорошо, а особенно в случае мобильных устройств. Если кто-то украл ваш ноутбук, он попытается взломать ваш пароль. Такие программы, как *John the Ripper*, могут взломать пароли из вашей системы, при условии, что у злоумышленника есть доступ к вашим файлам паролей и достаточно времени. С его временем поделаться ничего нельзя, но можно сделать важные файлы нечитаемыми. Требования наличия прав root для доступа к ним будет мало: компьютер можно загрузить со спасательного CD и прочитать от имени root всё что угодно, поэтому приоритетом становится шифрование вашего жесткого диска. Мы рассказывали о шифровании директории **home** [см. Учебники, стр. 76 **LXF196**], но шифрование всего диска означает переустановку или резервное копирование всего и шифрование диска с помощью *dm-crypt* [см. Учебники, стр. 74 **LXF186**]. Это большая работа, но вам стоит рассмотреть возможность шифрования всего диска при установке дистрибутива. Вы также должны установить пароль на BIOS, чтобы никто не смог изменить настройки, разрешив загрузку с другого носителя.

Блокировка

Зашифрованный диск или директория **home** — только часть решения. Это не позволит никому использовать live CD для чтения вашего диска; но если ноутбук украли, когда он был включен, ваши данные будут доступны для всех. Для этого есть простое решение: удобнее всего переносить ноутбук с закрытой крышкой — итак, нельзя ли заблокировать компьютер, если крышка закрыта? Ответ — да: система ACPI реагирует на разные события, в том числе на открытие и закрытие крышки. ACPI часто используется для отключения питания, чтобы продлить заряд, но можно заодно заставить ее заблокировать экран. Подробности отличаются для разных рабочих столов; в KDE вы настраиваете действие «закрытие крышки» в разделе настроек **Power Management**, пользователи Gnome должны заглянуть в **Privacy**, а для Unity эти настройки находятся в **Security & Privacy**. Можно также настроить это и независимо от среды рабочего стола. Пользователи *Systemd* могут отредактировать **/etc/systemd/logind.conf**, задав **HandleLidSwitch** как **'lock'**. Альтернатива — разрешить ACPI работать с этим напрямую, установив *xlock* через свой менеджер пакетов, и затем создать файл под названием **/etc/acpi/events/lid**, содержащий

```
event=button /lid.*close
action=sudo -u youruser /usr/bin/xlock -display :0
```

Это запустит *xlock* для блокировки вашего дисплея при закрытой крышке. Чтобы его разблокировать, вам понадобится пароль. Мы использовали *sudo* для запуска его от имени *youruser*, потому что демон ACPI запускает его от имени *root*, а *xlock* требует пароль пользователя, который его запустил. После создания

➤ Прежде чем применять устройство как триггер для *BlueProximity*, придется спарить его со своим компьютером, используя программу настроек рабочего стола.

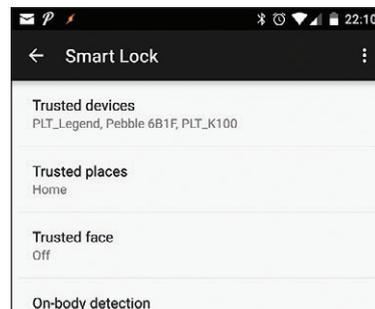


Опции Android

В основном мы рассмотрели ноутбуки, на которых используется рабочий стол Linux, однако столь же легко лишиться мобильного устройства на Android. Защита устройства с помощью PIN или комбинации для разблокировки намного надежнее, однако куда менее удобна. К счастью, редакция Android Lollipop (5.x) имеет функцию под названием Smart Lock, которая находится в настройках Security и позволяет обойти блокировку при выполнении определенных условий. На данный момент она работает

по местонахождению GPS или в присутствии устройства Bluetooth. Например, вы можете отключить блокировку, когда вы дома или в своей машине.

Определить местонахождение и даже дистанционно стереть данные с утерянного смартфона можно из менеджера устройств Google на <https://www.google.co.uk/android/devicemanager>. Вы дополнительно обезопасите свой телефон, включив аутентификацию и настройку, требующую ввода PIN при его включении.



➤ Функция Smart Lock в Android Lollipop позволяет автоматически разблокировать свое устройство при определенных условиях.

или редактирования этого файла надо будет перезапустить сервис *acpid*. Большинство дистрибутивов отключают использование Ctrl+Alt+F1 ради выхода в виртуальную консоль, и вам следует это учесть, поскольку *xlock* блокирует только X-дисплей. Добавьте к `/etc/X11/xorg.conf` или файлу в `/etc/X11/xorg.conf.d` следующее:

```
Section "ServerFlags"
    Option "DontVTSwitch" "true"
EndSection
```

Блокировка по близости

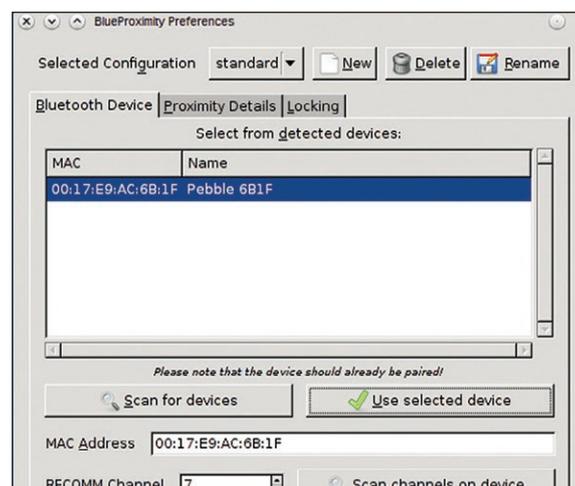
У описанного метода есть два недостатка: он предполагает, что вор закроет крышку, а обойти без этого не такая уж проблема; и притом вам надо вводить свой пароль при каждом ее открытии, а мы всегда советуем выбирать сильный пароль. Было бы удобно, если бы ноутбук работал, только когда им пользуетесь вы, но не для всех остальных. Мы здесь говорим не о сканерах отпечатков пальцев, а о том, что есть почти у всех: о смартфонах. *BlueProximity* (<http://BlueProximity.sourceforge.net>) проверяет спаренные устройства Bluetooth, и когда устройство удаляется на некое расстояние, оно блокирует компьютер; при возврате устройства в заданный диапазон компьютер снова разблокируется. Если ваш телефон всегда у вас в кармане или в руках, это работает очень хорошо. Еще лучше соединить ноутбук с умными часами.

Первый шаг — спарить ваше устройство Bluetooth с вашим компьютером, используя инструменты настройки вашего рабочего стола. Затем запускайте *BlueProximity*. При вызове его из программы запуск приложений с виду ничего не произойдет; надо осмотреть системный лоток, поскольку оно может запускаться в скрытом режиме. Начните с нажатия на кнопку Scan. Щелкните по своему устройству в списке и нажмите Use selected device; MAC-адрес устройства скопируется в поле ниже. Теперь задайте параметры обнаружения во вкладке Proximity Details. Настройки расстояния — это, по сути, измерения уровня сигнала: чем выше это число, тем слабее сигнал и тем больше расстояние. Длительность контролирует, как долго ваше устройство должно находиться далее этого расстояния до того, как начнет действовать *BlueProximity*. Желая указать определенное расстояние, нажмите кнопку Reset Max/Min, переместите свое устройство на расстояние, на котором вы хотите начать блокировку, подождите несколько секунд, затем вернитесь, и вы увидите максимальное расстояние, зафиксированное внизу дисплея. В большинстве случаев настройки по умолчанию будут примерно те же; неплохо также задать расстоянию блокировки большую величину, чем расстоянию разблокировки, иначе устройство начнет судорожно блокироваться и разблокироваться, когда вы будете около одинаковой настройки. А при слишком малой длительности могут возникнуть неожиданные события, потому что уровень сигнала Bluetooth колеблется.

Третья вкладка настраивает команды, которые запускаются при блокировке и разблокировке. По умолчанию запускается *gnome-screensaver*; щелкните по кнопке со стрелкой, чтобы вместо него использовать *xscreensaver*, или введите собственную команду. *BlueProximity* не будет запускать команду блокировки, пока ее окно открыто, но изменит цвет ключа на своем значке, чтобы показать, находитесь ли вы в пределах установленного расстояния. Значения красного и зеленого цветов понятны; желтый означает, что вы находитесь в пределах диапазона блокировки, но вне диапазона разблокировки. Команда *proximity* постоянно запускается через заданный промежуток времени, когда ваше устройство в пределах заданного расстояния. По умолчанию *gnome-screensaver* не очищает экран, даже если вы не касаетесь клавиатуры или мыши. Возможно, придется поэкспериментировать с этими настройками, однако в конечном итоге вы получите компьютер, который работает только рядом с вашим же устройством Bluetooth.

Поиск потеряшки

Если произойдет худшее и ваше устройство украдут, вы можете кое-что сделать, чтобы отследить его или защитить свои данные. Установив на свои компьютеры и мобильные устройства клиент *Prey* (<http://preyproject.com>) и создав бесплатную учетную запись, вы сможете добраться до своего компьютера, даже если он у кого-то другого. Основной сервис бесплатный; имеются дополнительные уровни премиум-сервиса для дополнительных функций и устройств, но и бесплатная опция работает с тремя устройствами и обеспечивает хороший уровень надежности. Есть и другие сервисы, такие, как *Cerberus*, однако у *Prey* есть преимущества — открытый код и совместимость со всеми рабочими столами и мобильными операционными системами. LXF



➤ Первая вкладка окна *BlueProximity* используется для выбора искомого устройства.

Ядро: Латаем на ходу

Джонни Бидвелл покажет, как работает одна из самых интересных функций ядра 4.0 — live-заплатки, и как к ним подступиться.



Наш
эксперт

Джонни Бидвелл призадумался, не грозит ли редакции, где все пренебрегают запятыми, гнетущая пауза.



Применение обновлений программ в Linux редко требует перезагрузки всей системы. Как правило, достаточно перезапустить только те приложения или сервисы, которых оно касается, что в большинстве случаев не отражается на выполнении текущих операций. Сравните это, скажем, с Microsoft Windows, где принудительные перезагрузки (или назойливые напоминания о них) в порядке вещей. Есть, правда, одно существенное исключение: это ядро.

Установка нового ядра и модулей не мешает текущей работе системы Linux, но лишь потому, что применены они будут только после перезагрузки. Для пользователей настольных ПК это неудобство невелико, а вот операторам критически важных систем зачастую приходится планировать перезагрузки заранее. Ведь даже отдельный сервис невозможно перезапустить без учета пользователей и текущей нагрузки. По этой причине сисадмины, как правило, относятся к обновлениям ПО очень консервативно, дабы свести к минимуму время простоя и не злить пользователей.

Но теперь мы вступили в эпоху уязвимостей, надуманных акронимов и стильных логотипов, так что все чаще требования безопасности (по праву) диктуют нам, что работу сервиса все же порой следует прерывать. Хотя ошибки в ядре, по счастью, дело редкое (таких, чтобы удостоились логотипа, мы пока не встречали), тем более в давних ветках, но они все же случаются, и администраторы пострадавших систем обязаны найти способ всё исправить как можно скорее. Частенько это предполагает ночные бдения, перерод кофеина и отчаянные призывы к различным божествам.

И хорошо бы иметь некий механизм для применения заплаток [patch] безопасности к уже работающему ядру, чтобы устранять ошибки или уязвимости, не прерывая обслуживания.

Такая «чистка машины на ходу» стала для администраторов чем-то вроде Святого Грааля. Сама по себе технология существует с 2008 г., реализована она в системе под названием Ksplice. Эта система была (и остается) лицензирована GPLv2, но сверх того ее авторы, под эгидой коммерческой Ksplice Inc, предоставляют платные инструменты для упрощения установки на различные дистрибутивы. В 2011 г. Ksplice Inc была приобретена Oracle, и хотя сама Ksplice остается расширением ядра Open Source, ее новые владельцы поддерживают только Oracle Linux. Так что остальным свою мечту об изменении ядра в рабочем режиме придется воплощать как-то иначе.

Пару лет эта технология находилась в процессе становления, но в Linux 4.0 (вышел в середине апреля 2015 г.) она уже полноценно присутствует. Она известна как Live Kernel Patching (livepatch) и объединяет в себе две конкурирующие технологии: kGraft от openSUSE и kpatch от Red Hat. В обоих проектах трассировка выполняется посредством CONFIG_FTRACE, сообщающей ядру, какие функции используются в данный момент. Трассировка имеет ключевое значение для обеих этих систем, так как запущенные процессы необходимо обходить, а не останавливать. Исходные версии исправленных функций ядра должны быть по-прежнему доступны для тех операций, что были начаты до применения заплат, иначе несоответствий, сбоев и пролитых слез не оберешься. Хотя обе родительские технологии, как и их детище, используют для этого разные подходы, суть у них одна. Пусть мы хотим

Скорая
помощь



Если вам хватит ума собрать ядро 4.0 самостоятельно, включите опцию SAMPLE_LIVE_PATCHING. Она создает модуль `livepatch-sample.ko`, который при загрузке создает `live-patch /proc/cmdline`, чтобы известить вас о результатах обновления.

The Hurr Durr Archives

About Contact us FAQ Releases Signatures Site news 

Protocol	Location
HTTP	https://www.kernel.org/pub/
Git	https://git.kernel.org/
RSYNC	rsync://rsync.kernel.org/pub/

Latest Stable Kernel:
 **4.0**

mainline:	4.0	2015-04-12	[tar.xz]	[pgp]	[patch]	[browse]
stable:	3.19.3	2015-03-26	[tar.xz]	[pgp]	[patch]	[inc. patch] [view diff] [browse] [changelog]
longterm:	3.18.11	2015-04-04	[tar.xz]	[pgp]	[patch]	[inc. patch] [view diff] [browse] [changelog]
longterm:	3.14.37	2015-03-26	[tar.xz]	[pgp]	[patch]	[inc. patch] [view diff] [browse] [changelog]
longterm:	3.12.40	2015-04-09	[tar.xz]	[pgp]	[patch]	[inc. patch] [view diff] [browse] [changelog]
longterm:	3.10.73	2015-03-26	[tar.xz]	[pgp]	[patch]	[inc. patch] [view diff] [browse] [changelog]
longterm:	3.4.106	2015-02-02	[tar.xz]	[pgp]	[patch]	[inc. patch] [view diff] [browse] [changelog]
longterm:	3.2.68	2015-03-06	[tar.xz]	[pgp]	[patch]	[inc. patch] [view diff] [browse] [changelog]
longterm:	2.6.32.65	2014-12-13	[tar.xz]	[pgp]	[patch]	[inc. patch] [view diff] [browse] [changelog]
linux-next:	next-20150410	2015-04-10				[browse]

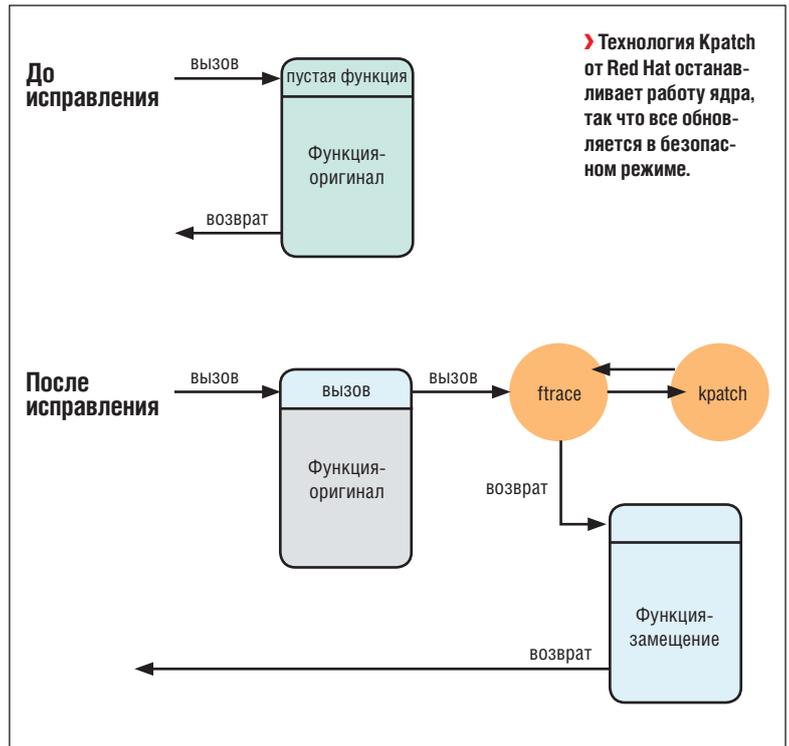
» Похоже, какой-то чудик разместил это зеркало kernel.org на <http://imasheep.hurrdurr.org>.

применить исправление для нашего текущего ядра. Поясним, что это будет текстовая заплатка к исходному коду ядра, использующегося в настоящее время. Механизм исправления компилирует и исправленные, и исходные фрагменты кода и сравнивает их, изучая полученные в результате двоичные файлы. Может показаться странным, что анализ выполняется именно на двоичном, а не более читаемом уровне исходного кода, но так гораздо надежнее, поскольку нам важно знать, как исправление влияет на конечный продукт. Проверки необходимы для того, чтобы убедиться в безопасности заплат (т. е. в том, что изменения будут не слишком радикальны), и если это так, то live-patch преобразуется в модуль ядра. Модуль выполняет необходимое перенаправление кода — добавляя в старые функции операторы перехода к новым.

Латаем на раз

Теперь необходимо прояснить, какие заплатки можно, а какие нельзя применять в live-режиме. У пользователей настольных ПК опыт применения обновлений ядра сводится разве что к установке пакета (обычно называемого *linux* или *linux-image*), когда таковой предоставляется их дистрибутивом. Будь то исправление к новой основной версии (например, переход от 4.0 до 4.1) или применение небольших обновлений безопасности, без изменения версии, подход к управлению пакетами один и тот же: устанавливается совершенно новое ядро, которое вступает в силу после перезагрузки. Механизм live принципиально иной: старое ядро остается в памяти системы, а любой новый код обрабатывается через модуль ядра. Позже мы опишем этот процесс более подробно, но в конечном итоге выходит, что применять можно только небольшие заплатки. Все, что касается ввода новых/других структур данных или новых функций ядра, не может быть реальным кандидатом на динамическое исправление. Таким образом, об обновлениях ядра (даже при малейших изменениях версии) и речи быть не может, но ведь заплатки для этого никогда и не предназначались. Скорее, нам нужен механизм для применения обновлений безопасности, которые крайне редко вносят изменения в структуру.

Подход Kpatch состоит в том, что программа ждет завершения вызовов всех функций перед внесением обновлений, тогда как kGraft выборочно обращается к старым функциям для выполнения текущих вызовов и к новым — для новых. То есть Kpatch немного проще, но там есть задержка в ожидании завершения вызова функции. По сравнению с ним, kGraft требуется принять веские решения, определяя, в какой «среде» (старой или новой) выполняется тот или иной вызов, производя «проверку на реальных данных» и преобразуя соответствующие функции. Хотя kGraft не вызывает дополнительные задержки, замена всех функций может занять некоторое время — могут возникнуть длительные процессы,



» Технология Kpatch от Red Hat останавливает работу ядра, так что все обновляется в безопасном режиме.

препятствующие обновлению любых связанных функций ядра. Livepatch, теперь официальное решение, используемое в ядре, объединяет оба этих подхода и совместимо с пользовательскими инструментами для обоих. Последнее получилось случайно — на момент написания у livepatch еще не было своих; но по факту динамическое обновление как таковое будет еще какое-то время недоступно, в ядре 4.0 мы видим лишь его зачатки. Требуется адаптировать некоторые сложные заплатки, чтобы они соответствовали требованиям динамической системы и, на данный момент, архитектурам помимо x86. Но это не страшно: ведь большинство основных дистрибутивов не используют последовательные версии ядра, и как Kpatch, так и kGraft работают со старыми версиями.

Мы покажем вам, как это работает, на инструментах kpatch от RedHat. Их нужно собрать из исходников, а сначала установить некоторые зависимости. Помимо стандартных *make* и *GCC*, понадобятся также инструменты для работы с исполняемыми файлами ELF и отладочные символы ядра. А также много дискового пространства — компирование двух ядер съест много, и лучше припасти гигабайт 15. Точные названия пакетов, как и точные »

Перезагрузкам не конец

По ходу работы над live-заплатками многие блогеры и новостные сайты вострубили, что машины на Linux теперь не будут нуждаться в перезагрузке. Но это невозможно ни сейчас, ни в ближайшем будущем. В ней нуждаются не только пакеты ядра: PID1, зависящий от обновлений системы *init* (например, *Systemd*) тоже требует перезагрузки. Можно попытаться перезагрузить весь процесс (например, `systemctl daemon-reexec` или даже `kill 1`), но это приведет разве что к панике ядра. Как и при обновлении *glibc*, можно перезапустить

только те службы, которых это касается (то есть практически все), но привычнее все же перезагрузиться.

Обновления драйверов графики, безусловно, требуют перезапуска экранного менеджера (и, возможно, перезагрузки, если они используют KMS) и даже добавить пользователя к новой группе можно только при повторном входе в систему. В последнем случае, если пользователь хочет использовать свои новые привилегии в графическом сеансе, необходимо начать новый (или

использовать менее известную команду `newgrp`). Для многих людей отдельный перезапуск X доставляет не меньше хлопот, чем полная перезагрузка — все равно нужно все закрыть, подождать несколько секунд, а затем ввести пароль. У сидящих с перезапуском оконного менеджера дело проще, зато придется помучиться с перезагрузкой других служб, планированием и еще кучей всяких сопутствующих заморочек. Так что live-обновления — не панацея для тех, кого достали перезагрузки; что, впрочем, не умаляет их пользы.

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

инструкции, зависят от используемого дистрибутива; у нас — Ubuntu 14.04, но инструкции для Debian, RedHat и их производных, и даже для Oracle Linux, есть на <https://github.com/dynup/kpatch>.

Сначала установите требуемые зависимости:

```
$ sudo apt-get install make gcc libelf-dev dpkg-dev
$ sudo apt-get build-dep linux
```

Инструменты Kpatch

Вам нужны отладочные символы текущего ядра; для этого добавьте репозиторий ddebs. От имени root, создайте файл `/etc/apt/sources.list.d/ddebs.list` со следующим содержимым, заменив `trusty` на `utopic` или `vivid`, если вы используете 14.10 или 15.04, соответственно:

```
deb http://ddebs.ubuntu.com/trusty/main/restricted/universe/multiverse
deb http://ddebs.ubuntu.com/trusty-security/main/restricted/universe/multiverse
deb http://ddebs.ubuntu.com/trusty-updates/main/restricted/universe/multiverse
deb http://ddebs.ubuntu.com/trusty-proposed/main/restricted/universe/multiverse
```

Надо еще добавить ключ этого репозитория в `apt`, а затем обновить список пакетов:

```
$ wget -Nq http://ddebs.ubuntu.com/dbgsym-release-key.asc -O- | sudo apt-key add -
$ sudo apt-get update
```

Также мы можем установить отладочные символы ядра:

```
$ sudo apt-get install linux-image-$(uname -r)-dbgsym
```

Теперь извлечем и соберем исходники Kpatch, найденные на GitHub:

```
$ git clone https://github.com/dynup/kpatch.git
$ cd kpatch/
$ make
$ sudo make install
```

Кроме того, нам нужна подходящая заплатка; вот эта изменит `/proc/meminfo`, чтобы вместо него отображалось `VmallocChunk` заглавными буквами. Это пустяковое изменение, и описывается

оно следующими строками, которые надо сохранить в `~/meminfo-string.patch`:

```
Index: src/fs/proc/meminfo.c
=====
--- src.orig/fs/proc/meminfo.c
+++ src/fs/proc/meminfo.c
@@ -95,7 +95,7 @@
     "Committed_AS: %8lu kB\n"
     "VmallocTotal: %8lu kB\n"
     "VmallocUsed: %8lu kB\n"
     -   "VmallocChunk: %8lu kB\n"
     +   "VMALLOCCHUNK: %8lu kB\n"
 #ifdef CONFIG_MEMORY_FAILURE
     "HardwareCorrupted: %5lu kB\n"
 #endif
```

Расшифровывать их утомительно (чтобы соответствовать исходному файлу, вы должны использовать вкладки), так что либо сделайте какую-нибудь тривиальную правку сами, либо скопируйте и вставьте из раздела Quick start репозитория dynup на GitHub. Теперь можно запустить длительный процесс компиляции:

```
$ kpatch-build -t vmlinux meminfo-string.patch
```

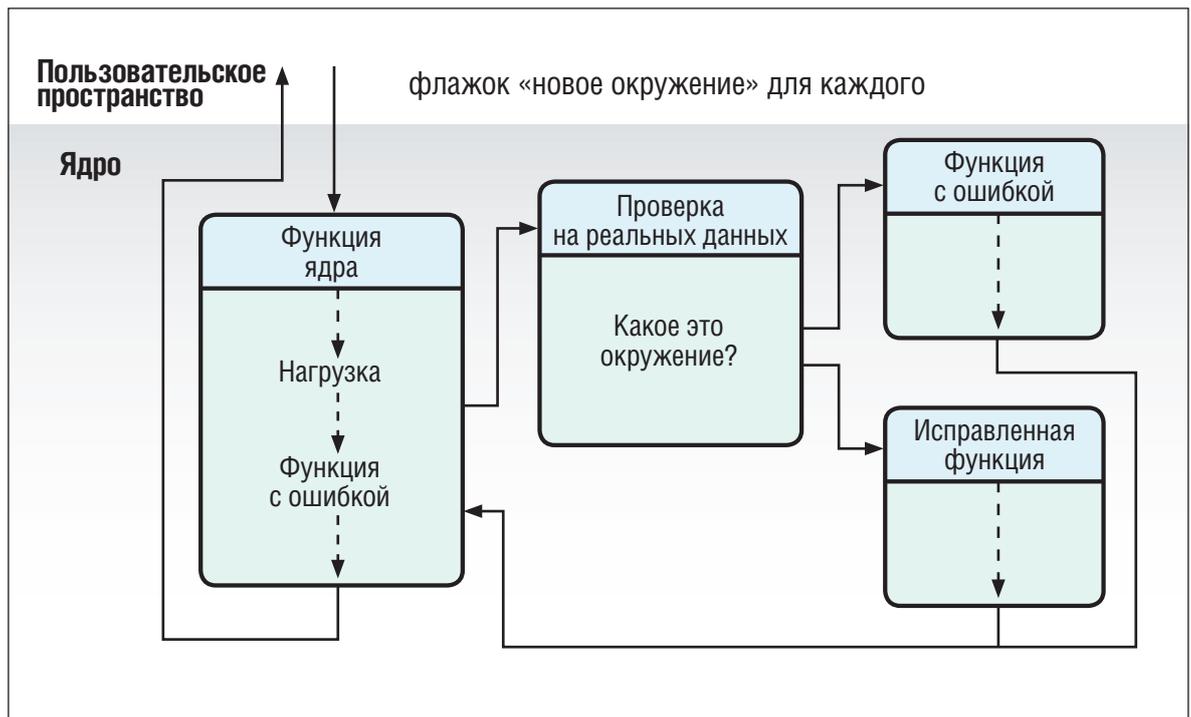
Это займет много времени, поэтому, как только начинается проверка исправления (появится сообщение «Сборка исходного ядра»), можете спокойно улизнуть на кухню и заварить себе первую и, видимо, не последнюю чашку чая. Kpatch скачает соответствующую версию ядра в `~/kpatch/src`, так что если ваша правка не работает, можете сделать новую с помощью `diff`. Наконец, вы увидите радостное сообщение:

```
Building patch module: kpatch-meminfo-string.ko
SUCCESS
```

Модуль будет собран в текущем каталоге, но загрузить его обычными инструментами `insmod` или `modprobe` не получится; используйте

```
$ sudo kpatch load kpatch-meminfo-string.ko
```

Это позволит загрузить основные модули и исправление, и теперь соответствующие строки в `/proc/meminfo` будут выглядеть как-то так:



➤ Подход openSUSE сложнее, но проверки на реальных данных гарантируют состоятельность, хотя задержек не избежать.

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

Заглядывая в будущее

Когда вы будете это читать, релиз ядра 4.1, наверное, уже состоится, и вы будете завидовать своим друзьям с Arch, кому уже будет доступно оно само и множество других потрясающих функций.

Здесь и море графических обновлений, в том числе генерация прошивок для GTX 750 от Nouveau, поддержка виртуальной графики Intel XenGT и vGEM (для ускорения программных рендеризаторов Mesa), не говоря уже о Radeon DisplayPort MST. Присутствует шифрование на уровне файловой

системы для Ext4 (благодаря Google, так как Ext4 используется в Android), улучшенная поддержка программного обеспечения RAID с четностью (т. е. уровни 5 и 6). Также улучшена поддержка ACPI для новых SoC, на базе Intel Atom, и больше внимания уделено процессорам Skylake. Наконец, множество исправлений и новых функций для Flash-Friendly FileSystem (F2FS).

Пока вопрос с 4.2 остается открытым, но уже заявлены поддержка кодирования видео AMD VCE1,

а также новый AMDGPU, и для проприетарных, и для открытых драйверов. Вдобавок ядро 4.2 будет поддерживать таблицу системных ресурсов EFI, и обладатели систем с UEFI смогут обновлять прошивку (согласно спецификации UEFI Capsule Update), не меняя рабочих столов. Видимо, появится также KDBUS, внутренняя реализация системы DBUS IPC, могущая улучшить безопасность и производительность, и мы будем наблюдать еще и расширение присутствия live-заплаток — не пропустите.

VmallocTotal:	34359738367 kB
VmallocUsed:	28348 kB
VMALLOCCHUNK:	34359700664 kB

В нашей заплатке вместо CamelCase [слитное написание, при котором каждое очередное слово начинается с заглавной буквы, — прим. ред.] видим заглавные буквы. Инструмент *kpatch* позволяет определять, какие заплатки доступны и активны. Запуск *kpatch list* подтвердит, что наша заплатка загружена. Мы также можем добавить ее к образу *initrd*, чтобы он запускался при каждой следующей загрузке; это окажется полезным, если новый пакет ядра будет недоступен из-за каких-то уязвимостей, против которых вы и применяли *live-patch*. Если вы решите, что заглавных букв как-то многовато, можете свою заплатку выгрузить:

```
$ sudo kpatch unload kpatch-meminfo-string.ko
```

Вот вам иллюстрация теории. Если вы ас по части ядра, то почему бы не попробовать более продвинутые правки — посмотреть, что работает, а что нет. А если вы новичок — не волнуйтесь: конечным пользователям все эти сумасшедшие фокусы понадобятся еще очень не скоро.

Новые функции, новые ошибки

Несмотря на числовой переход от 3.19 до 4.0, нельзя сказать, что в новейшем ядре Linux полно потрясающих новых функций. Так было и при переходе от 2.6 до 3.0; новый номер не всегда означает новое содержание, хотя по ходу было устранено много ошибок. Конечно, одни *live-заплатки* чего стоят, и, как мы показали на нашем игрешном примере, эта технология надежна, но с ней предстоит еще много работы, чтобы она была готова к использованию. Хотя кое-какие симпатичные новые функции все же есть:

» **DAX (Direct Access, eXciting)** позволяет избежать ненужного копирования в кэш ядра при работе с энергозависимыми устройствами памяти.

» **Lazytime** Системы UNIX используют различные метки времени (*atime*, *mtime*), которые довольно трудно поддерживать. Опция *relatime* позволяет минимизировать прерывания, но нарушает работу некоторых программ. *Lazytime* хранит временные метки в кэше, а не записывает их на диск, улучшая производительность (это подробно объясняется в отличной статье на LWN.net: <http://bit.ly/IntroToLazytime>).

» **KASan (Kernel Address Sanitizer)** Это умный детектор ошибок памяти, способный найти утечки памяти и ошибки быстрее, чем существующий *kmemcheck*.

» **NFS** теперь в версии 4.2 по умолчанию. Также появилась *Parallel NFS*, разделяющая пути данных и метаданных для улучшения масштабируемости.

» **Dm-crypt** стала гораздо более масштабируемой при работе на нескольких процессорах, благодаря использованию несвязанной рабочей очереди.

» **Overlays** Поддерживается несколько нижних слоев; теперь можно надстраивать несколько файловых систем друг над другом.

```

GNU nano 2.2.6 File: meminfo-string.patch
Index: src/fs/proc/meminfo.c
--- src.orig/fs/proc/meminfo.c
+++ src/fs/proc/meminfo.c
@@ -95,7 +95,7 @@
     "Committed_AS:   %8lu kB\n"
     "VmallocTotal:    %8lu kB\n"
     "VmallocUsed:     %8lu kB\n"
     "VmallocChunk:    %8lu kB\n"
+    "VMALLOCCHUNK:   %8lu kB\n"
-    "HardwareCorrupted: %5lu kB\n"
+    "HardwareCorrupted: %5lu kB\n"
 #endif
-----
jonn@Virmint ~ $ uname -r
3.13.0-52-generic
jonn@Virmint ~ $ sudo kpatch info kpatch-meminfo-string.ko
[sudo] password for jonn:
Patch information for kpatch-meminfo-string.ko:
filename: /home/jonn/kpatch-meminfo-string.ko
license: GPL
depends: kpatch
vermagic: 3.13.0-ckt18-52-generic SMP mod_unload modversions
parm: replace:replace all previously loaded patch modules (bool)
jonn@Virmint ~ $
  
```

После выхода ядра 4.0 (кодовое имя 'Hurr Durr I'm a sheep') Линус Торвалдс намекнул, что релиз 4.1 будет более крупным. Во врезке *Заглядывая в будущее*, вверху на этой странице, описано, что именно уготовано для версии 4.1 и что, вероятно, перекоцует и в версию 4.2.

На момент написания подтвердилась ошибка повреждения данных в ядре 4.0 (и рабочих версиях релиза 4.1), влияющая на работу файловых систем Ext4 RAID0. Отчасти ее выявили в ходе регрессионного тестирования с целью с исправления давней проблемы (начиная с 3.14), выдававшей ошибку при работе RAID 0 с блоками длиной [*chunksize*] не степени двойки. Сообщения о повреждении данных появились сразу после релиза, но выявить проблему оказалось сложнее. Исправление включено в 4.0.3 — правда, пострадавших от потери данных это мало утешит.

Ситуация напоминает выход Ext4 в 2012 г., когда отдельные сообщения о повреждении данных вызвали бурю негодования насчет стабильности всей файловой системы в целом. В том случае сообщения об ошибках оказались в основном ложными. Проблема действительно была, но касалась только тех, кто использует нестандартные параметры монтирования во всяких нестандартных ситуациях, в том числе при двойной перезагрузке с небольшим интервалом.

Ошибки не нужны никому, особенно если это файловая система по умолчанию для многих дистрибутивов, и данные конкретные ошибки служат доказательством тому, почему следует проявлять осторожность, доверяя свои важные данные [Ред.: — В смысле, «биты».] новейшим версиям ядра. Да не забывайте регулярно выполнять резервное копирование. **LXF**

» Немного повозившись с ядром, мы сумели залатать Mint 17.1.

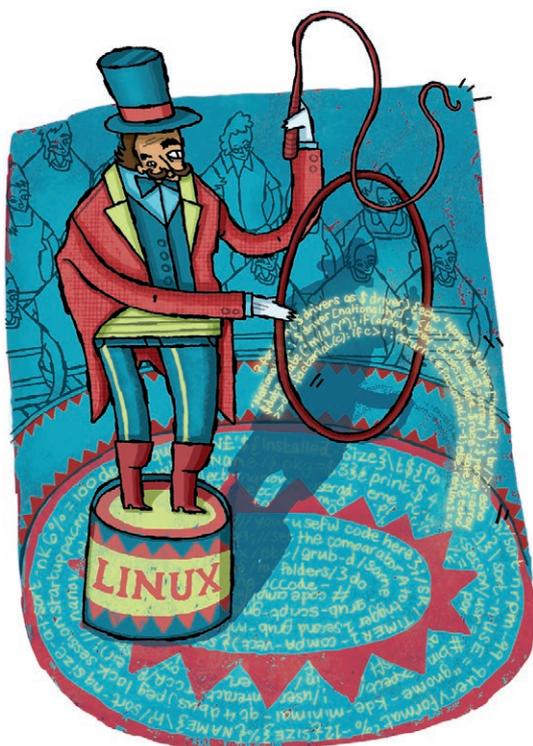
IPv6: Как к нему подключиться

Мы гадали, как построить убежище в защиту от неминуемого исчезновения IPv4, а Джонни Бидвелл подумал, что его урок решит проблему.



Наш эксперт

Джонни Бидвелл часто прячется под стол в страхе перед апокалипсисом IPv4, хотя обычно выманивается шоколадным батончиком или бесплатной пиццей.



Февраль 2011 г. ознаменовался тем, что IANA (Internet Assigned Numbers Authority — Администрация адресного пространства Интернет) выделила пять оставшихся блоков /8 (каждый на 16 миллионов адресов, и это последние поскребышки) пяти региональным интернет-регистраторам (RIR). Теперь выделять больше нечего: адреса закончились. Лот снят с продажи. Конечно, вы все еще можете получать адреса для новых устройств, виртуальных машин и прочего, но адреса рано или поздно иссякнут, а пополнить запасы будет неоткуда.

Впрочем, панике здесь не место — годы размышлений и тестирования привели к тому, что протокол для замены уже есть, и его рабочая версия включена в Linux с 2005 г. Некоторые крупные игроки уже ввели в действие свои сети IPv6. Если вы в Великобритании, то ваш провайдер скорее всего еще не предоставляет подключение по IPv6 (а это очень просто, и таким счастливицам наш урок не потребуется), но мы решим эту проблему с помощью туннеля. Здесь важно отметить, что полное изгнание IPv4 не планируется: оба протокола прекрасно работают бок о бок. Сетевым адаптерам можно назначать адреса обоих типов, а двустековые маршрутизаторы умеют работать с трафиком обоих типов. Но по исчерпанию старых адресов новые хосты будут доступны только через IPv6, и те, у кого будут только адреса IPv4, обнаружат, что больше

не могут подключиться к определенной части Интернета (а отчасти и наоборот). Хотя некоторое сетевое оборудование (например, роутеры), возможно, потребуется обновить, обычные сетевые карты и маршрутизаторы прекрасно сработают с IPv6 — все они передают фреймы Ethernet независимо от природы пакетов более высокого уровня, в которых те содержатся.

В IPv4 для адресации узлов используется 32 бита, причем по соглашению они разбиваются на четыре группы по 8 бит (байт, или октет); каждый октет записывается в десятичном представлении и отделяется от других точками. Поэтому доступный объем адресов IPv4 составляет около 4,3 миллиарда адресов — примерно по одному адресу на каждого жителя Земли, который наслаждается доступом к Интернету. Так как к Интернету подключается все больше и больше устройств, а режимы разделения адресов NAT на уровне носителя не выглядят лучшим решением, мы стремительно приближаемся к тому, что адресам IPv4 придет конец. Для сравнения, на каждый адрес IPv6 выдается 128 бит, а значит, такие адреса закончатся нескоро (их хватит даже каждому атому на поверхности Земли). В стандартной записи адрес разделяется на 8 групп по 16 бит в каждой. Каждая группа записывается четырьмя шестнадцатеричными цифрами и отделяется от других двоеточием.

Много шестнадцатеричного

Так как 32 шестнадцатеричных числа и семь двоеточий писать тяжело, есть несколько сокращений. Сокращение первое — все ведущие нули в каждой группе можно опустить: так, **0123** становится **123**. Сокращение второе — последовательность соседних нулевых групп (**0000**) заменяется на **::**. Например, рассмотрим адрес обратной петли (аналог **127.0.0.1** в IPv4), который, по первому сокращению, можно записать как **0:0:0:0:0:0:1**. Затем первые семь групп бесцеремонно сокращаются, и останется записать только **::1**. Обратите внимание, что двойное двоеточие в адресе можно использовать только один раз, потому что иначе появится двусмысленность. Адрес IPv6 разделяется на две части: первые 64 бита (или четыре группы) образуют префикс сети, а остальные 64 — идентификатор хоста. Сетевой префикс далее разделяется на префикс маршрута и идентификатор подсети, но здесь мы не будем в это углубляться.

Если у вас обновленный дистрибутив, то поддержка IPv6 скорее всего в нем уже есть. Проверить это можно командой `ip` из пакета `iproute2`:

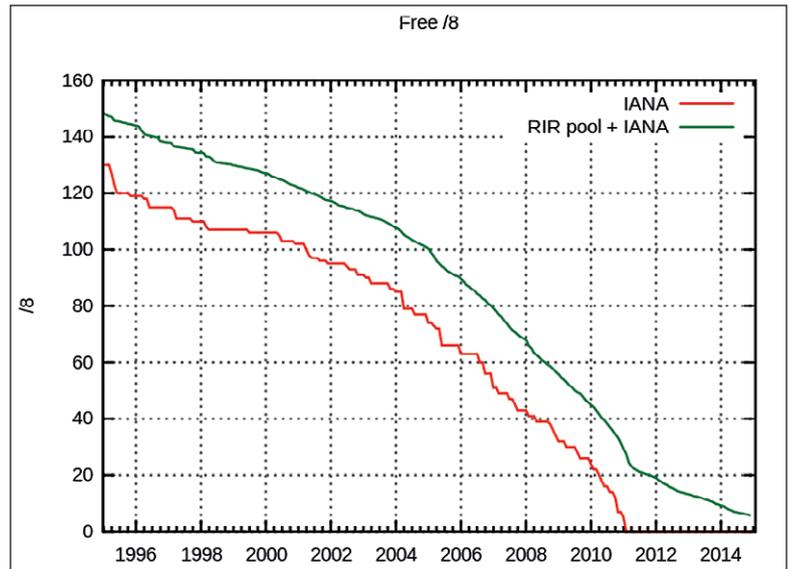
```
$ ip a
...
2: enp5s0: <BROADCAST,MULTICAST,UP,LOWER_UP>
mtu 1500 qdisc fq_codel state UP group default qlen 1000
link/ether 90:2b:34:aa:bb:cc brd ff:ff:ff:ff:ff:ff
inet 192.168.1.144/24 brd 192.168.1.255 scope global
enp5s0
```

```
valid_lft forever preferred_lft forever
inet6 fe80::922b:34ff:feaa:bbcc/64 scope link
valid_lft forever preferred_lft forever
```

Aga! У нашей сетевой карты (enp5s0) есть адрес inet6! Но это не глобально доступный адрес — префикс **fe80::** (где :: означает три группы нулей) подсказывает, что это локальный адрес. Это нечто вроде автоматически назначаемых адресов IPv4 **169.254.x.x**, которые вы, возможно, видели, когда барахлил ваш DHCP-сервер. Сравним идентификатор хоста с MAC-адресом (**90:2b:34:aa:bb:cc**), вы должны обнаружить нечто большее, чем мимолетное сходство. И это не совпадение: локальный адрес образуется путем добавления **ff:fe** в середину MAC-адреса и дополнения седьмого, самого главного бита результирующей строки (для которого ко второму шестнадцатеричному числу прибавляется или из него вычитается два). Локальный адрес должен быть у любого хоста IPv6 — приложения рассчитывают на то, что он есть. К тому же, так как MAC-адреса уникальны (ну, почти: их можно подделать или симитировать), локальные адреса являются уникальными для каждого устройства. Этот способ формирования идентификатора хоста — применяемый и к некоторым типам нелокальных подключений — приводит нас к вопросам конфиденциальности. Главный из них в том, что адреса IPv6 (или бóльшую их часть) можно привязать к отдельным устройствам, а следовательно, и к их владельцам. К счастью, существуют механизмы обеспечения приватности, позволяющие генерировать временные идентификаторы хоста сколь угодно часто.

Настройка маршрута

Конечно, в вашем домашнем роутере может быть настроена (или вовсе отсутствовать) всякая поддержка IPv6. Но не волнуйтесь — мы сможем легко настроить простую внутреннюю сеть IPv6 или один компьютер (если вы ленивы) и/или создать туннель для общения на языке IPv6 со внешним миром. Ваш роутер будет пропускать трафик IPv6 по внутренней сети, даже если не способен отправить его во внешнюю сеть или вообще не знает, что это такое. Сначала нужно назначить так называемый префикс уникального локального адреса (Unique Local Address — ULA) компьютеру внутренней сети, который будет выступать в качестве роутера. Он необходим для сообщения информации о себе и предоставления информации SLAAC другим компьютерам. Префиксы ULA выделяются из зарезервированного диапазона адресов (адреса, начинающиеся с **fd**), который не маршрутизируется из открытого Интернета. Собственный префикс можно сгенерировать на www.simplifiedns.com/private-IPv6.aspx или просто его придумать. Предположим, что мы будем



использовать следующий префикс: **fd00:dead:bad:1dea::/64**. Для простоты сначала настроим идентификатор хоста (оставшаяся часть IPv6-адреса нашего роутера) статическим, краткости ради установив его в ::1. Это удобно сделать командой ip:

```
$ sudo ip addr add dev int0 fd00:dead:bad:1dea::1/64
```

Здесь **int0** — имя сетевого интерфейса (в нашем случае это **enp5s0**, а у вас может быть **eth0**). Теперь можете попинговать себя через IPv6:

```
$ ping6 -c 5 fd00:dead:bad:1dea::1
```

Чтобы назначенный адрес сохранился после перезагрузки, нужно выполнить определенные действия, зависящие от дистрибутива — через графический интерфейс *NetworkManager* или с помощью различных сетевых скриптов (например, *netctl* в Arch Linux). Если вам нужна просто локальная сеть IPv6, пропустите следующую часть статьи (см. раздел «Оповещение о роутере» на стр. 80), в противном случае читайте дальше.

Для общения с остальной частью мира IPv6 (и если ваш провайдер еще не позволяет этого сделать), надо туннелировать наш трафик через IPv4 с использованием популярного протокола **6in4**. Один из вариантов — воспользоваться сервисом туннельного брокера, таким как Hurricane Electric или SixXS. Они предоставляют сервис бесплатно, но вам все-таки надо зарегистрироваться, что в случае с SixXS включает два подтверждения регистрации

» Их было много, но теперь они иссякли. То же можно сказать обо всех ресурсах Земли.

Упражнение по воздержанию от IPv4

Переход на IPv6 в целом происходит ужасающе медленно, несмотря на то, что многие крупные компании провели переключение соответствующих коммутаторов, и на то, что IPv6 является обязательным для мобильного стандарта 4G.

Причина отчасти в том, что IPv4 сумел отжать его с поля. Расширения безопасности IPSec и DNSSec, изначально разработанные как часть IPv6, были включены в IPv4, и еще одной причиной переходить на IPv6 стало меньше. Правда, в 2014 г. многие крупные корпоративные роутеры начали сбивать из-за того, что количество записей в таблицах BGP превысило 512 тысяч. Говорят, что именно это

стало причиной простоя eBay и Lastpass. Хотя всё, за исключением нескольких аукционов, довольно быстро удалось восстановить, это стало тревожным звонком для всех заинтересованных лиц.

Некоторые части системы начали довольно явно покряхтывать, и сетевые ребята взялись за дело. В результате, если сейчас отключить трафик IPv4, большая часть Интернета все равно будет работать.

Если вы пользуетесь туннелем Teredo и настроили его на порт UDP 3544, то следующие правила *iptables* заблокируют все исходящие запросы, фактически отрезав вас от Интернета IPv4, за исключением сервера DNS:

```
# iptables -A OUTPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
# iptables -A OUTPUT -p UDP --dport 53 -j ACCEPT
# iptables -A OUTPUT -p UDP --sport 3544 -j ACCEPT
# iptables -A OUTPUT -j REJECT
```

Вы обнаружите, что одни сайты работают как положено, другие — более или менее, но большинство не работает вообще.

Похоже, вам все еще нужен IPv4, поэтому, закончив эксперименты, удалите эти правила. Если не знаете, с какого начать, удалите сразу все, командой

```
# iptables -F
```

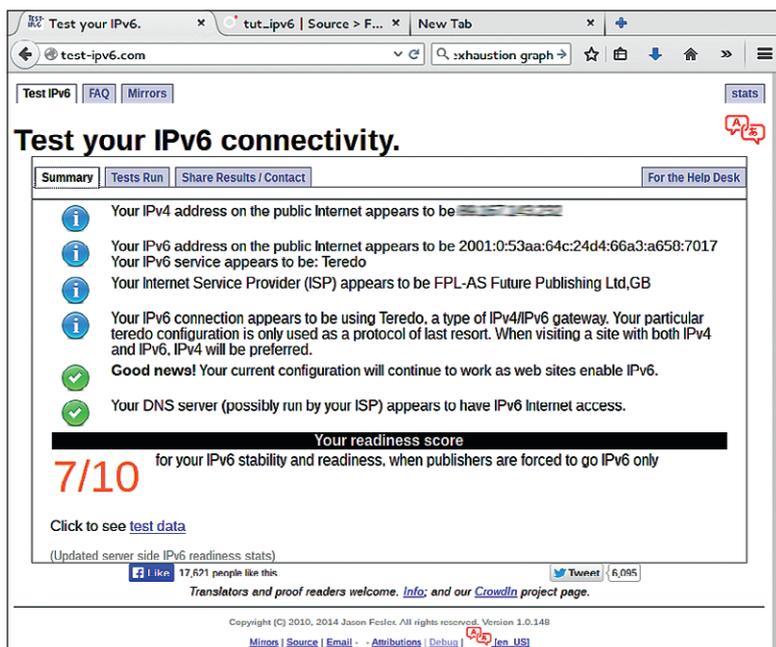
» Подпишитесь на печатную или электронную версии на www.linuxformat.ru/subscribe!

```
[root@bmachine jonnii]# systemctl start miredo
[root@bmachine jonnii]# route
target gateway
default 192.168.133.254 source proto scope dev tbl
192.168.133.0/24 broadcast 192.168.133.169 kernel linkenp0s20
127.0.0.0 local 127.0.0.1 kernel link lo local
127.0.0.1 local 127.0.0.1 kernel host lo local
127.255.255.255 broadcast 127.0.0.1 kernel link lo local
192.168.133.0 broadcast 192.168.133.169 kernel linkenp0s20 local
192.168.133.169 local 192.168.133.169 kernel hostenp0s20 local
192.168.133.255 broadcast 192.168.133.169 kernel linkenp0s20 local
2001::/32 kernel teredo
fe80::/64 kernel enp0s20
fe80::/64 kernel teredo
default kernel lo unspec
default local lo local
:1 kernel none lo unspec
2001:0:53aa:64c:24d4:66a3:a658:7017 local local none lo local
fe80::ffff:ffff:ffff:ffff local local none lo local
fe80::b0c1:b0d1:4c26:6df0 local local none enp0s20 local
ff00::/8 kernel enp0s20 local
ff00::/8 kernel teredo local
default unreachable kernel lo unspec
[root@bmachine jonnii]# ping6 -c 2 ipv6.google.com
PING ipv6.google.com(14hr08s07-in-x0e.1e100.net) 55 data bytes
64 bytes from 14hr08s07-in-x0e.1e100.net: icmp_seq=1 ttl=59 time=60.6 ms
64 bytes from 14hr08s07-in-x0e.1e100.net: icmp_seq=2 ttl=59 time=55.45 ms
--- ipv6.google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
```

➤ Туннели Teredo — последний шанс получить прямой туннель IPv6 к хосту. Но не делайте этого, хотя запретный плод сладок. Особенно если не умеете настроить роутер...

человеком вручную. Существует несколько способов инкапсуляции трафика IPv6 в пакеты IPv4, но если вы находитесь внутри NAT, проще всего воспользоваться протоколом Anything in Anything (AIYYA). SixXS предоставляет удобную утилиту автоматической настройки IPv6 (*Automatic IPv6 Connectivity Configuration Utility*) (доступна в составе пакета *aiccu*), которая запросто настроит туннель. При установке этой утилиты одни дистрибутивы запросят у вас учетные данные SixXS, а в других потребуется вручную изменить файл */etc/aiccu.conf*, чтобы он выглядел так:

```
username <username>
password <password>
protocol tic
server tic.sixxs.net
IPv6_interface sixxs
automatic true
requiretls true
pidfile /var/run/aiccu.pid
defaultroute true
```



➤ Хорошие новости: все стабильно и работает. Хотя я бы так не сказал...

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

```
makebeats true
behindnat true
```

Если вы запустите *aiccu*, всё должно получиться. Hurricane Electric (<https://tunnelbroker.net>) требует указывать конечные точки туннеля вручную (через web-интерфейс, где можно выбрать географически ближайшую точку — \$HE_ENDPOINT4 в примере ниже), а также настраивать туннель вручную на стороне клиента. Кроме того, компания требует, чтобы внешний IP-адрес роутера пинговался из внешнего мира, хотя перенаправление портов не требуется. Для настройки маршрутов IPv6 необходимы сообщения ICMP. Hurricane Electric назначает вам адрес клиентской конечной точки IPv6, начинающийся с **2001:** и заканчивающийся на **/64**, который в коде ниже называется \$CLI_IP6. На этот сетевой префикс можно подвесить любое количество хостов, и настраивать новый туннель для каждого не надо. Ниже приведены команды для настройки туннеля, в которых \$INT_IP4 — ваш внутренний IP-адрес. Обратите внимание, что эти команды следует выполнять с правами root, поэтому при необходимости используйте sudo:

```
# ip tunnel add he-IPv6 mode sit remote $HE_ENDPOINT4 local
$INT_IP4 ttl 255
# ip link set he-IPv6 up
# ip addr add $CLI_IP6 dev he-IPv6
# ip route add ::/0 dev he-IPv6
# ip -f inet6 addr
```

Вам надо направить трафик протокола 41 (не порта 41 — это нечто иное) на свой домашний роутер, хотя многие роутеры не позволяют такое сделать (но не роутеры с DD-WRT). Взамен, если вам хватит храбрости, можете поместить свой компьютер в демилитаризованную зону домашнего роутера (DMZ), чтобы на него направлялись все входящие подключения. Но это серьезный риск с точки зрения безопасности, так что лучше не делайте этого. Если что-то не получается, воспользуйтесь инструкциями Teredo (см. стр. 81 сверху).

Также следует знать, что DNS вашего роутера может не возвращать записей IPv6. Это можно обойти, воспользовавшись DNS-серверами Google. Для этого добавьте **nameserver 8.8.8.8** в */etc/resolv.conf* или измените настройки сервера DNS в *NetworkManager*. Запросы DNS, направляемые через IPv4, все еще могут возвращать адреса IPv6, но вы также можете воспользоваться адресом IPv6 **2001:4860:4860::8888**. Проверьте это командой `$ ping6 -c 5 ipv6.google.com`

Оповещение о роутере

Кроме настройки статического адреса IPv6 вручную или получения префикса от брокера, которое мы описали ранее, можно получить адрес IPv6 одним из двух способов. Первый — автоматическая настройка адреса без состояния (SLAAC), в котором хост общается с роутером по протоколу Neighbor Discovery Protocol. Затем локальный роутер отвечает на запрос, сообщая сетевой префикс и прочие параметры. Объединив эту информацию с идентификатором хоста и воспользовавшись процедурой получения адреса из MAC-адреса, описанной выше, или расширениями приватности, описанными далее, мы получим адрес IPv6. Кроме того, можно применить более традиционный подход с сохранением состояния, получив адрес с DHCPv6, где сервер запоминает, какие адреса кому назначаются.

Чтобы клиенты в вашей сети могли получить информацию об адресах IPv6, ваш роутер должен отвечать на запросы оповещениями о себе. Без этого никто не услышит мольбы наших безадресных интерфейсов. Демон отправки информации находится в пакете *radvd*, который надо установить. Он создает простую

Туннели Teredo

В порядке альтернативы использованию брокера туннеля на одном компьютере можно воспользоваться туннелем Teredo. Teredo — это технология, разработанная Microsoft, которая опирается на полосу пропускания передатчиков Teredo. Передатчики выступают в качестве шлюзов, разворачивая IPv6-пакеты и отправляя их и оборачивая их в пакеты IPv4 для возврата. Это уж последнее прибежище, и поскольку на туннель можно назначить только один IPv6, оно непригодно в сетях, даже в небольших. Так как создаваемый туннель может успешно обойти ваш брандмауэр, это считается явным

риском для безопасности. В целом использование туннелей не одобряется и постепенно прекращается. Однако они очень удобны для переноса слоев NAT (teredo [корабельный червь] — это на самом деле вид существ, которые прогрызают дыры в корпусе корабля), поэтому ими удобно пользоваться в случаях, когда ваш внешний адрес IPv4 регулярно меняется или ваша сеть «недружелюбна» по какой-либо другой причине.

Клиент с открытым кодом по имени *Miredo* доступен в репозиториях вашего дистрибутива. Установив его и запустив сервис (`systemctl start miredo`

для *SystemD*), вы сможете насладиться IPv6. Для адресов Teredo зарезервирован префикс **/32**, а начинаются они с **2001:0**. Если все пройдет как планировалось, вам назначат адрес. Чтобы это проверить, зайдите на сайт <http://test-IPv6.com>, который даст вам (или, по крайней мере, вашему IPv6-подключению) оценку по десятибалльной шкале. С Teredo вы десяти баллов не получите, и вдобавок сервис *Miredo* может неправильно определить ваш внутренний IP-адрес. В последнем случае просто измените файл `/etc/miredo/miredo.conf`, добавив туда строку `BindAddress 192.168.1.10`

локальную сеть IPv6 без внешнего подключения, хотя возможно сообщать информацию и о туннелях брокера из предыдущего раздела. В файле `/etc/radvd.conf` есть много примеров определенных. Для простоты мы сохраним эту конфигурацию и воспользуемся гораздо более простой настройкой (для обеих задач потребуются права root).

```
# mv /etc/radvd.conf{,example}
# nano /etc/radvd.conf
```

Теперь добавьте следующие параметры в конфигурацию — замените `int0` на имя своего интерфейса:

```
interface int0 {
  AdvSendAdvert on;
  MinRtrAdvInterval 3;
  MaxRtrAdvInterval 10;
  prefix fd00:dead:bad:1dea::64 {
    AdvOnLink on;
    AdvAutonomous on;
    AdvRouterAddr on;
  };
};
```

Если вы настраивали туннель Hurricane Electric, можете воспользоваться назначенным вам префиксом **/64** вместо указанного. Это позволит вашей сети IPv6 увидеть внешний мир. Чтобы это работало, надо включить перенаправление трафика IPv6, командой

```
# echo 1 > /proc/sys/net/IPv6/conf/int0/forwarding
```

Теперь запустите сервис — командой

```
$ sudo service radvd start
```

или командой

```
# systemctl start radvd
```

в зависимости от системы `init` вашего дистрибутива. Если все хорошо, то при подключении других компьютеров к нашей сети они будут автоматически настраиваться через SLAAC, и в качестве дополнительного бонуса вы также настроите и локальный компьютер. Чтобы настройки остались постоянными, включите сервис *radvd* — например, командой

```
$ systemctl enable radvd
```

и измените параметр

```
net.IPv6.conf.default.forwarding=1
```

в файле `/etc/sysctl.conf` (или в таком файле, как `/etc/sysctl.d/10-ipv6-forward.conf`). Ваш роутер сообщит информацию о себе также и самому себе, поэтому при запуске данного сервиса вы автоматически получите адрес IPv6. Обычно идентификатор хоста формируется из MAC-адреса как описано ранее, но в некоторых дистрибутивах по умолчанию включены расширения приватности, и в них формируется случайный идентификатор (обычно раз в день). Проверьте настройку клиентов (опять же заменив `int0` на имя своего интерфейса):

```
$ cat /proc/sys/net/IPv6/conf/int0/use_tempaddr
```

Если команда вернет `0`, расширения приватности не включены. Чтобы это исправить, выполните следующую команду с правами root:

```
# echo 2 > /proc/sys/net/IPv6/conf/int0/use_tempaddr
```

Как и прежде, чтобы сделать изменения постоянными, добавьте `net.IPv6.conf.int0.use_tempaddr = 2`

скажем, в файл `/etc/sysctl.d/20-ipv6-tempaddr.conf`.

Но помните, что после получения адреса IPv6 (либо от провайдера, либо от туннельного брокера или Teredo, см. врезку вверху) ваш хост будет доступен из внешнего мира — а также и любые сервисы, запущенные на вашем компьютере, в зависимости от их настроек. С IPv6 мы более не зависим от NAT и не испытываем огорчений при настройке клиентов BitTorrent. Здесь было бы разумно настроить какой-то брандмауэр, но пока ваши сервисы на интерфейсах IPv4 настроены только на прослушивание, все должно быть отлично. Единственный неприятный побочный эффект IPv6 — появление теневых сетей. Они возникают, когда трафик IPv6 может избежать мер безопасности, заданных только для IPv4, что позволяет атакующему свободно обходить правила брандмауэра. Авось в ближайшем будущем провайдеры предоставят подключения по IPv6, и наш урок отправится в утиль. Мы также надеемся, что Интернет не сгинет и что скоро настанет зима... **LXF**

The screenshot shows a web browser displaying the 'IPv6 test' results. The page title is 'IPv6 test - IPv6/4 connectivity and speed test - Mozilla Firefox'. The browser address bar shows 'http://ipv6-test.com'. The page content includes a navigation menu with 'General', 'Speed', 'Ping', 'Website', 'Stats', and 'API'. Below the menu, there is a description of the service and two main sections: 'IPv4 connectivity' and 'IPv6 connectivity'. The 'IPv4 connectivity' section shows 'IPv4' as 'Not supported'. The 'IPv6 connectivity' section shows 'IPv6' as 'Supported' with a green checkmark. It lists the IPv6 address as '2001:0:53aa:54c:ba56a3:a658:7017', the type as 'Teredo', and the Teredo server as '83.170.6.76'. Other details include 'v4 address', 'SLAAC' (No), 'ICMP' (Reachable), 'Hostname' (No), and 'ISP' (FutureNet). At the bottom, there is a 'Score' section showing a red bar and the value '11 / 20'.

➤ В отказе от земных радостей, предлагаемых IPv4, мало осязаемого результата, но 6 июня 2014 года это сделали многие системные администраторы.

Система: С чем едят процессы

На пятом уроке **доктора Криса Брауна** мы изучим процессы — вызовы `fork()`, `exec()` и другие, а также взглянем на каналы как системные программисты.



Наш эксперт

Доктор **Крис Браун** обучает, пишет и консультирует по Linux. Ученая степень по физике элементарных частиц ему в этом совсем не помогает.

Вот небольшой мысленный эксперимент для вас — вроде того, что проделал Шредингер со своим котом (не рекомендуем повторять его эксперимент, если не хотите, чтобы за вами пришли из Королевского общества по защите животных от жестокого обращения). Представьте, что вы могли бы разобрать свой компьютер вплоть до отдельных компонентов. У вас оказались бы кучки проводов, разъемов, интегральных схем, креплений, винтов и даже несколько резисторов и конденсаторов. Но не оказалось бы кучки процессов! И это странно, потому что если перед разбором компьютера вы выполнили бы команду:

```
$ ps -e | wc -l
```

то увидели бы, что их внутри компьютера более ста.

Так что же это — процесс? И как его создать? Процесс — это несколько абстрактное понятие. Распространенное определение процесса — «экземпляр выполняемой программы», хотя лично мне оно не кажется особенно удачным. Я предпочитаю считать процесс своего рода контейнером, который содержит контекст, необходимый для выполнения программы. Чтобы немного дополнить его, в таблице (внизу) показаны некоторые важные атрибуты процесса.

О некоторых из них я рассказывал в предыдущих статьях, и о некоторых расскажу в следующих. Процессы создаются с помощью системного вызова `fork()`. С точки зрения синтаксиса более простых системных вызовов не придумать, так как `fork()`

не принимает никаких аргументов и возвращает обычное целое число. Но в прочих отношениях это самый странный из всех системных вызовов, потому что его выполняет всего один процесс, а после него их уже два. Непонятно? Возможно, вам поможет следующее сравнение. Программу как список команд можно сравнить со сценарием пьесы как списка фраз и действий. Если расширить эту аналогию, то процесс — это актер, отвечающий за выполнение команд. Можно сказать, что программа пассивна, но процесс активен.

Подумайте о детях

Итак, наш актер стоит в центре сцены и читает реплику “`fork()`”, и тут по его сигналу из-за кулис появляется другой актер и присоединяется к нему. В руках у второго актера — точно такой же сценарий, и дальше, сразу после `fork()`, его читают уже оба актера. Это отдельные процессы, но оба они, по крайней мере поначалу, выполняют одну и ту же программу. Аналогия несовершенна. На самом деле новый процесс не ждет за кулисами, заранее созданный и готовый к действию — он создается вызовом `fork()`.

Исходный процесс называется родительским [parent], а новый — дочерним [child], и вначале дочерний процесс — почти точная копия родительского. Однако между ними есть одно важное различие, которое позволяет процессам понять, кто есть кто, после завершения вызова `fork()`. В родительском процессе `fork()` возвращает идентификатор вновь созданного дочернего процесса. В дочернем `fork()` возвращает ноль. Это означает, что в вызове `fork()` всегда проверяется следующее условие:

```
if (fork()) {
    /* Я родительский процесс */
}
else {
    /* Я дочерний процесс */
}
```

Вот более полный пример, в котором родительский и дочерний процесс крутятся в цикле вывода сообщений:

```
#include <stdio.h>
void main()
{
    int i;
    if (fork()) {
        for (i=0; i<100000; i++)
            printf("*** PARENT %d\n", i);
    } else {
        for (i=0; i<100000; i++)
            printf(" ** CHILD %d\n", i);
    }
}
```

Не буду тратить журнальную площадь на результат, который я получил — наберите код и попробуйте сами. Вы обнаружите,

Атрибут	Описание
Сегмент кода	Область памяти, содержащая исполняемый код
Сегмент данных	Область памяти для глобальных и других статических данных
Стек	Область памяти переменного размера с локальными переменными
Куча	Память, используемая для динамического выделения пространства
Идентификатор процесса	Уникальный числовой идентификатор процесса
Реальный идентификатор пользователя	Числовой идентификатор пользователя, выполняющего процесс
Эффективный идентификатор пользователя	Пользователь, с правами которого выполняется процесс
Текущий каталог	Каталог, к которому процесс будет применять относительные пути
Переменные окружения	Список пар ИМЯ=ЗНАЧЕНИЕ, используемый для задания пользовательских параметров
Открытые файлы	Состояние (например, указатель позиции файла) всех открытых файлов
Обработка сигналов	Как процесс будет обрабатывать сигналы различных типов



Скорая помощь

Список всех именованных каналов в своей системе легко получить командой `$ sudo find / -type p`.

Зачем нужно ветвление?

Зачем программам требуется ветвление? Основных причин на это две. Первая — достижение параллелизма внутри одного приложения. Например, web-сервер, почтовый сервер или сервер баз данных могут создавать дочерние процессы для одновременного обслуживания нескольких клиентов. Хотя многопоточность (которую я здесь не обсуждаю) могла бы быть эффективнее, модель

«по дочернему процессу для каждого клиента» остается популярной.

Вторую причину для ветвления я назову делегированием. Идея делегирования состоит в том, что вы хотите передать какую-то работу другой программе, но сохранить контроль над ней, чтобы вернуть управление после выполнения этой работы. Именно этому посвящен жизненный цикл процесса

fork/exec/exit/wait, что и происходит в оболочке при вводе каждой команды. Вы намереваетесь получить список содержимого каталога, и оболочка делегирует эту задачу команде ls, выполняя ее, но проделывает это в отдельном дочернем процессе, чтобы сохранить свой контроль над текущей командой и вывести приглашение для ввода следующей.

что строки PARENT и CHILD чередуются непредсказуемым образом в соответствии с тем, как планировщик разделяет процессорное время между двумя процессами. Если запустить программу снова, чередование будет другим. То, что поведение этой маленькой программы не детерминировано, немного беспокоит [Ред.: — В детерминистской вселенной?], но это возникает, как только в дело вступает параллелизм. В принципе, вы можете увидеть в выводе примерно такую строку:

```
** PA ** CHILD 5RENT 547713880
```

Сомневаюсь, что вы увидите подобное переключение контекста на практике, но принцип сохраняется: два процесса выполняются одновременно, и предсказать очередность их действий невозможно.

Иногда при запуске программы будет казаться, что командная строка исчезла. На самом деле она просто находится за пределами экрана. Это произойдет, если дочерний процесс завершится после родительского, так как оболочка ожидает завершения родительского процесса, но не ожидает дочернего. Мы вернемся к завершению процессов позже.

Иногда родительский и дочерний процесс продолжают выполнять одну и ту же программу (см. врезку «Зачем нужно ветвление?» вверху), но нередко дочерний процесс предназначен для выполнения совсем другой программы. Для этого он использует семейство системных вызовов, которые (за отсутствием общего имени) я назову `exec()`. Если вернуться к нашей аналогии с актером и сценарием, то `exec()` похожа на строку сценария, которая говорит «Сыграть пьесу „Макбет“». И актер откладывает сценарий, который он играет сейчас, берет сценарий «Макбета», открывает его на первой странице и начинает читать: «Когда при молниях, под гром мы вновь сойдемся в дождь втроем?» [пер. Ю. Корнеева, — прим. пер.] Актер всё тот же, он просто читает другую пьесу.

Семь версий `exec()`

Как показано на дереве решений (на рис. справа), на самом деле существует семь версий `exec()`. Эти семь названий могут запутать, и чтобы этого не случилось, воспользуйтесь схемой:

- 1 Если название заканчивается на 'e', то передается новое окружение; в противном случае сохраняется старое.
- 2 Если в названии есть 'r', то исполняемый файл должен быть обнаружен по пути поиска; в противном случае необходимо передать абсолютный путь с именем файла.
- 3 Если в названии есть 'l', то аргументы командной строки передаются явным списком при вызове функции; в противном случае, если в названии есть 'v', они передаются в векторе (массиве).

Вам помогут примеры:

```
char *argv[] = {"ls", "-l", NULL};
execl("/bin/ls", "ls", "-l", NULL);
execv("/bin/ls", argv);
execlp("ls", "ls", "-l", NULL);
execvp("ls", argv);
```

Обратите внимание, что в случае успеха возврата из функции `exec()` не происходит. Процесс теперь далеко и выполняет какую-то

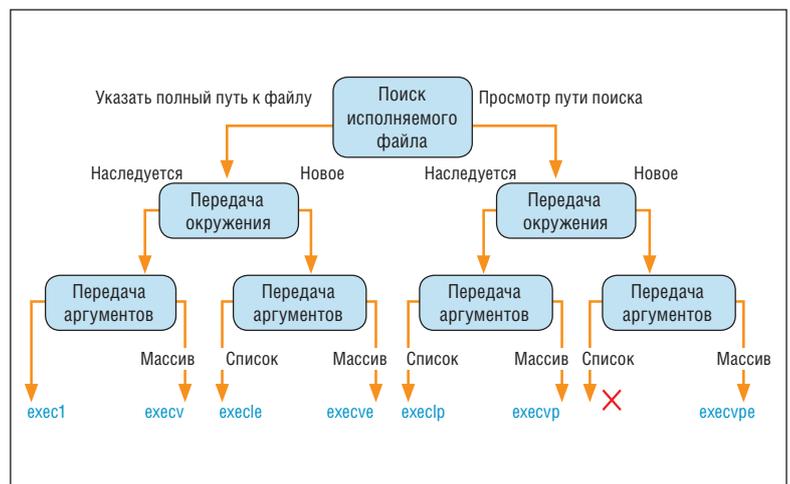
другую программу. Если возврат из `exec()` произошел, значит, вызов завершился неудачно, так как исполняемый файл не был найден или отсутствуют права на его запуск.

Завершение процессов

На этом о создании процессов достаточно. Как же от них избавиться? Самый распространенный способ — добровольное завершение программы вызовом `exit()`. Эта функция принимает целочисленный аргумент, который становится доступен родительскому процессу и называется статусом завершения программы. По соглашению нулевой статус означает успешное завершение, а ненулевой (1–255) свидетельствует о каком-либо сбое. Содержание ошибки зависит от программы; например, `grep`, не найдя соответствия шаблону, возвращает 1. Это соглашение путает программистов C, привыкших, что 0 соответствует неудаче, а ненулевое значение — удачному завершению.

Родительский процесс может подождать, пока дочерний процесс завершит свою работу, вызвав функцию `wait(&status)`, где `status` — целочисленная переменная. Вызов блокируется до тех пор, пока все дочерние процессы не завершат работу (помните, что мы могли разветвлять процесс несколько раз и иметь несколько «детей»). Функция возвращает идентификатор завершившегося дочернего процесса. Статус завершения возвращается в старшем байте переменной `status`, и к нему удобно обратиться с помощью макроса `WEXITSTATUS`. Теперь мы знаем достаточно для того, чтобы накропать маленькую оболочку. Да, правда! Вот и программа:

1. #include <stdio.h>
2. #include <string.h>
3. #include <stdlib.h>
- 4.
5. int prompt_and_parse(char **args) »



» Семь вариантов `exec()` сбивают с толку. Дерево принятия решений должно помочь вам выбрать правильный вариант.

Скорая помощь



В современных ядрах Linux размер канала по умолчанию составляет 65536 байт. Однако предельный объем данных, которые можно гарантированно записать автоматически (т.е. за один раз, не боясь того, что в них попадут другие данные, записанные другим процессом), гораздо меньше — 4096 байт. Это важно, если несколько процессов пишут в один и тот же канал.

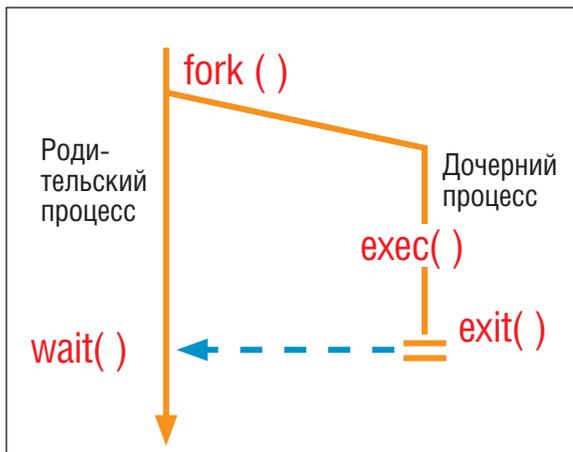
```

6. {
7. static char line[100];
8. printf("> ");
9. if (gets(line) == NULL)
10. return -1;
11. *args++ = strtok(line, "\t");
12. while (*args++ = strtok(NULL, "\t"))
13. /* Пустое тело цикла */;
14. return 1;
15. }
16.
17. main()
18. {
19. char *args[20];
20.
21. /* Цикл основной команды */
22. while (prompt_and_parse(args) > 0) {
23. if (fork() == 0) /* Дочерний */
24. execvp(args[0], args);
25. printf("%s: не найден\n", args[0]);
26. exit(1);
27. }
28. else /* Родительский */
29. wait(0);
30. }
31. }
32. }

```

Вот краткое пояснение. Функция `prompt_and_parse` (строки 5–15) запрашивает команду, считывает ее в строку и разделяет на части, помещая их в массив, переданный в качестве аргумента. Предупреждение: фиксированная длина буфера строки в строке 7 и использование метода `gets()` открывают возможность атаки методом переполнения буфера; это способ нарваться на неприятности, и в реальном коде так делать нельзя. (Фиксированная длина буфера в строке 19 — тоже плохой подход.) Цикл обработки строки в строке 12 — это как «Мармит» [рекламный лозунг этого пищевого продукта со специфическим вкусом — «любить или ненавидеть», — *прим. пер.*]: вы ее либо полюбите, либо возненавидите. Вся фактическая работа выполняется как побочный эффект оценки предиката условия для цикла `while()`.

Цикл основной команды (строки 22–31) выполняет повторяющиеся вызовы метода `prompt_and_parse()`, каждый раз выполняя разветвление и заставляя дочерний процесс выполнить команду, которую мы только что разобрали. Заметьте, что здесь мы выбрали `execvp()`. В строке 29 родительский процесс просто ждет завершения дочернего процесса (игнорируя статус завершения), а затем выполняется очередная итерация цикла. Также обратите внимание на обработку ошибок в строке 25. Она выполняется, только если



➤ Четыре системных вызова — `fork()`, `exec()`, `exit()` и `wait()` — являются кирпичиками жизненного цикла процесса.

вызов `execvp()` завершился неудачно, вероятно, потому, что не удалось найти исполняемый файл. Попробуем программу в действии. Первая строка приглашения (\$) — от моей исходной оболочки (*Bash*); остальные — от *Smallshell*.

```

$ smallshell
> date
Wed May 13 19:50:59 BST 2015
> date -l
2015-05-13
> file smallshell.c
smallshell.c: C source, ASCII text
> ls -l *.c
ls: нет доступа к *.c: Нет такого файла или директории
> cd ..
cd: не найдено

```

Так, первые три команды отработали хорошо; значит, разбор команды, видимо, работает нормально. Со следующей начинаются неприятности, так как наша оболочка не раскрывает маски. Последняя попытка с `cd` тоже завершается неудачно (из-за ошибки `execvp()`), поскольку `cd` — не внешняя программа, и ее нужно встроить в оболочку. Но чего и ждать от 30 строк кода?

Каналы программиста

Что такое каналы, знают все. Это самый популярный механизм межпроцессного взаимодействия в Linux, и они находятся в центре классической философии построения утилит, в которой команды комбинируются друг с другом, например:

```
$ ps -e | wc -l
```

Давайте рассмотрим каналы поближе, глазами системного программиста. Каналы являются однонаправленными. У них есть вход, куда данные записываются, и выход, откуда они считываются. Канал имеет конечный размер и обладает простой синхронизацией поставщика (вход) и потребителя (выход) — если канал полон, то поставщик не сможет записать в канал, а если пуст, то потребитель не сможет прочесть данные из канала. Каналы создаются с помощью системного вызова `pipe()`:

```
int p[2];
pipe(p);
```

После вызова вы получаете два файловых дескриптора: `p[0]` (выход канала) и `p[1]` (вход канала). Чтобы воспользоваться каналом для обмена данными между процессами, мы полагаемся на два момента. Во-первых, открытые файловые дескрипторы наследуются дочерним процессом во время вызова `fork()`. Во-вторых, дескрипторы переживают вызов `exec()`; например, если файловый дескриптор 4 открыт на входе канала до вызова `exec()`, то он будет открытым и в новой программе после вызова `exec()`. (Это упрощение: можно явно задать флаг «закрывать дескриптор при вызове `exec`», и тогда данное поведение изменится, но я не буду углубляться в эту тему.)

Я разделю создание и использование канала на четыре этапа [см. стр. 85]:

```

int p[2];
pipe(p);
if (fork()) {
/* Родительский */
dup2(p[0], 0);
close(p[1]);
exec( ... и так далее ...);
}
else {
/* Дочерний */
dup2(p[1], 1);
close(p[0])
exec( ... upstream ...);
}

```

Регулируем канализацию

Системные вызовы `dup()` и `dup2()` допускают переназначение файловых дескрипторов и являются незаменимыми инструментами «водопроводчика» Linux. Обычно они используются для переназначения стандартного ввода или стандартного вывода процесса.

Вызов `dup(fd)`; копирует дескриптор `fd` в наименьший доступный, т. е. фактически открытый,

дескриптор. Например, следующая последовательность команд —

```
fd = open("foo", ...);
close(0);
dup(fd);
```

подключит стандартный ввод к файлу **foo**. (Помните, что файловый дескриптор 0 по определению ссылается на стандартный ввод.)

Системный вызов `dup2()` чуть понятнее в использовании, потому что вам следует явно указывать, в который именно дескриптор необходимо произвести копирование:

```
fd = open("foo", ...);
dup2(fd, 0);
```

Если исходный дескриптор (в данном случае это 0) уже открыт, он предварительно будет закрыт.

Вот что происходит. На первом этапе наш родительский процесс (A) создает канал, получая дескрипторы каждого конца канала. На этапе 2 процесс разветвляется, и дочерний процесс (B) наследует дескрипторы. На этапе 3 родительский процесс (A), которому в этом примере суждено стать процессом-потребителем, копирует выход канала в стандартный ввод [подробную информацию о `dup2()` см. во врезке «Регулируем канализацию» вверху]. Затем он закрывает дескриптор входа. Это довольно важно, потому что когда процесс-потребитель впоследствии попытается прочитать данные из канала, у него ничего не получится, если канал пуст, до тех пор, пока хотя бы один процесс не откроет дескриптор входа канала. Процесс A, конечно же, не собирается записывать в канал, но то, что он все еще удерживает дескриптор открытым, вызывает проблему. Я обнаружил это несколько лет назад, но все еще помню, как извел немало времени, пытаясь понять, что происходит.

На этапе 4 дочерний процесс копирует вход канала в стандартный вывод и закрывает выход канала. Теперь все готово: процесс B может писать в канал, а процесс A — читать из него.

Я сделал родительский процесс потребителем, а дочерний поставщиком произвольно — можно было и наоборот. На самом деле, более распространенным вариантом является двойное ветвление родительского процесса и настройка дочерних процессов как поставщика и потребителя. Но это лишь пример, который показывает, как работают каналы. Еще один момент, который хотелось бы отметить: дочерние процессы и понятия не имеют о тех штучках, которые проделывал родительский процесс с файловыми дескрипторами — они просто читают свой стандартный ввод или пишут в свой стандартный вывод.

В большинстве механизмов межпроцессной коммуникации должна быть какая-то согласованная «точка встречи», чтобы один процесс мог найти другой. Например, для сокета TCP клиент должен знать IP-адрес и номер порта, который слушает сервер. Другой пример — для использования очереди сообщений System-V все процессы должны использовать один и тот же идентификатор очереди. Но каналы, о которых мы сегодня говорим, не имеют ни имени, ни идентификатора; их иногда называют анонимными. Они работают, потому что два взаимодействующих процесса на самом деле не связаны друг с другом; у них должен быть общий предок, который создал канал и передал им дескрипторы канала.

Именованный канал

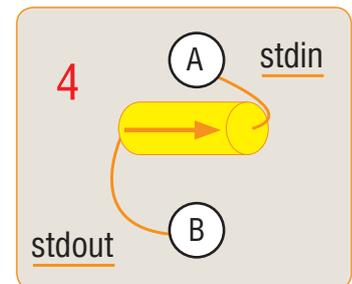
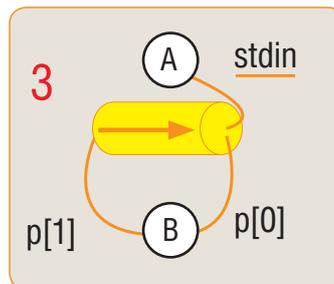
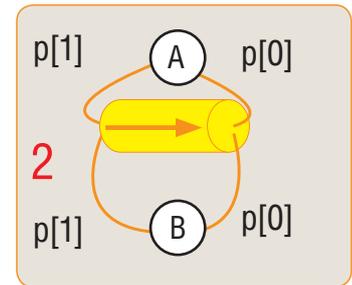
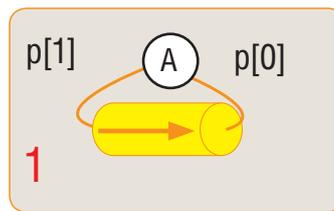
И это плавно подводит меня к теме именованных каналов. Они ведут себя так же, как анонимные каналы, но имеют запись в файловой системе, благодаря которой несвязанные друг с другом процессы могут общаться друг с другом, зная имя канала. Из командной строки именованный канал создается так:

```
$ mkfifo /tmp/mypipe
```

А в программе — с помощью системного вызова `mkfifo()`:

```
mkfifo("/tmp/mypipe", 0666);
```

Если вам интересно, `fifo` — это сокращение от «first in, first out [первым вошел, первым вышел]», ключевого свойства доступа



к каналам. Именованные каналы постоянно существуют в файловой системе. В подробном списке они отображаются с типом 'p':

```
$ mkfifo /tmp/mypipe
$ ls -l /tmp/mypipe
prw-rw-r-- 1 chris chris 0 May 19 18:25 /tmp/mypipe
```

В отличие от них, анонимные каналы существуют только пока есть процесс, который открыл файловый дескриптор этого канала.

С точки зрения системного программиста именованные каналы открываются, читаются и записываются точно так же, как файлы, за тем исключением, что для обмена данными один процесс должен открыть канал для чтения, а другой — для записи. Вместо того, чтобы приводить новые примеры кода на C, вот маленький эксперимент с именованными каналами в командной строке:

- 1 Откройте два окна терминала. Назовем их A и B.
- 2 В окне A создайте канал `/tmp/mypipe`, как я только что показывал.
- 3 В окне A выполните команду `$ cat /tmp/mypipe`. Эта команда будет ждать, пока другой процесс не откроет канал для записи и не запишет туда что-нибудь.
- 4 В окне B выполните команду `$ cat > /tmp/mypipe`.
- 5 Теперь наберите в окне B строки какого-нибудь текста. По мере того, как команда `cat` записывает его в канал, он будет подхватываться и отображаться командой `cat`, запущенной в окне A.
- 6 Введите `^D` в окне B. Эта команда завершит команду `cat`, записывающую в канал. Когда это происходит, команда `cat`, считывающая из канала (в окне A), при попытке чтения из канала получает «конец файла» (потому что вход канала теперь закрыт) и завершается, возвращая вас в командную строку.

На этом пока все. В следующем месяце мы поговорим об идентификации процессов и контроле доступа. Увидимся! **LXF**

» Четыре этапа настройки канала. Подробное объяснение см. в тексте.

**Через месяц:
Процесс
и контроль
доступа**

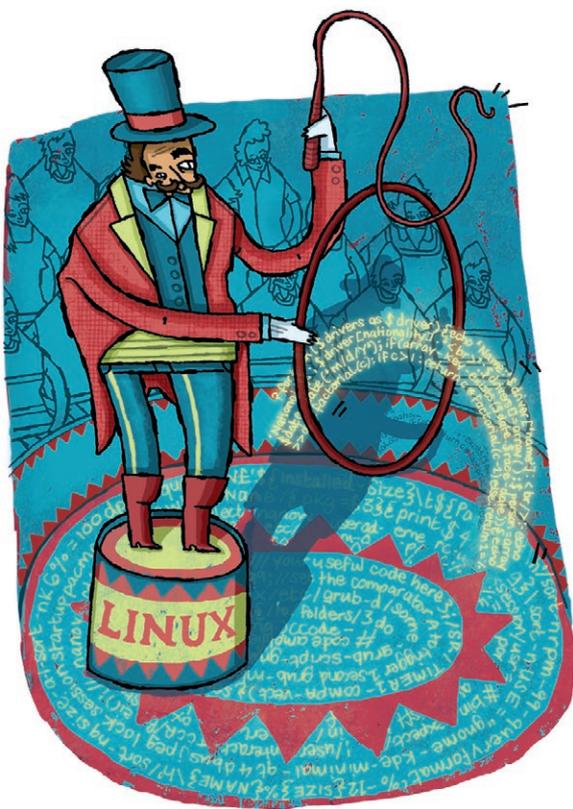
RHR: Поиграем в слова

Программировать не умеете и не будете? Пол Хадсон, бывший редактор *Linux Format*, вернулся, чтобы это исправить, и пишет игру в 80 строк кода PHP. Умница!



Наш эксперт

Пол Хадсон пишет руководства по программированию на hackwithswift.com, пользуется Mac-Book и интенсивно общается на латыни. Обожает Vim.



Когда я в последний раз писал для *Linux Format*, Linux Mint 9 был лучшим из лучших, *Systemd* только возник на горизонте, а 2010-й определенно и бесспорно считался годом Linux на рабочем столе. Сегодня *Systemd* — одна из самых неоднозначных вещей для всего сообщества, Linux на рабочем столе все еще материализуется, а Linux Mint? Что же, Linux Mint все еще хоть куда, а одно попадание из трех — уже неплохо, правда?

В этом учебнике, специально к круглой дате, я намерен научить вас основам PHP. Для этого мы вместе напишем простую игру в слова для командной строки. Игра будет просить игроков угадывать слова по наборам букв, как в мобильной игре *Seven Little Words*. Например, 'MIL', 'KSH' и 'AKE' составят 'MILKSHAKE' и соответствуют подсказке «Корова, попавшая в торнадо». Для еще большей простоты игра будет сообщать точное количество букв в каждом ответе, чтобы на угадывание не ушла целая вечность.

Не волнуйтесь, если ваш опыт в программировании невелик: я объясню всё, на что хватит места, и даже у новичков всё должно получиться. Прежде чем начать, установите PHP. Если у вас дистрибутив на базе Debian, для этого достаточно выполнить команду `apt-get install php5-cli`.

Но это только часть установки. В других дистрибутивах должен быть похожий пакет, но убедитесь, что вы устанавливаете версию PHP для командной строки, а не модуль *Apache*: если менеджер пакетов просит установить *Apache*, это верный признак ухода на неверный путь.

Другая часть установки — пакет, который позволит читать ввод пользователя с командной строки. По какой-то странной причине он не входит в стандартные пакеты PHP Debian, но это легко исправить. Сначала скомандуйте

```
sudo nano /etc/apt/sources.list
```

и добавьте следующий источник в конец списка источников:

```
deb http://packages.dotdeb.org wheezy all
```

Если у вас версия Debian на основе Jessie или Squeeze, можно указать *jessie* или *squeeze* вместо *wheezy*, и все будет отлично. Сохранит файл, выйдите из *Nano* (Ctrl+o, Enter, затем Ctrl+x) и выполните следующие команды:

```
wget http://www.dotdeb.org/dotdeb.gpg
```

```
sudo apt-key add dotdeb.gpg
```

```
apt-get update
```

```
apt-get install php5-readline
```

Эти команды обновят установку PHP, чтобы она подошла для работы с командной строкой. Возможно, вы подумали, что в так называемом PHP-CLI это было стандартом, но вы ошибаетесь. Печально, однако...

Уровни

Прежде чем начать программировать, надо обдумать, какими должны быть наши уровни. На каждом уровне будет семь загадок, и в каждой загадке будет 2-3 фрагмента, которые составляют ответ на эту загадку.

Я не собираюсь усложнять наш проект, добавляя в него чудовищный XML — вместо этого мы воспользуемся сверхпростым форматом уровней:

1 На каждом уровне будет семь строк.

2 Каждая строка содержит фрагменты, двоеточие и ответ на загадку.

3 Фрагменты отделяются друг от друга символом канала: |.

С учетом этого файл уровня будет выглядеть так — сохраните его на рабочий стол под именем **level1.txt**:

```
MOIOSEIUM: Любимое учреждение коров
```

```
ROIADHIOG: Свинья — дорожный нарушитель
```

```
TRUINK: Багажное отделение слона
```

```
MILIKSHIAKE: Корова в торнадо
```

```
BAIBOIOIOM: Взрывчатая обезьяна
```

```
BULILDOIZER: Бык-дозатор
```

```
HISISTOIRY: Любимый предмет змей
```

Отсмеявшись над моими довольно-таки убогими шутками, можете создать еще несколько файлов уровней (**level2.txt**, **level3.txt** и т.д.), и позднее будет легко добавить их в свой код. Так или

иначе, убедитесь, что общее количество фрагментов составляет 20, чтобы появилась очередь.

Формат уровня я выбрал так, чтобы и людям его было удобно читать и редактировать, и PHP удобно разбирать (т.е. превращать этот текст в настоящую игру). Удобство для людей важно, потому что каждый раз, когда вы читаете эти ужасные шутки, вы теряете одну единицу IQ. А удобство PHP важно потому, что мы можем загрузить уровень всего за 20 строк кода, включая пробелы. Не верите? Пожалуйста!

Разбор файла с PHP

Любой стоящий язык программирования для UNIX прекрасно обрабатывает текст, и PHP — не исключение: одной строкой кода можно загрузить файл, еще одной — разбить строки по двоеточиям и символам канала, и третьей — удалить лишние пробелы. Хотя по шуму в строке PHP далеко до Perl, этот язык определенно очень сжат.

Мы напомним функцию `loadLevel()`. При вызове функции ей надо передать имя файла уровня для загрузки (`level1.txt`, который мы только что создали), но она сделает всю трудную работу по поиску загадок и решений в тексте и вернет всё это в удобно упорядоченном виде.

Чтобы функция заработала, нам понадобятся три массива, или коллекции переменных. Первый будет содержать все буквенные фрагменты, например, 'MIL', 'KSH', 'AKE', 'RO', 'ADH' и 'OG'. Второй массив будет хранить все загадки, то есть на каждую загадку три вещи: саму загадку ('Корова в торнадо'), количество букв в ответе ('9 letters'), и фрагменты, составляющие ответ. Последнее значение необходимо для учета фрагментов слов, угаданных игроком. Третий массив будет хранить только ответы для каждого уровня, чтобы их было легко найти.

Прежде чем окунуться в код, надо осознать шесть фундаментальных положений PHP:

- 1 Все переменные начинаются с \$. Не потому, что дорого стоят, но если хотите удивить друзей, вот занятный факт: символ \$ в начале переменных называется «сигилом [sigil]».
- 2 Создать пустой массив можно с помощью двух квадратных скобок: []. С их же помощью добавляется значение в массив. Массивы в PHP нумеруются с нуля, поскольку... вообще-то разумных причин этому нет, разве что — ну, так сложилось.
- 3 Отдельные элементы массива можно прочитать в цикле. Мы будем пользоваться тремя различными типами циклов, но не для того, чтобы терзать ваш мозг — все эти циклы вправду нужны!
- 4 Функция `file()` загружает файл в массив, каждым элементом которого становится одна строка файла. Она идеально подходит для наших целей, но будьте внимательны: функция оставляет в конце каждой строки символ перехода на новую строку, и его придется удалять.
- 5 Функция `explode()` преобразует текст в массив, разбивая его по заданному символу. Мы сначала разобьем строку по двоеточиям, а затем снова воспользуемся функцией, чтобы разбить фрагменты по символу канала: |.
- 6 Когда функции нужно вернуть значение в код, который ее вызвал, просто напишите 'return', а затем укажите значение.

Нам понадобятся еще несколько функций, но у них «говорящие» имена, поэтому вы сможете догадаться об их назначении сами, просто увидев их в коде.

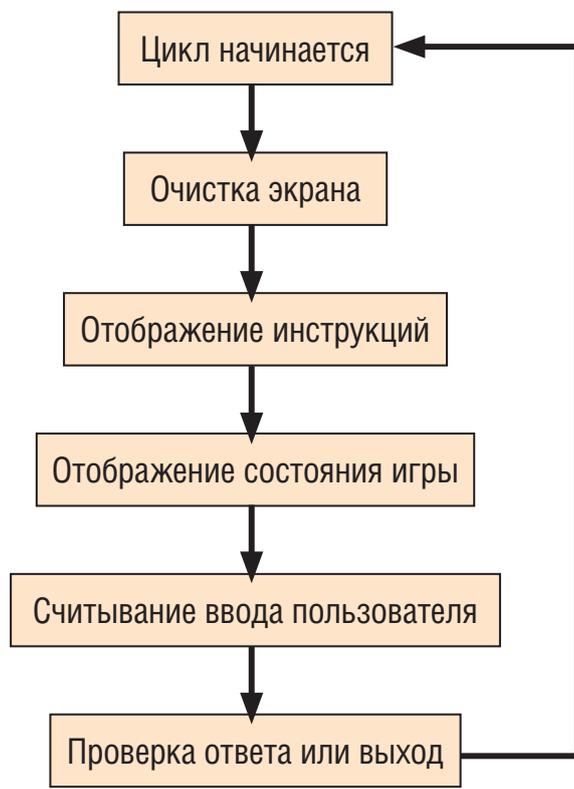
Сохраните следующий код в файле `words.php` на рабочем столе рядом с `level1.txt`:

```
<?php
function loadLevel($filename) {
    $tiles = [];
```

```
    $clues = [];
    $solutions = [];
    $input = file("level1.txt");
    foreach ($input as $line) {
        $line = explode(":", $line);
        $parts = explode("|", $line[0]);
        $tiles = array_merge($tiles, $parts);
        $solution = str_replace("l", "", $line[0]);
        $solutions[] = $solution;
        $clues[] = ["clue" => trim($line[1]), "length" =>
            strlen($solution) . "letters", "parts" => $parts];
    }
    shuffle($tiles);
    return ["letters" => $tiles, "solutions" => $solutions, "clues" =>
        $clues];
}
```

После этого вы сможете запустить файл из командной строки — `php words.php` — и... ничего не произойдет. Отсутствие награды за ваши труды может малость демотивировать, но расслабьтесь: скоро всё будет. Однако сначала я объясню другие функции на случай, если ваши версии иссякли:

- » **array_merge()** объединяет два массива. С ее помощью мы присоединяем фрагменты для каждой загадки к общему набору фрагментов.
- » **str_replace()** заменяет текст указанным значением. В нашем случае символ канала (т.е., как вы помните, |) заменяется пустым значением. Код `$line[0]` означает просто «первый элемент массива» — именно это я имел в виду, говоря, что массивы в PHP нумеруются с нуля.
- » **trim()** удаляет из текста лишние нечитаемые символы. Она необходима, потому что функция `file()` оставляет переходы на новую строку в конце строк.
- » **strlen()** измеряет длину строки.
- » **shuffle()** перемешивает массив, немного усложняя игру.



» Цикл бесконечный, поэтому игра продолжается до тех пор, пока пользователь не наберет `quit` или в ярости не захлопнет терминал.

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

Сравнение с false

Функция `array_search()` (см. стр. 89) говорит нам, где находится значение в массиве, но в ней есть уловка, о которой надо знать: в PHP ноль означает false, а 1 — true, поэтому `if (1)` аналогично `if (true)`.

Если `array_search()` находит значение в позиции 0, она вернет 0. Если она вообще не нашла значение, она вернет

false. Здесь и кроется проблема: напиши мы `$position != false`, это означало бы «если \$position равно false или значению, равному false, такому как 0».

Использование `!==` (с двумя знаками равенства) означает «если \$position равно false», то есть 0 не будет считаться равным false.

Единственная сложность в функции `loadLevel()` касается обработки ею массивов. Например, она может возвращать массивы таким образом:

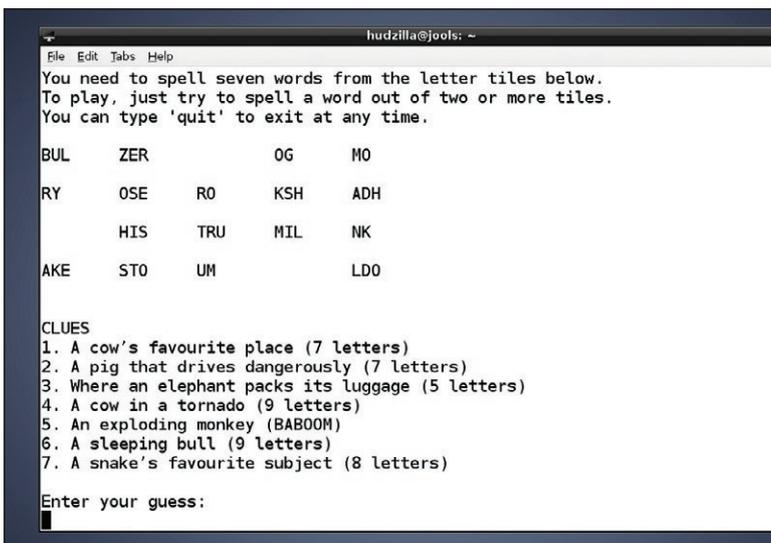
```
return ["letters" => $tiles, "solutions" => $solutions, "clues" => $clues];
```

Я уже объяснял, что массивы содержат коллекции значений, но делать это они могут двумя способами. Первый способ — когда значения снабжаются числовым индексом, например, `$line[0]` и `$line[1]`. Второй способ вы видите в коде выше, и он означает «вместо того, чтобы использовать числовые индексы 0, 1, 2 и т. д., дайте им имена». Тогда к массиву можно обращаться как `$array["letters"]` или `$array["solutions"]`, а это проще запомнить. Если вы спрашивали себя — а я уверен, что спрашивали! — то `<?php` означает «все, что идет дальше — код PHP, и его следует обрабатывать как код». Выйти из этого режима можно с помощью тэга `?>`, и все, что идет за ним, будет просто выводиться на экран.

Состояние игры

Функция `loadLevel()` подготавливает нашу игру к началу, но не содержит никакой игровой логики. Нам все еще надо доделать следующее:

- 1 Отображать текущее состояние игры, то есть все фрагменты слов и все загадки. Рядом с загадками мы покажем количество букв в каждом ответе, если он еще не угадан, или правильный ответ, если он угадан.
- 2 Предлагать пользователю угадать слово до тех пор, пока игра не завершена.



» С отступами и переходами на новую строку наша игра выглядит понятной и приятной для глаз. Что дальше — цвет?

3 Когда пользователь угадывает слово, удалить использованные фрагменты слова и обновить загадку так, чтобы рядом с ней отображался правильный ответ.

Первым делом надо показать текущее состояние игры: все фрагменты и все загадки. Мы сделаем это с помощью новой функции `printStatus()`, которая принимает три параметра: массив фрагментов для вывода и наш массив данных загадки.

Чтобы аккуратно вывести фрагменты слов, нам нужен цикл второго типа, называемый «циклом ИЛИ». Он позволяет перебирать диапазон чисел, в нашем случае от нуля до количества букв. Мы также воспользуемся делением по модулю, более известным как «остаток от деления». Операция «по модулю Y» означает «что останется, если разделить X на Y?». Например, если разделить 20 по модулю 5, получится 0, так как 20 делится на 5 без остатка, а если 22 на 5 — будет 2, так как 20 — это четыре раза по пять и остаток два.

Деление по модулю в этой функции поможет нам расположить фрагменты аккуратно. В сочетании с циклом `for` будем считать от нуля до количества фрагментов (20) и выводить каждый фрагмент на экране, дополняя его символом табуляции, который отделяет фрагменты. Затем с помощью деления по модулю каждый раз, когда текущее число в цикле будет делиться на 5 с остатком 4, мы будем выводить два перехода на новую строку. На практике у нас получится матрица из фрагментов размером пять по горизонтали на четыре по вертикали. Печать загадок по сравнению с этим проста: мы используем второй цикл `for`, который считает от 0 до 7, считывая загадку и длину ответа. Чтобы перейти к коду, надо знать еще три вещи:

1 Чтобы добавить к числу единицу, мы используем два плюса: `++$вашеЧисло`. Два плюса означают «добавить к этому числу единицу и сохранить его в той же переменной».

2 Если вы хотите добавить в текст табуляцию, используйте управляющий символ `\t`. Для перехода на новую строку — управляющий символ `\n`.

3 Для обращения к массивам в тексте используются фигурные скобки, чтобы не засорять текст. Например, «Вот наше значение: `{ $массив["мое_значение"] }`».

А теперь код — поместите его в свой скрипт PHP ниже:

```
loadLevel() :
function printStatus($letters, $clues) {
    for ($i = 0; $i < count($letters); ++$i) {
        echo "{$letters[$i]}\t";
        if ($i % 5 == 4) {
            echo "\n\n";
        }
    }
    echo "\nCLUES\n";
    for ($i = 0; $i < 7; ++$i) {
        echo $i + 1, ". {$clues[$i]["clue"]} ({$clues[$i]["length"]}\n";
    }
    echo "\nВведите свою догадку:\n";
}
```

Самая сложная строка кода находится в цикле загадок, поскольку в ней приходится искать конкретную информацию в большом массиве. `$clues[$i]["length"]` означает «получить значение `$clues`, найти элемент на позиции `$i`, затем получить его длину», и все это делается одной строкой.

Теперь у меня для вас есть хорошая и плохая новость. Плохая — если запустить скрипт, он всё равно ничего не выведет. Но хорошая новость в том, что всего через пару минут всё начнет образовываться, хотя для этого потребуется цикл еще одного типа...

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

Переменные в строках

В PHP широко используется так называемая интерполяция строк. Это означает, что можно написать “Foo bar \$variable meh”, а внутри текста переменная \$variable заменится ее значением. Это, несомненно, очень полезно, но здесь есть и свои проблемы: откуда PHP знает, где заканчивается переменная и начинается текст? В предыдущем примере все было ясно: в переменных PHP не может быть пробелов,

и когда PHP замечает пробел после \$variable, он знает, что \$variable — переменная. В случае массива вы воспользовались бы \$variable[“wombat”], но означает ли это «содержимое переменной \$variable, за которым следует текст [“wombat”]» или «содержимое элемента “wombat” массива \$variable»?

В PHP есть простое решение — оборачивать переменные в фигурные скобки: { и }. Для простых

случаев вроде \$variable это не требуется, но в нашем коде необходимо, поскольку в коде встречается \$clues[\$i][“clue”], и значение в данном случае непонятно какое.

Оказавшись в подобной ситуации, просто заключите переменную в фигурные скобки: {\$clues[\$i][“clue”]}, и они прояснят, какое брать значение переменной.

Пока мы написали две функции, но именно они образуют скелет нашей собственной программы. Чтобы ввести игру в действие, надо вызвать функцию loadLevel() и сохранить возвращенное ею значение, а потом попросить пользователей угадать.

Чтобы все заработало, нужен цикл нового типа, называемый бесконечным. Это цикл, который выполняется всегда или, по крайней мере, пока мы не остановим его, и отлично подходит для игры, в которой игроки должны угадывать до тех пор, пока не выиграют. Внутри этого цикла мы должны очищать экран, выводить инструкции и текущее состояние игры, затем считывать ввод пользователя и выполнять какие-то действия.

Ко вводу пользователя мы скоро вернемся, но пока просто будем получать его, удалять все пробелы из начала или конца, а затем преобразовывать в верхний регистр. Это важно, потому что нам незачем разбираться, набрал ли пользователь ‘milkshake’, ‘MILKSHAKE’ или какой-то промежуточный вариант. Сделав это, мы проверяем два особых случая: когда пользователь ввел пустую строку (т.е. просто нажал Enter) — в этом случае мы начинаем цикл с начала; а если пользователь ввел ‘Q’ или ‘QUIT’ (обязательно большими буквами!), мы выходим из программы.

Чтобы слегка все упростить, следующий фрагмент кода разобьет возвращаемое loadLevel() значение на три различных значения. Это не обязательно для компьютера, но обязательно для сохранения вашего рассудка, поверьте!

Так или иначе, поместите следующий код в конец скрипта PHP:

```
$currentLevel = loadLevel(“level1.txt”);
$letters = $currentLevel[“letters”];
$solutions = $currentLevel[“solutions”];
$clues = $currentLevel[“clues”];
while (true) {
    system(“clear”);
    echo “\nSEVEN LINUX WORDS\n”;
    echo “Вам надо составить семь слов из показанных буквенных фрагментов.\n”;
    echo “Для этого попробуйте составить слово из двух-трех фрагментов.\n”;
    echo “Выйти можно в любой момент, набрав ‘QUIT’.\n\n”;
    printStatus($letters, $clues);
    $line = strtoupper(trim(readline()));
    if ($line == “”) continue;
    if ($line == “QUIT” || $line == “Q”) {
        echo “Спасибо за участие!\n”;
        exit;
    }
}
// проверка ответа
}
```

Кое-что заслуживает прояснения.

Прежде всего, бесконечным циклом управляет while (true), которое означает «выполнять этот код, пока true равно true». Это, разумеется, верно всегда, поэтому и цикл выполняется всегда. Вот только, для очистки экрана терминала мы вызываем функцию system() и передаем ей параметр ‘clear’. Это аналогично ситуации,

при которой пользователь сам набрал бы ‘clear’, и это отличное сокращение.

В-третьих, функция strtoupper() — новая, но тоже простая: она всего лишь конвертирует текст в верхний регистр и возвращает его. В-четвертых, функция readline() тоже новая и тоже на удивление говорящая сама за себя. В-пятых, || в условии означает «или», поэтому эта строка означает «если пользователь ввел Q или QUIT...» Наконец, ближе к концу цикла есть строка // проверка ответа. PHP игнорирует все строки, начинающиеся с ‘/’, так как это комментарии.

Поиск ответов

Если теперь запустить код, то вы увидите, что всё работает — да, честно говоря, уже и пора. Правда, наша игра — еще не вполне игра: вы можете набирать все что угодно, а экран просто обновляется и ничего не происходит. Но благодаря некоторым хитрым условиям осталось всего несколько минут до того, как вы завершите написание этой игры и откинетесь на спинку кресла с довольной ухмылкой на лице.

Всё, что надо сделать — заменить комментарий

// проверка ответа

на код, который проверяет, верен ли ответ пользователя. Если да, то нужно удалить эти фрагменты из игры, обновить список отгадок для отображения ответа и сделать так, чтобы в будущем пользователю уже не попала эта загадка.

Для этого мы воспользуемся новой функцией с именем array_search(), которая определяет, существует ли элемент в массиве. У нас уже есть массив \$solutions, содержащий все возможные ответы, поэтому мы воспользуемся array_search() для поиска ответа пользователя в этом массиве. И вот очередной удобный трюк: положение ответа в массиве \$solutions соответствует положению этого ответа в массиве загадок. А значит, для обновления списка загадок так, чтобы в нем был ответ, а не длина ответа, надо изменить загадку с соответствующим индексом в массиве. Для удаления использованных фрагментов мы используем похожее решение: находим каждый фрагмент ответа в основном массиве \$letters и затем записываем туда пустую строку, чтобы фрагмент больше не был виден.

Следующий код заменяет наш комментарий:

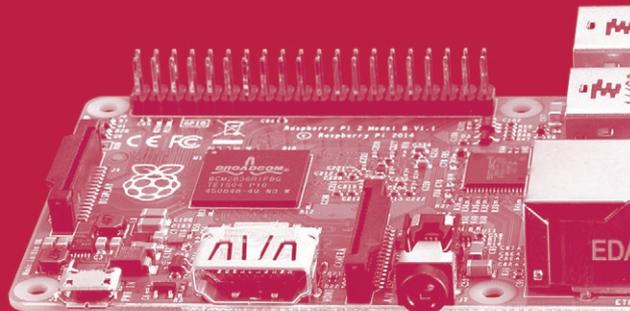
```
$position = array_search($line, $solutions);
if ($position !== false) {
    $clues[$position][“length”] = $solutions[$position];
    foreach ($clues[$position][“parts”] as $letterPart) {
        $letterPosition = array_search($letterPart, $letters);
        if ($letterPosition !== false) {
            $letters[$letterPosition] = “”;
        }
    }
    $solutions[$position] = “”;
}
```

Вот и всё! Мы благополучно завершили написание игры, а теперь в нее можно и поиграть. **LXF**

Скорая помощь

У вас в руках готовая игра, но можно сделать еще немало. Как минимум, можно написать еще один уровень, затем изменить код, чтобы игрок, угадав все слова, переходил на новый уровень. Сделав это, добавьте подсчет очков: добавляйте 10 за правильный ответ и вычитайте 1 за неправильный.

Через месяц:
Web-приложения в Haskell



Ваша порция смачных новостей, обзоров и учебников от Raspberry Pi

БЕН НАТТОЛЛ
специалист
по обучению
в Pi Foundation.



Привет...

В офисе Raspberry Pi Foundation в Кембридже, который мы ласково зовем Башнями Pi, мы занимаемся проектами в области оборудования, ПО и образования.

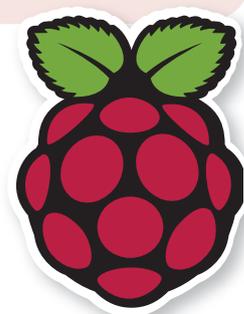
Видимо, вы слышали, что в конце этого года мы отправим два Pi на Международную космическую станцию, в рамках нашей образовательной и информационно-просветительской деятельности в области космических исследований и как часть миссии британского астронавта Тима Пика [Tim Peake]. Пока мы завершаем работу над API Python для платы датчика и учебными материалами, модели Pi B+ и его SenseHAT проходят тесты безопасности для сертификации на полет, и мы нетерпеливо ожидаем, когда же Европейское космическое агентство даст нам зеленый свет, сказав, что мы готовы для работы в космосе.

Тим Пик вернется на Землю через 6 месяцев, а Pi останутся. Мы пока ничего не планируем, но посмотрим, что принесут 2016-й и 2017-й — надеемся, шансы для школьников всего мира запустить свой код в космос!

Мы открыли выставку в Foundation of Art and Creative Technology (FACT) в Ливерпуле, по проекту Build Your Own [Сделай сам]. Neurotic Machines, детище продюсера Фонда Рэйчел Рэйнс [Rachel Rayns], включает вертикальный сад, цветочный горшок с функцией плеера и имитатором панели управления, набитой приборами, с матричным дисплеем 10x10 со звездчатыми пикселями и гигантской кнопкой. Было здорово его собирать, да и выглядит красиво. Через 3 месяца он переедет в замок Норидж: <http://bit.ly/LXFneuroticPI>

Picademy в США

Фонд объявляет о пришествии в Америку — к новым свершениям!



Р icademy, программа повышения квалификации от Raspberry Pi Foundation, уже имеет огромный успех в Великобритании, всего за 8 мероприятий собрал около 200 учителей.

Фонд хочет обучать сотни тысяч педагогов, и с этой целью программа расширяет свой охват и отправляется в США.

Raspberry Pi Foundation объединились с Музеем компьютерной истории, Mountain View, штат Калифорния (www.computerhistory.org), чтобы запустить ряд курсов Picademy в США в начале 2016 г. Первый продлится два дня; шанс узнать, как использовать Raspberry Pi в классе, получат 25 учителей,

которые по окончании станут сертифицированными специалистами обучения Raspberry Pi. Такие курсы пройдут в разных городах США, с целью подготовки около 100 учителей, и если вы заинтересованы в участии, регистрируйтесь на <https://www.raspberrypi.org/picademy/usa>.

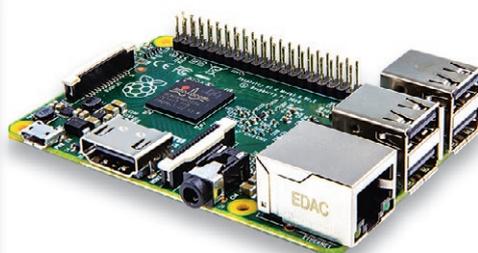
Picademy в Великобритании продолжает укрепляться, так что если вы преподаете здесь и хотите узнать, как принять участие, непременно посетите официальный сайт Raspberry Pi (www.raspberrypi.org). Опыт работы с Raspberry Pi не требуется — только желание учиться и заинтересованность в расширении кругозора своих студентов.

Pi — № 1 и № 3 для хакеров

Народный Pi.

Д ва популярных сайта, LinuxGizmos.com и Linux.com, совместно решили узнать у читателей, какие их три любимых одноплатных компьютера на базе Linux или Android, и с результатом 1721 голос уверенную победу одержал Raspberry Pi2 Model B.

Еще больше впечатляет то, что Model B оказалась не единственным Pi в тройке лидеров. Если второе место занял BeagleBone Black, то на третьем расположился Raspberry Pi Model B+. Результаты опроса и полный список победителей см. на <http://bit.ly/HackerSBCSurvey2015>.



Как насчет ноутбука на Pi?

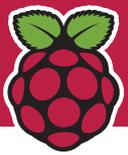
Создайте собственный наладонник.

М ini Raspberry Pi Handheld Notebook от Adafruit — компактное портативное устройство. Изготовленный на 3D-принтере корпус, 3,5-дюймовый сенсорный дисплей и аккумулятор на 2000 мА·ч. А если вы не хотите иметь заляпанный экран, есть также мини-клавиатура и трекпад.

В Adafruit говорят, что с его помощью можно проверить web-камеры, вещать музыку или играть с интерактивными текстами, следя за ходом печати.

За подробностями, в том числе о том, как сделать такой самому, включая образы программ и макеты, обращайтесь на <http://bit.ly/RaspPiHandHeldNotebook>.





AstroPi SenseHAT

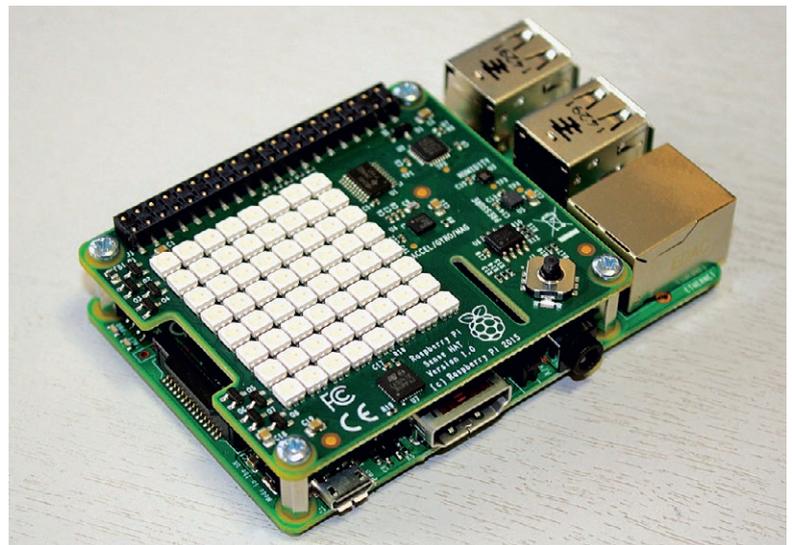
Космос, последний рубеж. Pi-лотная миссия: обучать и развивать. За штурвалом корабля капитан Лес Паундер.

Вкратце

» Новейшая плата-надстройка (HAT, Hardware Attached on Top) от Raspberry Pi Foundation. Имеет множество датчиков для применения в научных проектах. Это часть проекта, позволяющего детям проводить эксперименты по изучению Земли, который будет реализован британским астронавтом Тимом Пиком на Международной космической станции в конце 2015 г.

Raspberry Pi использовался для путешествий по морям, создания роботов, обучения целых классов; теперь же он устремляется в космос с проектом AstroPi. Это научный проект, в рамках которого всем школам Великобритании предложено создать эксперименты, избранные из которых проведет на МКС астронавт Европейского космического агентства (ESA) Тим Пик [Tim Peake]. Также AstroPi — название аппаратной платформы, применяемой с платами A+, B+ и Raspberry Pi 2 (хотя она ладит со всеми моделями Pi). По своим размерам AstroPi повторяет конструкцию A и свободно надстраивается на B+. Плата AstroPi — платформа для сбора данных и научных экспериментов, оснащенная богатым набором датчиков. Для начала, здесь есть датчики температуры и влажности, а рядом — датчик давления. Еще один датчик сочетает акселерометр, гироскоп и магнитометр для получения данных о местоположении. Помимо этого, AstroPi предлагает также сетку разноцветных светодиодов (неопикселей), размером 8×8 и простой джойстик для ввода данных. Светодиодная матрица необычайно яркая (так что для защиты глаз не помешает диффузор вроде листка бумаги).

Но аппаратная часть — ничто без ПО, поэтому AstroPi использует библиотеку Python 3, дающую легкий доступ к датчикам и другим компонентам платы. В составе каждого компонента имеется своя собственная функция для обработки полученных данных, благодаря чему их легко осваивают дети; например, мы тестировали



» SenseHAT — это полное название HAT с датчиками для AstroPi. Он создан для A+, B+ и Raspberry Pi 2.

датчик температуры и смогли отобразить температурное значение на светодиодной матрице, управляя цветом светодиодов, всего за 20 минут — запросто!

Для AstroPi найдется масса применений: от обычного измерения температуры до сложных научных экспериментов по отслеживанию атмосферного давления на высоте с помощью зондов или дронов. Подойдя творчески, AstroPi можно легко связать, например, с *MinecraftPi* и использовать датчики для управления миром игры или светодиодную матрицу, чтобы привести игрока к тайнику.

Raspberry Pi Foundation, Airbus и ESA проработали вместе почти год, проектируя, тестируя и сертифицируя плату для полета на МКС, где ее подключат к Raspberry Pi B+ для проведения экспериментов. B+ выбрали потому, что проверка надежности устройства занимает много времени. МКС диктует строгие требования безопасности: в частности, Raspberry Pi нельзя подключать к сети питания станции — а только к USB-порту ноутбука.

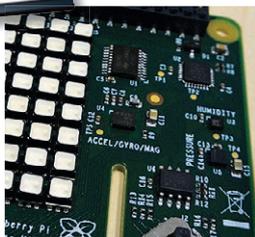
В общем, это фантастическая плата с огромным потенциалом как для использования в сфере образования, так и для хакеров-любителей. **LXF**

Pi-летим на Луну

Конечно, целью AstroPi является Международная космическая станция (МКС), где проведут серию экспериментов, придуманных британскими детьми в рамках национального конкурса. Код и данные, полученные в ходе экспериментов, будут доступны любому ребенку мира, и каждый сможет воспроизвести их и сравнить свои результаты с тем, что получится у Тима Пика. Это будет первый подобный проект, и это очень здорово. Благодаря датчикам давления и температуры, мы ясно увидим, как меняются эти показатели на борту космической станции, пока она движется по околоземной орбите. Любопытно будет узнать, что за эксперименты создадут дети, и попробовать провести их самим.

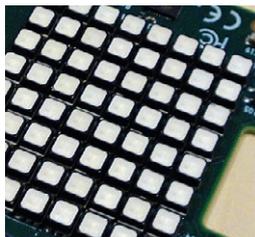


Свойства навскидку



Куча сенсоров

Некоторые сенсоры многофункциональны, к примеру, единый датчик температуры и влажности.



Светодиодная матрица

Панель из 64 светодиодов можно применить для вывода данных от датчиков, посредством слов или цвета.

LINUX FORMAT Вердикт

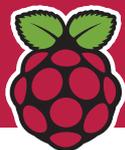
AstroPi SenseHAT

Разработчик: Raspberry Pi Foundation
Сайт: www.raspberrypi.org
Цена: Уточняется (около \$35/£26)

Функциональность	9/10
Производительность	9/10
Удобство в работе	9/10
Документация	6/10

» Простота использования плюс изобилие датчиков — хороший задел для научных открытий.

Рейтинг 9/10



Pigame: Играем с зуммером

Лес Паундер, вдохновленный «Флагом отплытия», собирает классическую ярмарочную игру на Raspberry Pi (но из чуть менее клейкого пластика).



Наш эксперт

Лес Паундер входит в состав образовательной команды Raspberry Pi и любит превращать проекты в жизнь с помощью Pi и завалов пластилина. Он ведет блог на <http://bigl.es>.

Понятие «самоделкин» медленно, но верно набирает популярность в масс-медиа. Теперь оно включает дух «разбирательства и починительства», внесенный в массы одноплатами компьютерами вроде Raspberry Pi и микроконтроллерами вроде Arduino. На этих устройствах можно создать впечатляющий набор проектов: от простых мигающих светодиодов до двуногого друга-робота. Ну, а что можно сделать, если взять немного декоративно-прикладного материала и несколько дешевых компонентов?

На нашем уроке мы создадим игру на твердость руки. В ней надо будет провести металлический щуп вдоль медного проводника, стараясь не коснуться его. При касании мы потеряем одну из трех жизней, а при потере всех жизней игра закончится.

Для этого проекта нам понадобятся: любая модель Raspberry Pi, свежий релиз Raspbian, пластиковый контейнер, пластилин или специальная замазка-сугру, макетная плата, три светодиода, четыре резистора по 220 ом (КРАСНЫЙ, КРАСНЫЙ, КОРИЧНЕ-ВЫЙ), один зуммер, 16 соединительных кабелей (11 «папа-мама», 1 «мама-мама» и четыре «папа-папа»), одна кнопка, динамик для гнезда 3,5 мм (или монитор с динамиками, если вы решите воспользоваться HDMI) и, разумеется, металлическая скрепка для крючка.

Для начала выключим Raspberry Pi и подсоединим компоненты согласно схеме на соседней странице. В высоком разрешении она доступна на <http://bit.ly/LXF200WireGame>. Мы здесь используем последний релиз Raspbian, который теперь включает улучшенный рабочий стол. И «под капотом» — одну из важнейших библиотек Python для этого проекта (*pygame*), предустановленную для использования с Python 2 и 3.

IDLE и библиотеки

Начнем наш проект, открыв терминал. Для этого щелкните по иконке, напоминающей монитор с черным экраном, в левом верхнем углу. В терминале выполните с правами root следующую команду, для запуска среды IDLE3:

```
$ sudo idle3 &
```

& приставлено для того, чтобы отправить команду в фоновый режим и вернуть управление командной строкой пользователю. Теперь должно запуститься приложение IDLE3. По умолчанию оно откроется с окном оболочки, в которой можно тестировать логику и писать небольшие программы, и оболочка будет сразу возвращать ответ. Оболочка также используется как средство для вывода при запуске более крупных проектов, как тот, который мы создадим. Мы не хотим оставаться в оболочке, поэтому выберите File > New Window [Файл > Новое окно], чтобы создать новый документ. В нем мы будем писать код проекта, но чтобы увидеть результаты его работы, придется сначала сохранить и запустить код. Сохраните пустой документ, выбрав File > Save [Файл > Сохранить],



➤ Наша цель — создать классическую ярмарочную игру на проверку навыков, в которой надо проследовать вдоль проволоки, не касаясь ее. При касании проволоки мы теряем одну жизнь, а Raspberry Pi отображает предупреждающую картинку и проигрывает мелодию.

и назовите его **wire.py**. Не забывайте регулярно сохранять свою работу, чтобы снизить риск потери данных.

Начнем свою Python-программу с импорта трех библиотек:

```
from time import sleep
import RPi.GPIO as GPIO
import pygame
```

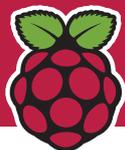
Сначала мы импортируем из библиотеки *time* функцию *sleep*; в определенные моменты она будет задерживать выполнение кода. Мы импортируем только одну функцию, потому что нам только она и нужна; импортировать всю библиотеку было бы неэффективно. Во второй строке мы импортируем библиотеку Python, которая позволит нашему проекту связываться с 40 входами/выходами общего назначения Raspberry Pi, обозначаемыми GPIO. Последней импортируется библиотека *pygame*, ради воспроизведения музыки и отображения картинок на экране.

Для использования библиотеки *pygame* сначала нужно инициализировать ее и функцию микшера аудио, это выполняется двумя следующими строками кода:

```
pygame.init()
pygame.mixer.init()
```

Теперь переключим внимание на подготовку GPIO, и для начала сообщим Pi, какую из схем нумерации контактов мы будем использовать. Популярных схем две:





Функции игры

Чтобы начать объявление функции, мы указываем ее имя и затем добавляем аргумент в скобках. В данном случае аргумент — это количество оставшихся жизней:

```
def life_counter(i):
```

Указав имя функции, нажмем Enter, и весь последующий код будет оформлен с отступом в четыре пробела. Таким способом Python показывает, что код принадлежит функции. Мы создадим оператор условия `if..elseif`, который сравнивает количество жизней, переданное в аргументе, со значениями 3, 2, 1 и 0, и будем зажигать соответствующее количество светодиодов с помощью функции `GPIO.output`, которая изменяет состояние контакта с 0 («выключено») на 1. Наше первое условие — сравнение количества жизней с тройкой, и если их три, зажжем все светодиоды:

```
if i == 3:
    GPIO.output(14,1)
    GPIO.output(15,1)
    GPIO.output(18,1)
```

Если первое условие ложно, сравним количество жизней с двойкой с помощью условия `else if`, в Python это `elif`. В таком случае мы увидим, что загорятся два светодиода, 13 и 15, а светодиод 18 погаснет, то есть у нас осталось две жизни.

```
elif i == 2:
    GPIO.output(14,1)
    GPIO.output(15,1)
    GPIO.output(18,0)
```

Затем повторим выражение `else if` для единицы и нуля, выключая соответствующее количество светодиодов в каждом случае. На том функция и заканчивается. Перейдем ко второй функции — она применяется для вывода изображений с помощью библиотеки `pygame`.

```
def picture(img,w,h):
```

Мы начинаем функцию с указания ее имени (`picture`) и затем создаем три аргумента в скобках. Первый, `img`, используется для передачи имени файла изображения; `w` и `h` используются для передачи ширины и высоты изображения.

Создав шапку функции, нажмем Enter, и последующий код снова будет отделен отступом в четыре пробела. Затем создадим две переменных (ниже) — первая из них поименована `pic`:

```
pic = pygame.image.load(img)
background = (255, 64, 64)
screen = pygame.display.set_mode((w,h))
```

```
screen.fill((background))
```

С помощью переменной `pic` мы сократим себе работу по набору длинного имени функции загрузки изображения. Вторая созданная переменная, `screen`, служит для установки размеров окна с изображениями. Далее мы создаем кортеж. Кортеж — это неизменяемый список. После создания переделать его никак нельзя. Наш кортеж называется `background`, и в нем мы храним значения, описывающие цвет фона нашего окна.

В последней части этой функции мы сбрасываем изображение на экран:

```
screen.blit(pic,(0,0))
pygame.display.flip()
sleep(2)
pygame.display.quit()
```

Это быстрый метод отображения пикселей [blitting], часто используемый в 2D-видеоиграх. Затем мы обновляем всё окно с помощью функции `flip`, ждем две секунды и закрываем окно.

Проверка и переключение

Перейдем к основной функции проекта, начав с конструкции `try...except`, для чего добавим новую строку с выражением `try`. Она, как ей и следует, пробует выполнить указанный код, но если возникнут исключения — а в этом проекте мы попозже воспользуемся исключением `Keyboard Interrupt` [Прерывание с клавиатуры] — выполнение кода будет остановлено, и проект завершится.

В трех следующих строках мы создаем переменную с именем `toggle` и устанавливаем ее в `False`:

```
toggle = False
picture('./pi.jpg',640,771)
while True:
```

С ее помощью мы будем определять, хочет ли пользователь играть в игру. Затем вызывается функция `picture`, созданная нами ранее, и ей передаются изображение, которое мы хотим увидеть (имя файла), и его размеры. И мы начинаем наш бесконечный цикл.

Следующий фрагмент кода — условие проверки нажатия кнопки запуска игры:

```
if GPIO.input(21) == False:
    lives = 3
    pygame.mixer.music.load('./fanfare.mp3')
    pygame.mixer.music.play(1)
```

В предыдущем фрагменте мы запустили бесконечный цикл, и в нем постоянно отслеживается нажатие кнопки, проверкой ее состояния. Наша кнопка подключена к контакту 21 (`GPIO.input(21)`), одному из четырех входных контактов, которые вначале мы установили в «единицу». Когда игрок нажимает кнопку, состояние «контакта» изменяется с «единицы» на «ноль», так как другая сторона кнопки соприкасается с «землей», выключая контакт. Выполняемый после этого код устанавливает количество жизней игрока в три; затем открывается и воспроизводится звуковой файл, после чего переменная `toggle` устанавливается в `True` (`toggle = True`).

Теперь создадим вложенную еще на один уровень часть кода, которая постоянно проверяет, равно ли `True` значение переменной `toggle`:

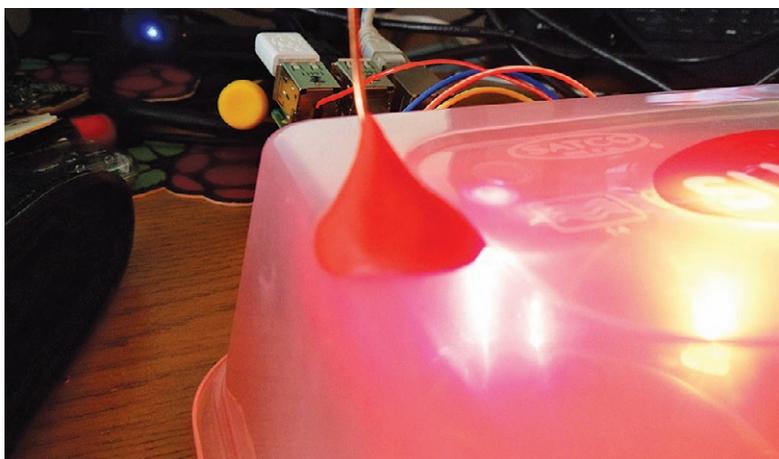
```
while True:
    if toggle == True:
        print("У вас осталось "+str(lives)+" жизней")
        life_counter(lives)
```

Тогда в оболочку Python выводится количество жизней, с которого начинается игра, а ранее созданная нами функция `life_counter` зажигает соответствующее количество светодиодов.

В следующей части кода мы, всё еще внутри созданного бесконечного цикла, создаем новое условие `if`, на сей раз отслеживая

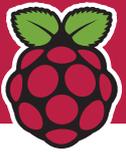
Скорая помощь

Мы создали репозиторий Github для всего кода и файлов мультимедиа, используемых в этом проекте. В нем также можно найти набор схем электрических цепей в высоком разрешении. Все это загружается с <http://bit.ly/LXF200WireGameZIP>.



➤ Очень важно прочно приделать проволоку, ведь сигнал сработает при малейшем касании. Для закрепления проволоки, изоляции проволоки и места для размещения щупа мы применили сугру — резину, которая на воздухе отвердевает.

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



изменение состояния контакта 2: это входной контакт, подключенный к щупу (скрепке):

```
if GPIO.input(2) == False:
    GPIO.output(24,1)
    sleep(0.2)
    GPIO.output(24,0)
    pygame.mixer.music.load('./wrong.mp3')
    pygame.mixer.music.play(1)
```

Если состояние изменяется на False («ноль»), то зуммер, подключенный к контакту 24, будет издавать звуковой сигнал в течение 0,2 с, затем откроется и воспроизведется звуковой файл — это будет означать, что игрок коснулся проволоки.

Elif в if

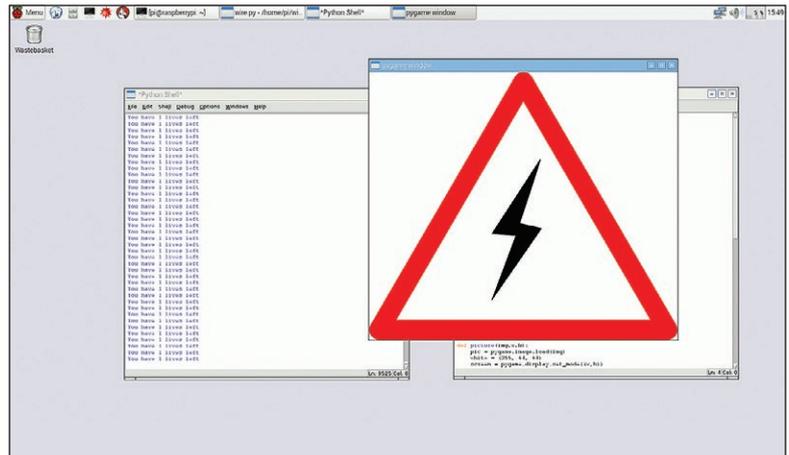
В последней порции кода в операторе if мы выведем изображение на экран, снова употребив ранее созданную функцию picture().

```
picture('shockdanger.jpg',724,634)
lives = lives - 1
life_counter(lives)
print("You have "+str(lives)+" lives left")
sleep(3)
```

Затем мы вычтем одну жизнь из переменной lives (lives = lives - 1), отобразим количество жизней светодиодами (life_counter(lives)), выведем количество жизней на экран и в течение трех секунд будем ждать возвращения игрока к началу. Теперь перейдем к блоку elif этого выражения if...elif:

```
elif lives == 0:
    pygame.mixer.music.load('./wrong.mp3')
    pygame.mixer.music.play(2)
    print("GAME OVER")
    sleep(3)
    break
```

Elif совершает действия в том случае, если количество жизней равно нулю. В нем три раза воспроизводится звуковой файл... но вы видите вызов функции pygame.mixer.music.play(2), почему же файл воспроизводится три раза? А потому, что Python начинает отсчет с нуля! Пока играет музыка, на экране будет надпись 'GAME OVER', затем после трехсекундной задержки (sleep(3)) бесконечный цикл завершится, и мы вернемся в главный бесконечный цикл, где мы ждем, пока игрок нажмет кнопку.



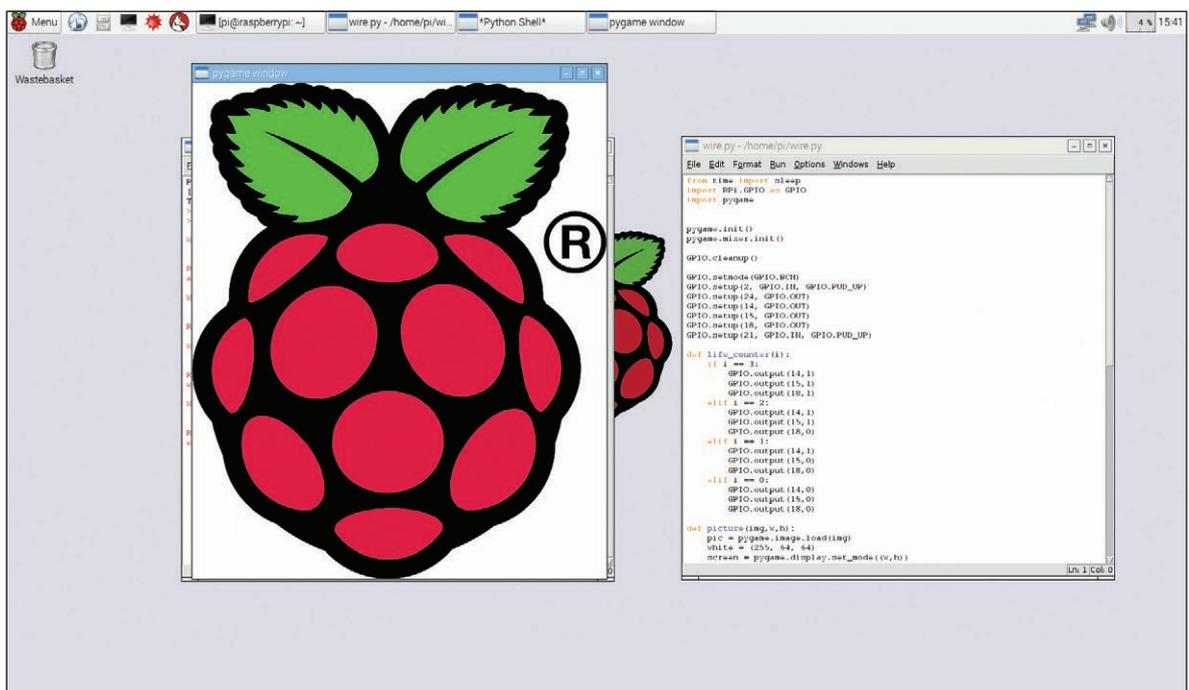
► При касании проволоки зуммер издает сигнал, воспроизводится аудиофайл, а на экране отображается предупреждающий знак.

В последней части кода мы закрываем выражение if, которое открыли для проверки состояния контакта 21, нашей кнопки для запуска игры:

```
else:
    print("Нажмите кнопку запуска")
except KeyboardInterrupt:
    GPIO.cleanup()
    print("EXIT")
```

Если кнопка не нажата, оболочка Python попросит пользователя нажать кнопку для запуска игры (print("Нажмите кнопку запуска")). Следующая строка — это исключение в нашей конструкции try...except. Она позволит игроку нажать CTRL+C для окончания игры, перевода контактов GPIO в их обычное состояние и выхода из проекта.

Покончив с кодом игры, не забудьте сохранить свою работу, а затем запустите ее, выбрав пункт меню Run > Run Module [Запуск > Запустить модуль]. На экране должен появиться логотип Raspberry Pi. Нажмите кнопку, чтобы запустить игру, и будьте осторожны — не касайтесь проволоки, иначе вы потеряете жизнь. Если вы не слышите звука из динамика, обратитесь к врезке про *raspi-config* [см. стр. 93] для подробной настройки *raspi-config*. [LXF](#)



► Чтобы показать готовность начать игру, мы отображаем на экране логотип Raspberry Pi. Это удобный способ взаимодействия с пользователем, да и выглядит, мы думаем, неплохо.

ОТВЕТЫ

Есть вопрос по открытому ПО? Пишите нам по адресу answers@linuxformat.ru, и мы найдем ответ.

В этом месяце мы ответим на вопросы про...

- 1 Обновление SQL- и web-серверов
- 2 Новые ядра и драйверы графического сопроцессора
- 3 Начала работы с Linux
- 4 Таймеры Systemd и задания Cronjob
- 5 Домашнюю сеть
- 6 Досаду от Firefox

1 Сохранение базы

Когда вы будете читать это письмо, Ubuntu 15.04 уже выйдет. Хотя я предпочитаю обновлять дистрибутивы сразу же, сейчас у меня большая проблема — я пытаюсь найти последние инструкции по переносу стека LAMP и базы данных MySQL в новую систему. Эти инструкции, похоже, меняются при каждом обновлении, и если у вас есть база данных, то каждое обновление просто отвлекает от всего остального, и думаю, в этот раз я без него обойдусь. Зачем так усложнять жизнь?

Майк Хиггинс [Mike Higgins]

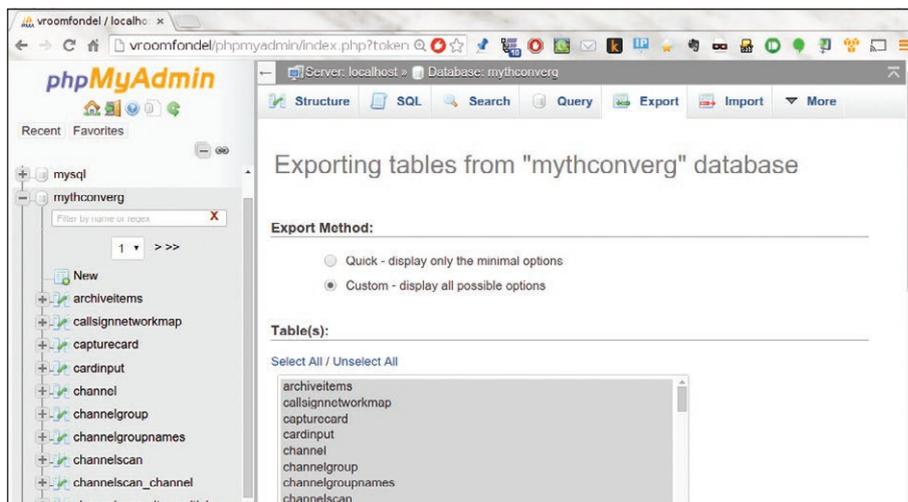
С помощью утилиты `mysqldump` легко сделать копию базы данных MySQL и потом ее восстановить. Через `phpMyAdmin` сделать это еще проще. Однако для сервера есть еще один важный момент. Мы часто советуем размещать `/home` на отдельном разделе, чтобы пользовательские файлы оставались нетронутыми при переустановке или обновлении системы. Для стека LAMP домашние каталоги различных программ расположены в `/var`. А значит, отдельная файловая система для `/var` в этом случае имеет большой смысл. Правда, нам следует позаботиться о файлах настроек в `/etc`, но они невелики и легко копируются; для стека LAMP надо сохранить копии каталогов `/etc/apache` и `/etc/mysql` (это если А в LAMP в Вашем случае означает Apache, а не другой web-сервер). Для резервного копирования базы данных с помощью `mysqldump` можно воспользоваться следующей командой:

```
$ mysqldump -p --all-databases --add-drop-database --create-options --opt --allow-keywords --flush-logs --hex-blob --max_allowed_packet=16M --result-file=mydatabases.sql
```

Она сформирует файл с SQL-запросами, передав который обратно MySQL, можно воссоздать свои базы данных:

```
$ mysql -p <mydatabases.sql
```

Файлы SQL-запросов бывают немаленькими, поэтому можно опустить параметр `--result-file` и пропустить вывод команды через `gzip`:



Создайте копию базы данных MySQL с графическим интерфейсом с помощью `phpMyAdmin`.

```
$ mysqldump -p --various-options | gzip > mydatabases.sql.gz
```

```
$ zcat mydatabases.sql.gz | mysql -p
```

Перед переустановкой системы запустите `mysqldump`, а затем скопируйте файл в безопасное место, чтобы потом в новой системе импортировать из него свои базы данных. Также понадобится создать резервную копию содержимого `/var/www`.

Альтернативный вариант — воспользоваться параметром `dist-upgrade` в `apt-get` для обновления дистрибутива до последней версии без переустановки. Эта команда обновит все пакеты и версию дистрибутива до последней версии и настроит в качестве источников репозитории для последнего релиза. Откройте терминал и выполните команду

```
$ sudo apt-get update
```

```
$ sudo apt-get dist-upgrade
```

Первая команда просто информирует Вашу систему о последних доступных пакетах, вторая выполняет само обновление. Если Вы хотите увидеть, что произойдет, не выполняя фактического обновления, добавьте к команде параметр пробного прогона — `--dry-run`:

```
$ sudo apt-get --dry-run dist-upgrade
```

Тогда Ваши данные и настройки должны остаться нетронутыми; хотя резервная копия никогда не будет лишней, обновляетесь Вы или нет.

2 Черный экран пустоты

Я обновил свою SolydX (с диска LXF) на EeePC, включая обновление ядра с версии 3.13-1-486 на 3.16.0-4-586. Это

дистрибутив на базе Debian, с оконным менеджером LXDE. После появления нескольких ошибок, связанных с подписями, я установил подписи Debian, как рекомендовалось на одном из справочных форумов. Теперь система загружается нормально, но в момент, когда должна появиться заставка входа в систему, я вижу только черный экран. Я могу загрузиться в предыдущую версию, и в ней все хорошо. Хотел откатиться к предыдущей версии ядра, но на map-страницах `apt` не смог найти ничего подходящего.

Грег [Greg]

Пакеты ядра отличаются от большинства других тем, что при установке новой версии старая не удаляется. Вы уже обнаружили, что старое ядро сохранилось, и нужно лишь сделать так, чтобы оно загружалось по умолчанию. Заглянув в файл `/etc/default/grub`, Вы найдете параметр `GRUB_DEFAULT`, в котором задается элемент меню, загружаемый по умолчанию. Сейчас он установлен в 0, так как по умолчанию `Grub` отсчитывает пункты меню с нуля. Поэтому для загрузки третьего пункта меню нужно установить `GRUB_DEFAULT=2`. Затем запустите `grub-update`, чтобы обновить меню для новых настроек.

Однако лучше попробовать решить проблему с графикой, не касаясь нового ядра. При появлении черного экрана нажмите `Ctrl+Alt+F1` (или подождите несколько минут), и откроется новая консоль. Войдите в систему как обычный пользователь, после чего Вы сможете заглянуть в лог-файл `X /var/log/Xorg.0.log`. Там надо искать строки, начинающиеся с [EE]. Скопируйте файл

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`.

в свой домашний каталог, после чего Вы сможете загрузить старое ядро, посмотреть в старый лог-файл и поискать решение в Сети. Обычно важное всего первая строка ошибки, остальные могут быть лишь следствием первой.

Наиболее вероятная причина — в том, что драйверы Вашей видеокарты не работают с новым ядром, и их нужно переустановить; это особенно характерно для внешних драйверов, таких как официальные драйверы Nvidia. Перед переустановкой любых драйверов убедитесь, что символическая ссылка в `/usr/src/linux` указывает на новое ядро, так как с ее помощью пакеты драйверов часто определяют версию ядра, под которую устанавливаются драйверы.

3 Проблема новичка

В Мне уже за восемьдесят, и серые клеточки дают об этом знать. Сейчас я увяз в последней истории с помешательством MS. Меня обеспокоило заявление о том, что Windows XP выходит в тираж, и я купил 15,6-дюймовый ноутбук Toshiba с Windows 8. С тех пор я несколько раз пытался им воспользоваться, и понял, что единственный способ, с помощью которого я могу улучшить Windows 8.1 — это удалить ее.

Я хочу установить Linux, но перед этим удалить Windows 8.1 и начать с чистого компьютера. Во всех руководствах по Linux сначала советуют сделать резервную копию файлов. У меня шестисантиметровая стопка DVD-дисков, и я готов приступить, но зачем это-то копировать, если копировать нечего? Или есть что? Я придаю этому большой значение, потому что один мой друг, у которого обновленный компьютер с Windows 7 Professional, решил попробовать установить Linux с одного из дисков, и теперь компьютер у него вообще не загружается!

Артур Лоуренс (Arthur Lawrence)

О Совет делать резервную копию файлов предполагает, что на компьютере есть данные, которые Вы хотите сохранить. Поэтому сохраните все документы, фотографии и другие необходимые файлы на DVD или USB-диске. При установке системы обычно не возникает проблем с загрузчиком, но даже если они произойдут, беспокоиться не о чем, так как на компьютере все равно не было ничего для Вас ценного. Единственная проблема, с которой Вы можете столкнуться в Windows 8 — это безопасная загрузка [Secure Boot].

Простое решение — зайти в меню загрузки (в руководстве пользователя написано, какую клавишу для этого нажать) — и выключить безопасную загрузку в разделе Boot [Загрузка].

Сделав это, можете выбрать дистрибутив Linux, который Вам подходит, и загрузиться с его DVD. Когда на экране появится вопрос об использовании диска, выберите использование диска полностью; в этом случае перед установкой новой системы Windows будет полностью удалена. Многие ноутбуки сейчас продаются без дисков с Windows, но с возможностью создать диск восстановления. Я бы создал такой диск перед удалением Windows на случай, если в будущем Вы захотите восстановить Windows, чтобы не лишиться возможности вернуть ноутбук по гарантии.

Остается только выбрать дистрибутив. Большинство популярных дистрибутивов — Ubuntu, openSUSE, Fedora, Linux Mint, Debian — подходят для новичков, но все равно потребуют некоторого обучения: Linux отличается от Windows не одним только названием. Если у Вас есть друг, который пользуется Linux, Вам лучше выбрать тот дистрибутив, который у него (да и для него так будет лучше), чтобы при необходимости Вы смогли поинтересоваться/надоеть ему своими вопросами.

4 Анахронизмы Cron

В Я недавно обновился до Ubuntu 15.04 и пытаюсь найти общий язык с Systemd. Среди прочего меня смущают таймеры, которые используются в Systemd для замены Cron. У меня есть несколько задач, которые запускаются с Cron. Нужно ли переносить их из Cron в Systemd? Если да, то как?

Джордж Уокер (George Walker)

О Во-первых, переставать пользоваться Cron совершенно не обязательно. Он работает точно так же, как и всегда, и перенастраивать его не нужно. Таймеры обладают большими возможностями по сравнению со стандартными записями Cron, зато и использовать их сложнее. Вместо одного файла, как в Cron, для каждого задания здесь придется определять два



Коротко про...

D-bus

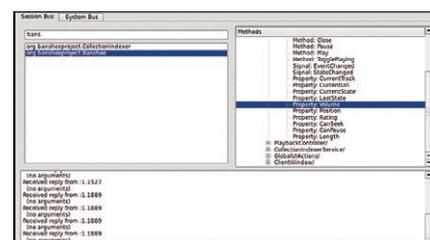
D-Bus, или Desktop Bus [настольная шина] — низкоуровневая система межпроцессного взаимодействия, используемая в Linux и других операционных системах POSIX. С ее помощью программы могут отправлять сообщения друг другу для запроса информации или обмена командами или запросами. До нее использовались по меньшей мере две несовместимые системы — DCOP в KDE и Вопо во в Gnome.

Архитектура DCOP в значительной степени повлияла на D-Bus, но D-Bus не зависит от рабочего стола. На самом деле, несмотря на название, она не используется вообще ни в одной

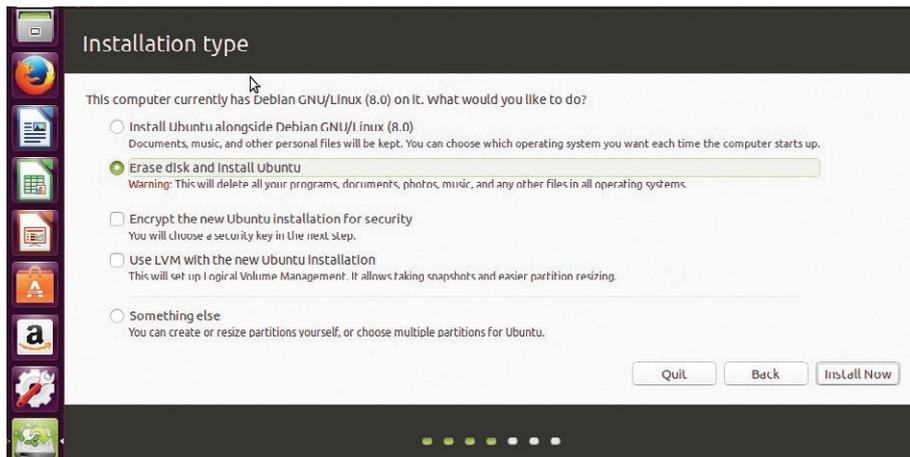
настольной системе. В настольных компьютерах в общем используются две шины — системная шина, используемая программами root, такими как демоны, ПО определения устройств и подобными им, и сессионная шина, используемая рабочим столом.

Возможности D-Bus гораздо шире, и многие программы, от программ определения устройств до системных оповещений, незаметно для пользователя пользуются ею. Команды для управления шиной из оболочки с длинными именами сервисов и путями до объектов могут показаться громоздкими, но это необходимо для того, чтобы все программы могли получить доступ к D-Bus без возникновения конфликтов. Простейший способ познакомиться с шиной — выполнять различные команды и наблюдать за результатом, просматривая дерево шины

утилитой `qdbusviewer`. Если вы хотите получить доступ к интерфейсу программы на D-Bus с помощью скрипта, самый простой вариант — `qdbus`. Также есть `dbus-send`, но она работает на более низком уровне, и чтобы сделать в ней что-то, придется сначала почитать man-страницу (и, пожалуй, немного поскрести в затылке).



➤ Секреты D-Bus раскрыты!



➤ Если при установке Linux вы хотите удалить Windows, большинство установщиков дает возможность очистить диск полностью и начать жизнь сначала.

отдельных файла. Это стандартные файлы модулей *Systemd*, которые надо будет сохранить в **/etc/systemd/system**. У первого файла — расширение **.timer**, и он описывает условия выполнения действия. Например, в приведенном ниже файле **ssdtrim.timer** дважды в неделю, по понедельникам и четвергам, запускается операция чистки SSD. Сначала файл таймера:

```
[Unit]
Description=Trims the root filesystem
[Timer]
OnCalendar=Mon,Thu *-*- * 10:00
[Install]
WantedBy=basic.target
```

Он выглядит почти так же как любой другой файл модуля, за исключением параметра **OnCalendar**. Там содержится строка вида

```
Weekday Year-Month-Date Hours:Minutes:Seconds
```

Поля **Weekday** и **Seconds** являются необязательными; кроме того, как и в *crontab*, отдельные поля могут представлять собой списки. Итак, этот таймер будет срабатывать в 10 утра каждый понедельник и четверг; но что он будет запускать? При срабатывании таймера он запускает сервис с одним и тем же именем (с помощью параметра **Unit** можно указать другое имя); и вот Вам сервис **ssd-trim.service**.

```
[Unit]
Description=Trims the root filesystem
[Service]
Type=oneshot
ExecStart=/sbin/fstrim /
```

Если опустить параметр **Type**, он примет значение "simple". Для большинства отдельных заданий *Cron* этот параметр не имеет значения. Модули таймера можно запускать и делать доступными точно так же, как и любые другие. Если у Вас в **/etc/cron** много задач, которые выполняются ежедневно, ежедневно и т.д., то создавать отдельные файлы таймеров и сервисов для каждой из них неудобно, поэтому создайте следующий файл **run-hourly.timer**:

```
[Unit]
Description=Hourly timer
[Timer]
OnBootSec=3min
```

```
OnUnitActiveSec=1h
OnCalendar=hourly
Unit=run-hourly.target
[Install]
WantedBy=basic.target
```

Параметр **OnBootSec** устанавливает задержку в три минуты от загрузки модуля до того, как он попытается что-либо запустить. Следующая строка означает, что модуль нужно запускать, только если с предыдущего запуска успело пройти не менее часа, чтобы избежать перезагрузки из-за избыточных запусков. Этот скрипт вызывает сервис **run-hourly.target**, который задан в параметре **Unit**:

```
[Unit]
Description=Hourly timer target
StopWhenUnneeded=yes
```

Затем создайте каталог **/etc/systemd/system/run-hourly.target.wants** и положите туда несколько файлов сервисов, каждый из которых будет запускаться раз в час. Сделайте то же самое для ежедневных и еженедельных заданий.

5 Основы сети

В у меня ноутбук с Debian с именем "Lenovo". Наверху у меня настольный компьютер с именем "Debian", также с Debian. Оба подключены по Wi-Fi к моему BT Home-Hub 4, и на обоих — рабочий стол Xfce. Я хочу сохранять документы с одного компьютера на другой. Фактически, я хочу, чтобы компьютер наверху был сервером. Сейчас оба компьютера не видят друг друга.

Rojj, С форумов

Сначала надо сделать так, чтобы два компьютера друг друга увидели. Запустите

```
$ ping -c 3 Debian
```

с ноутбука, и Вы должны увидеть несколько строк ответов на **ping**. Если Вы видите ошибку «неизвестный хост [unknown host]», попробуйте заменить имя компьютера на его IP-адрес, который можно получить от утилиты настройки сети на настольном компьютере или на странице состояния роутера. Если компьютеры видят друг друга по IP-адресу, значит, они не знают имен хостов

друг друга. Можно ограничиться и IP-адресами, но это не особенно удобно; поэтому откройте файл **/etc/hosts** с правами **root** в каждой из систем и добавьте строки

```
IP-ADDRESS Lenovo
IP-ADDRESS Debian
```

заменяв **IP-ADDRESS** IP-адресом каждого компьютера. Теперь компьютеры должны пинговаться по имени. Для доступа к файлам надо настроить сетевые ресурсы. Здесь есть два основных варианта — **NFS** и **Samba**. **NFS** — традиционные сетевые файловые ресурсы UNIX, тогда как **Samba** использует файловую систему **CIFS** Windows. У каждого варианта есть свои «за» и «против», но **Samba** также совместима с сетевыми ресурсами Windows, и ее можно настраивать через web-интерфейс. **Samba** можно установить на сервер и затем открыть в браузере адрес <http://localhost:901> для ее настройки.

Для настройки **NFS** используется текстовый файл **/etc/exports**, содержащий по строке на каждый ресурс:

```
/mnt/music 192.168.1.0/24(no_root_squash,no_subtree_check,rw,sync,fsid=5)
```

Здесь каталог **/mnt/music** экспортируется всем хостам в сети **192.168.1.***; при необходимости измените эти настройки на свои. Другие параметры подробно разъясняются на map-странице *exports*, но их можно оставить без изменений, за исключением **fsid**. В более поздних версиях **NFS** у каждого ресурса должен быть уникальный идентификатор. Для применения изменений выполните команду

```
$ sudo exportfs -r
```

На стороне клиента ресурсы **NFS** и **Samba** можно просматривать в большинстве файловых менеджеров или добавить в **/etc/fstab** для постоянного монтирования.

```
Debian:/mnt/music /mnt/music nfs soft,noauto 0 0
//Debian/music /mnt/music cifs username=xxx,password=yyy,noauto 0 0
```

Первая строка монтирует ресурс **NFS**, вторая — ресурс **Samba**. В **NFS** в качестве источника указывается каталог ресурса, а в **Samba** у каждого ресурса есть имя, которым мы пользуемся. Параметр **noauto** означает, что ресурсы не должны монтироваться при загрузке системы, что вызвало бы задержки в том случае, если Вы не подключены к своей домашней сети.

6 Досада от Firefox

Firefox задумал свести меня с ума. Например, сейчас адресная строка решила пожить своей жизнью. Я щелкнул по длинному адресу, изменяю его в адресной строке, но когда нажимаю **Enter**, в строку подставляется предыдущий адрес. Поэтому я копирую адрес в текстовый редактор, меняю его и вставляю обратно в адресную строку. А вставляется опять-таки исходный адрес!

Я очищаю кэш за последний час и перезапускаю **Firefox**. То же самое. Точно так же, без предупреждения он изменяет размер окна, сдвигая его в сторону, когда я открываю настройки, и аналогичным образом меняет ширину вкладок, когда я что-то открываю и закрываю.

Несмотря на то, что я имел дело с разнообразными дополнениями, почти каждому виджету во всем браузере нужно индивидуально приказывать, чтобы он перестал морочить мне голову. Нет ли какой-нибудь ручной гранаты, которой можно было бы запустить в этот беспорядок? **guu, с форумов**

Когда программа начинает вести себя подобным образом, первым делом стоит сбросить ее в настройки по умолчанию. Настройки *Firefox* хранятся в каталоге `~/.mozilla/firefox`, файл `profiles.ini` содержит строку вида «Путь=случайная_строка.default», а каталог `случайная_строка.default` содержит все Ваши настройки, закладки и прочие данные. Если переименовать файл `profile.ini` и запустить *Firefox*, он создаст новый файл `profile.ini` и новый каталог `другая_случайная_строка.default`, где все настройки будут установлены в значения по умолчанию. Это должно решить проблему, но тогда Вы потеряете все свои настройки и закладки. Теперь можете начинать копировать файлы по несколько штук из старого каталога с настройками в новый. Во время копирования *Firefox* не должен быть запущен. Скопировав очередную порцию файлов, запускайте браузер и проверяйте, все ли работает. Так методом исключения можно обнаружить причину почти любой проблемы в *Firefox*.

Альтернативным подходом будет сделать резервную копию каталога `случайная_строка.default` и поэкспериментировать со значениями настроек *Firefox*.

Для просмотра всех настроек наберите в адресной строке `about:config`. С помощью поисковой строки ограничьте настройки теми, в которых содержится `'urlbar'`: это настройки, определяющие поведение адресной строки. Настройки можно менять произвольно или сравнивая текущие значения с настройками по умолчанию (делая скриншоты настроек и сравнивая их). Для изменения любого параметра дважды щелкните по нему — при этом логические параметры (`true/false`, да/нет) будут переключаться в противоположные значения, а для параметров других типов откроется поле ввода.

Мы уже не в первый раз слышим о том, что настройки *Firefox* начинают жить своей жизнью, поэтому имеет смысл делать их резервные копии — но не всего каталога, который может стать очень большим, а хотя бы файла `prefs.js` и, возможно, закладок.

Также можно попробовать запустить *Firefox* без расширений, чтобы сразу же определить, не в них ли кроется проблема. Для этого послужит команда

```
$ firefox-bin --safe-mode
```

Запустите ее в терминале. **LXF**

Помогите нам помочь вам

Ежемесячно мы получаем несколько писем, на которые не в состоянии ответить, поскольку проблема описана в них недостаточно полно. Чтобы дать вам наилучший ответ, нам необходимо знать как можно больше.

Если у вас появляется сообщение об ошибке, приведите его точный текст и опишите конкретные условия, когда оно появляется. При возникновении проблемы с устройствами перечислите нам все установленные устройства.

Если Linux уже запущен, можете применить для этого отличную программу *Hardinfo* (<http://sourceforge.net/hardinfo.berlios>) — она сохранит подробную информацию об устройствах и о состоянии системы в HTML-файле, который вы сможете приложить к своему письму.

Не уступающий в удобстве альтернативный вариант — *lshw* (<http://ezix.org/project/wiki/Hardware-LiSter>). Одна из указанных программ непременно должна быть включена в ваш дистрибутив (а иногда и обе).

Если вы не хотите или не можете их установить, выполните следующие команды в терминале от имени root и приложите файл `system.txt` к письму. Это здорово поможет диагностике.

```
uname -a >>system.txt
lspci >>system.txt
lspci -vv >>system.txt
```



Часто задаваемые вопросы

Rsync и Unison

Что это за программа *rsync*, о которой все болтают?

Это способ синхронизации содержимого двух каталогов, гарантирующий, что один каталог будет точной копией другого.

А ср разве не годится?

Годится, но **ср** копирует все. *Rsync* копирует только те файлы, которые отличаются. Если изменились большие файлы, копируются только измененные фрагменты.

Как ею воспользоваться?

После выполнения команды `rsync --archive --delete /path/to/source/ /path/to/dest/` второй каталог будет содержать точную копию первого. Параметр `--delete` удаляет файлы, которых нет в первом каталоге, а `--archive` копируются все права доступа и временные отметки файлов. С *Rsync* важно указывать завершающие слэши; они показывают,

что вы хотите синхронизировать содержимое каталогов. Если их опустить, это может привести к тому, что один каталог скопируется в другой, и вы рискуете потерей данных.

А если я хочу синхронизироваться с каталогом на другом компьютере?

Если у вас есть SSH-доступ к другому компьютеру, можете воспользоваться командой

```
rsync --archive --delete /my/local/site/ hostname:/path/to/site/
```

Когда в одном или в обоих путях указано имя хоста, *Rsync* использует для передачи файлов удаленную оболочку. По умолчанию это SSH, но ее можно изменить на что-то другое — даже на RSH, если вы хотите, чтобы во время передачи ваши файлы мог прочесть кто ни попадя!

А что если я изменил файлы на двух компьютерах и хочу их синхронизировать?

Нужно нечто другое, более приспособленное для этой задачи, например, *Unison* (www.cis.upenn.edu/~bcpierce/unison). Программа использует *Rsync* и SSH для передачи файлов, поэтому все их преимущества по-прежнему в вашем распоряжении, но она предназначена для двунаправленной синхронизации. Если вы изменили какие-то файлы на одном компьютере и какие-то на другом, она гарантирует, что на каждом компьютере будет последняя версия файлов.

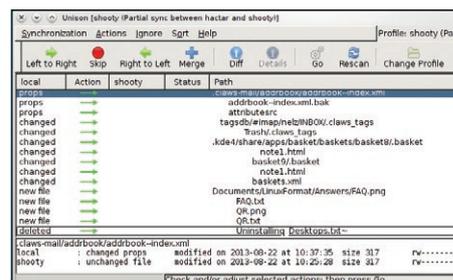
Умно придумано; ну, а если я изменил один и тот же файл на обоих компьютерах?

Программа не может прочесть ваши мысли и узнать, какую версию сохранить, но может сказать, что изменены обе копии, и спросить, какой воспользоваться. *Unison* отслеживает все изменения, поэтому будет знать, что

файл был изменен на обоих компьютерах с момента ее последнего запуска.

Нужно ли учить десятки параметров командной строки?

Можно воспользоваться параметрами командной строки или выбрать файлы через графический интерфейс. Графический интерфейс запускается только на том компьютере, который инициировал передачу данных, поэтому с его помощью можно синхронизироваться с удаленным компьютером, если вы предварительно установите там *Unison*.



Синхронизация ноутбука и настольного компьютера с *Unison*.



LXF HotPicks



Александр Толстой

Собрал самые спелые приложения в обширном саду Интернета, чтобы испечь для вас сочный пирог FOSS с хрустящей корочкой и кремом.

qBittorrent » uGet » Tmux » Qt5CT » Krita » Scribus » Kid3 » Tanks of Freedom » Dust Racing 2D » Lollypop » CVAssistant

Клиент Bittorrent

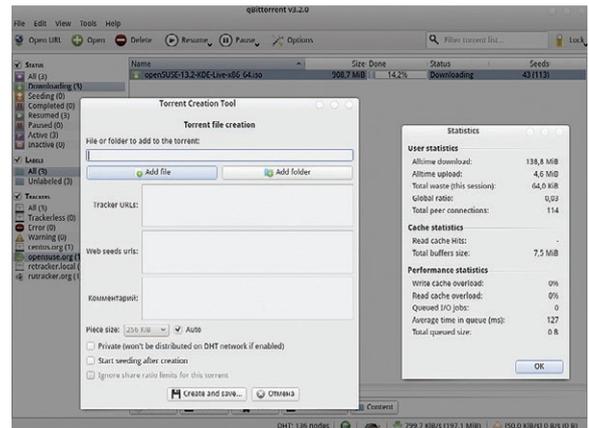
qBittorrent

Версия: 3.2.0 Сайт: www.qbittorrent.org

Клиент BitTorrent с открытым кодом, *qBittorrent*, доступен для всех основных операционных систем, и особенно для Linux, где *qBittorrent* позиционируется как ближайший коллега проприетарного *µTorrent*.

У него классический интерфейс пользователя с текущими загрузками в основном окне и функциональной боковой панелью с отображением состояния, меток и списка трекеров (это недавно появившаяся функция). Простейший способ использовать *qBittorrent* — предложить ему файл *.torrent* посредством выбора скачанного локального файла или посредством вставки URL, и оба этих действия можно

выполнить из панели инструментов приложения. Однако *qBittorrent* предлагает множество функций для опытных пользователей. Например, он поддерживает почти все расширения BitTorrent, включая ссылки DHT и Magnet/BitComet, адреса IPv6, переадресацию uPNP и NAT-PMP (если вы используете роутер), фильтры IP, основанные на Ajax web-интерфейсы, сортировку торрентов, постановку в очередь



» В *qBittorrent* есть все, что только может понадобиться поклоннику торрентов.

«Можно настроить его на потоковое скачивание.»

и многое другое. Удобная функция поиска по торренту оказывается весьма кстати, поскольку она позволяет делать запросы по множеству поддерживаемых трекеров внутри *qBittorrent*. Пользователи также могут сами создавать новые торренты благодаря встроенному мастеру (который находится в меню Tools).

Новая версия 3.2 предлагает разнообразные отладки и улучшения GUI, который теперь можно компилировать с зависимостью от библиотеки *Qt5* (учтите: на скриншоте версия *Qt4*). Скачанные файлы теперь выделяются в *Dolphin*, *Konqueror* и *Nautilus*, и имеется также новый статус закачек 'Finished'. Есть еще и новый *.service*-файл для *Systemd* (для использования *qBittorrent* в режиме poX), который поддерживает скачивание файлов в сетевые расположения, и тонны разных отладок. Приложение отлично вписывается в любую среду рабочего стола благодаря своему оптимальному интерфейсу *Qt4/Qt5* и четкому дизайну. Его также можно настроить на потоковое скачивание вместо одновременного режима, что является большим преимуществом для пользователей, имеющих низкую скорость соединения.

Найти *qBittorrent* довольно легко — просто загляните в репозитории своего дистрибутива и отыщите нужный пакет. Есть пакеты Ubuntu и Mint в соответствующих PPA (qbittorrent-team/qbittorrent-stable). Чтобы найти сборки *Qt5* для ровной установки, можете также исследовать OBS для *openSUSE* или AUR для Arch.

Исследуем интерфейс qBittorrent

Основные инструменты управления

На панели инструментов располагаются часто используемые инструменты, применяемые к выделенному в данный момент торренту.

Боковая панель

Категории Status и Labels уже были здесь раньше, но недавно появился список трекеров.

Основная область

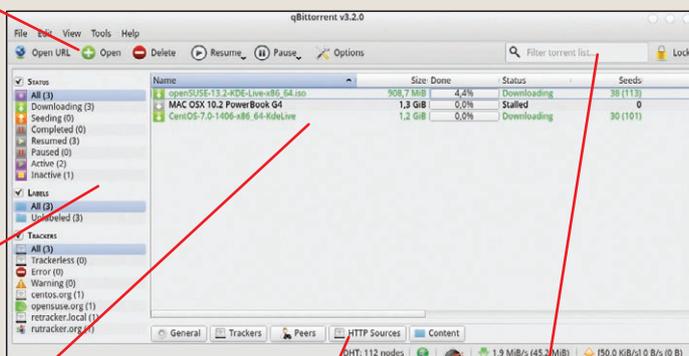
Mission Control Centre выводит всю необходимую для отслеживания закачек информацию.

Добавочные функции

Нижняя панель предлагает индикаторы и полезные кнопки для изучения сетевого соединения и контента торрентов.

Поле поиска

Если у вас одновременно работает несколько торрентов, вы можете отфильтровать их с помощью ключевых слов и указания категорий.



Менеджер загрузок

uGet

Версия: 2.0 Сайт: <http://ugetdm.com>

Вот еще одно приложение, которое служит отличным клиентом загрузок. *uGet* — это кросс-платформенный менеджер загрузок с открытым кодом. По сравнению с *qBittorrent*, у него множество особенностей и упор совершенно на другое.

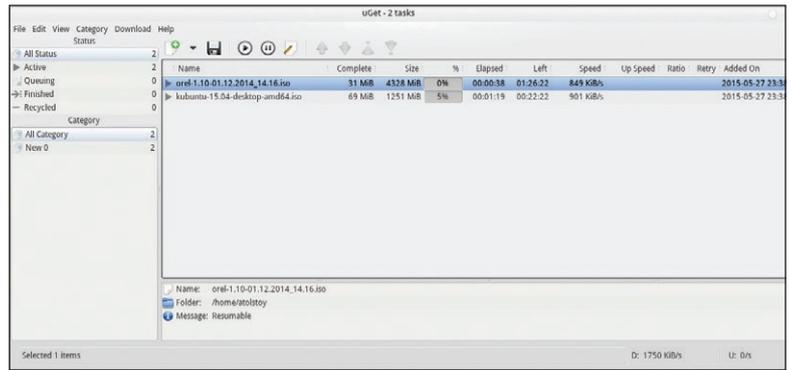
Если BitTorrent является отдельной технологией, объединяющей технологические и социальные элементы, то обычный менеджер загрузки в меньшей степени социален и в большей технологичен. До эпохи BitTorrent — когда широкополосное соединение с Интернетом было еще в младенческом возрасте и пользователи Linux возились с `/etc/ppp/options` — менеджеры загрузок были исключительно актуальны для всех, кому надо было контролировать использование данных и у кого были особые временные фреймы с более низкой скоростью передачи данных. Сегодня эти проблемы по-прежнему актуальны для тех, кому приходится использовать платное мобильное соединение. И если

Управляйте областью ваших загрузок с помощью тэгов и устраняйте любой беспорядок.

вам надо часто скачивать много больших файлов, стандартный web-браузер будет не лучшим решением; специальное приложение лучше справится с этой работой.

uGet — легковесный менеджер загрузок на GTK3 с расширенными функциями. Он может помещать файлы в очередь загрузки, ставить на паузу и возобновлять загрузки, и имеет расширенное управление категориями. *uGet* также интегрируется с *Firefox* через расширение *FlashGot*, способен на мониторинг буфера обмена (он там

«*uGet* — продвинутое и зрелое приложение из мира GTK3.»



ищет URL) и массовые загрузки. Клиент поддерживает ряд протоколов, в том числе HTTP, HTTPS, FTP, BitTorrent и Metalink. Им можно управлять удаленно из командной строки или работать без уведомлений в Quiet Mode, использовать регистрационные данные Proxu, ограничить число соединений и т.д. Загрузки можно организовать по категориям, что весьма удобно для фильтрации, когда вам начинает казаться, что их стало слишком много.

uGet может стать хорошим менеджером загрузок по умолчанию для тех, кому нужно продвинутое и зрелое приложение из мира GTK3. Приложение также доступно во множестве дистрибутивов Linux и в других системах, включая FreeBSD и Android для мобильных устройств.

Мультиплексор терминала

Tmux

Версия: 2.0 Сайт: <http://tmux.sourceforge.net>

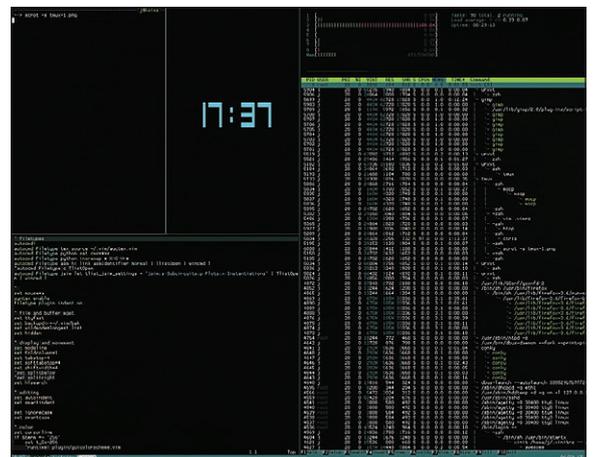
После года разработки недавно появился новый *Tmux 2.0*. Это мультиплексор терминала, разработанный сообществом OpenBSD в качестве более расширенного и гибкого замещения *Screen* [о котором рассказывалось в нашей статье «Побег из GUI», стр. 46 LXF197]. *Tmux* позволяет расщепить терминал на несколько экземпляров с целью создания многозадачной среды, что делает его идеальным для фанатов командной строки.

Терминалы можно создавать, получать к ним доступ и контролировать из одного экрана, чтобы каждый экземпляр *Tmux* мог быть отделен от экрана и продолжал работать в фоновом режиме, а затем снова присоединен. При запуске *Tmux* создает новую сессию с одним окном. Строка состояния внизу экрана показывает информацию по текущей сессии и используется для ввода интерактивных команд. *Tmux* может управляться из прилагаемого

клиента посредством клавиатурных комбинаций: префикс-ключ, по умолчанию это `Ctrl+b`, а за ним следует клавиша команды. Например, клавиша `%` делит текущую панель на две: левую и правую, а клавиши `0-9` выделяют окна от 0 до 9. Чтобы посмотреть весь список горячих клавиш, загляните в соответствующий раздел руководства с помощью `man tmux`.

Сессия является одной подборкой псевдотерминалов под управлением *Tmux*. Каждая сессия имеет одно или более окон, соединенных в ней. Окно занимает весь экран и может быть разделено на прямоугольные панели, каждая из которых является псевдотерминалом (страница руководства `pty(4)` объясняет все подробности

«*Tmux* позволяет разбить терминал на несколько экземпляров.»



Расширенный мультиплексор терминала помогает избавиться от GUI.

псевдотерминалов). С той же сессией может соединяться любое количество экземпляров *Tmux*, и в той же сессии может присутствовать любое количество окон. Когда все сессии прекращают работу, происходит выход из *Tmux*.

В версии 2.0 некоторые опции были вычищены или перенесены из сессии на сторону сервера (например, `'terminal-overrides'` или `'message-limit'`), плюс улучшена поддержка UTF-8. *Tmux* есть в десятках дистрибутивов Linux, так что найти его весьма просто, а вот научиться «правильной» настройке намного сложнее.

Инструмент настройки

Qt5CT

Версия: 0.12 Сайт: <http://qt5ct.sourceforge.net>

Когда среда GUI обновляется до основной версии, одним из обычных результатов является полный визуальный беспорядок. Мы помним переход Gnome с GTK2 на GTK3, и то, как KDE отказался от своей серии 3.x в 2008–2009, предложив совершенно новый KDE4.

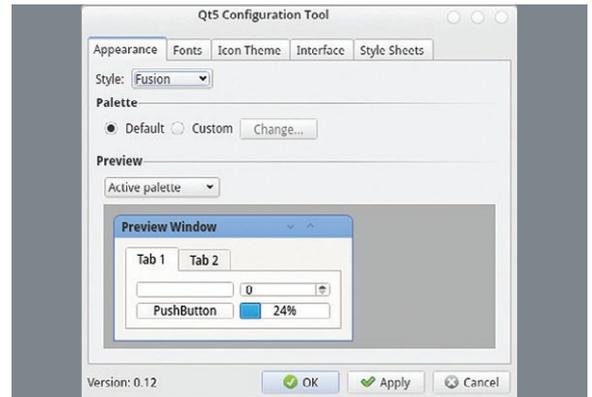
И вот пришло время для нового перехода: пока вы читаете этот материал, многие производители Linux и многие пакеты переходят на новый рабочий стол Plasma 5, пришедший на смену 4.x. Это также означает, что все больше и больше основанных на Qt приложений переходят на самую последнюю версию Qt5, и рано или поздно вы получите набор разных инструментов, которые изменят вид ваших приложений рабочего стола до неузнаваемости. Чтобы обойти эту проблему, команда Qt выпустила Qt5CT, инструмент, который позволяет настроить Qt5 (темы, значки, шрифты и т.д.) в среде рабочего стола/менеджере окон без интеграции с Qt. Qt5CT полезен не только в Gnome/Unity/Cinnamon

и иже с ними, но также и в KDE 4, который не слишком заботится о стиле Qt5.

В приложении имеется шесть вкладок для изменения внешнего вида виджетов, шрифтов, настройки тем значков, эффектов GUI и применения индивидуальных таблиц стилей. По умолчанию приложения Qt5 используют стиль Fusion плюс шрифты и цветовая гамма KDE4, но все это можно изменить. Кроме того, Qt5 принимает индивидуальные таблицы стилей; например, следующая таблица стилей указывает, что все QLineEdit в качестве цвета фона должны использовать желтый, а все QCheckBoxes в качестве цвета текста должны использовать красный:

```
QLineEdit { background: yellow }
QCheckBox { color: red }
```

«Настраивайте Qt5 в среде рабочего стола без интеграции с Qt.»



► Инструмент настройки Qt5 выглядит профессионально и предлагает богатый набор функций.

Поддерживаются почти все элементы стилей, и вы можете настраивать их и резвиться с ними как угодно, освоив синтаксис (см. <http://doc.qt.io/qt-stylesheet-syntax.html>). В некоторых системах Qt5CT может не оказывать никаких изменений, а это значит, что некоторые важные переменные их подавляют. Исправьте это, введя следующие строки в вашем файле `~/.bashrc`:

```
QT_STYLE_OVERRIDE=gtkexport
QT_QPA_PLATFORMTHEME=qt5ct
```

Qt5CT доступен в Arch AUR, ROSA и openSUSE, но, как ни странно, не в семействе Ubuntu, где предполагается его установка из исходника; а следовательно, надо установить пакеты разработки Qt5 и соответствующую среду компиляции.

Программа рисования

Krita

Версия: 2.9.4 Сайт: <https://krita.org>

Krita пребывает в тени не слишком известного офисного пакета Calligra (бывший KOffice); у большинства офисных пакетов флагманами являются мощный текстовый редактор и редактор электронных таблиц, но Krita — программа растровой графики, сосредоточенная на цифровом рисовании и живописи — самый мощный и самый профессиональный компонент Calligra (Krita по-шведски означает «мелок»).

Многие пользователи сравнивают Krita с GIMP и Adobe Photoshop, но это всегда было приложением для художников, чьей сильной стороной являются кисти и инструменты рисования, а набор эффектов и фильтров весьма скромнен. За последние годы разработка Krita весьма ускорилась, и были устранены многие проблемы, в частности, низкая производительность 2D и случайные сбои. Серия 2.9.x становится всё более впечатляющей с каждым новым релизом, и вот что мы получили на сей раз.

Среди самых ожидаемых и желанных функций в Krita 2.9.4 — поддержка стилей слоев, как в Photoshop. Да, теперь Krita может добавлять к своим слоям тени, внутренний/внешний отблеск, эффект выпуклости, выдавливание и множество всяких скевоморфных эффектов, и, конечно, она теперь намного лучше обрабатывает сторонние файлы PSD.

Прочие изменения в этой версии — по большей части, отладки и улучшения производительности. Теперь Krita запускается намного быстрее, не ожидая загрузки предварительных настроек (уменьшая время загрузки на несколько секунд), плюс к тому маски и фильтры работают лучше, как и некоторые инструменты,

«Среди желанных функций — поддержка слоев, как в Photoshop.»



► По умолчанию Krita намеренно использует темную тему, поскольку вы должны сконцентрироваться на содержании.

применяемые в эффектах наложения, что ведет к более ровной живописи.

Конечно, еще многое предстоит сделать; разработчики Krita отмечают, что на данный момент все функции приложения можно сохранять только в родном формате файлов Krita (пока что не в PSD). Некоторые ошибки в масках и слоях тоже будут исправлены в будущих вспомогательных релизах.

Найти Krita будет довольно легко, есть PPA для Ubuntu и прочих этого семейства ([ppa:dimula73/krita](http://ppa.dimula73/krita)), а большинство других дистрибутивов Linux предоставляют пакеты Krita вместе с остальными частями Calligra Suite.

Программа публикаций

Scribus

Версия: 1.5.0 Сайт: <http://www.scribus.net>

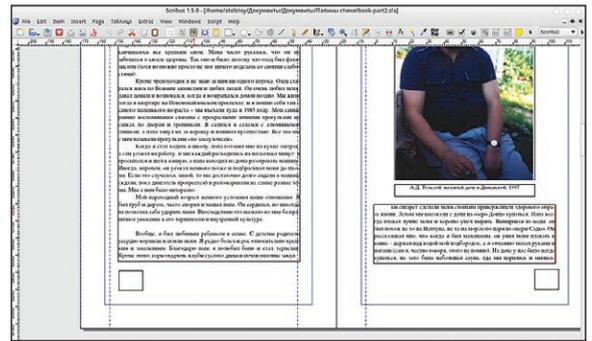
Разработка постеров, буклетов и тому подобных готовых печатных изделий всегда была областью, где Linux блистал долгие годы. Но и по сию пору приложений, способных выполнять эту работу на профессиональном уровне настольной издательской системы [DTP], не так уж много; хотя есть приложения, пригодные для школьных проектов и полупрофессиональных публикаций. С начала 2000 г. Scribus был самой продвинутой программой DTP — хотя он до сих пор не сравнялся с *Adobe Pagemaker* 1990-х, но во многом сокращает этот разрыв.

Scribus неоднократно попадал на страницы **LXF** [в последний раз — в версии 1.4.4, Обзоры, стр. 20 **LXF187**], но уже пора снять покров с новой версии 1.5, являющейся результатом многолетней работы и более тысячи отладок. Хотя разработчики заявляют, что *Scribus 1.5* пока недостаточно стабилен для использования в реальных производственных сценариях (хотя будущий релиз 1.6 планируется как

стабильный), в некритической домашней среде он работал очень неплохо, без каких-либо сбоев или ошибок.

Среди важнейших технических изменений — переход на современный набор инструментов *Qt5*, значительный пересмотр SLA, внутреннего формата файлов *Scribus*, и интеграция с *UniConverter* (рекомендуем установить его заранее) для импорта большего числа векторных форматов файлов. Пользовательский интерфейс тоже был улучшен благодаря пересмотру диалогового окна *Document Setup/Preferences*, и многие меню теперь занимают куда меньше места на экране.

Scribus 1.5 также научился встраивать растровую графику прямо внутрь своего формата SLA (раньше их можно было



► **Scribus наконец-то начинает не только выглядеть, но и работать, как профессиональный пакет DTP.**

только объединять ссылками), перемещать несколько объектов, не группируя их, и также клонировать свойства главного объекта в дочерние объекты. Все эти функции вполне обычны для многих качественных коммерческих программ векторной графики, хотя *Scribus* стремится брать пример с *Adobe Illustrator*.

Scribus доступен почти во всех дистрибутивах Linux, однако не все они предлагают самую последнюю версию. Если вы не можете найти прекомпилированную версию, следуйте инструкциям в *wiki Scribus*, чтобы скомпилировать из исходника. У *Scribus* превосходная документация и очень достойное руководство по компиляции для различных платформ, даже для таких нетривиальных, как *Haiku OS* (см. Сравнение, стр. 24).

«Важным изменением был переход на набор инструментов Qt5.»

Аудиоинструмент

Kid3

Версия: 3.2.1 Сайт: <http://kid3.sourceforge.net>

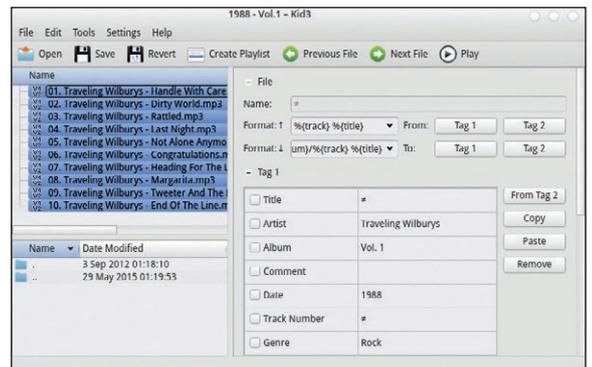
Некоторым пользователям подборка обширной музыкальной библиотеки на жестком диске может показаться устаревшей задачей в мире, где столь повсеместно используется потоковые медиа; однако это позволяет нам контролировать свою музыку.

Но по мере того, как растет ваша папка с музыкой, все труднее становится поддерживать структуру директории и соглашения о наименовании файлов. Файлы MP3 вместе с другими конкурирующими форматами, такими, как OGG или AAC, хранят метаданные трека внутри файла. Именно поэтому файлы в директории могут выглядеть правильными и не требующими дополнительного описания, но как только вы добавляете их в базу данных медиа-плеера, они резко становятся бессистемными.

Метаданные трека представляются в виде тэгов, ID3v1 (с фиксированным набором полей ограниченной длины) или

ID3v2 (которые являются полностью динамическими структурами, допускающими не только произвольный размер поля, но также и произвольный тип поля).

Kid3 работает с тэгами файлов, и приложение имеет обширную область слева для отображения загруженных аудиофайлов, мини-файловый менеджер в нижней левой части и редактор метаданных справа. Инструмент поддерживает массовое присвоение тэгов множеству файлов MP3, Ogg/Vorbis, Opus, DSF, FLAC, MPC, MP4/AAC, MP2, Opus, Speex, TrueAudio, WavPack и WMA (например, полные альбомы) без необходимости вводить одну и ту же информацию каждый раз. Это дополняется употреблением трафаретных символов



► **Реформируйте неорганизованную музыкальную библиотеку на своем жестком диске — используйте Kid3 для борьбы со своими метаданными.**

в полях *Format*, что позволяет определять источник и назначение структуры тэгов.

Попривыкнув к *Kid3*, вы сможете создавать тэги из имен файлов — и наоборот; автоматически конвертировать верхний и нижний регистр, заменять строки, создавать плейлисты и т.д. *Kid3* предлагает такие полезные функции, как импорт метаданных из *freedb2.org*, *MusicBrainz*, *Discogs*, *Amazon* и иных источников, редактирование синхронизированных слов и коды времени событий. Хотя номер версии *Kid3* не кажется особо большим, это приложение существует с 2003 г. и кроме официального сайта имеется в большинстве дистрибутивов Linux, а его страница на *SourceForge* предлагает пакеты *Ubuntu*.

«Вы сможете генерировать тэги из имен файлов — и наоборот.»

HotGames Развлекательные приложения

Пошаговая стратегия

Tanks of Freedom

Версия: Beta3 Сайт: <http://bit.ly/TanksOfFreedom>

Любители пиксельной графики и инди-игр, вероятно, сочтут *Tanks of Freedom* чудесным пошаговым шедевром. Суть игры заключается в том, чтобы продвигаться вперед с помощью своих танков, оказывать поддержку с воздуха своими вертолетами и захватывать стратегически важные здания своей пехотой.

Каждый уровень являет собой часть городского пейзажа с дорогами, зданиями и разными препятствиями. Каждый уровень предполагает битву между армиями двух игроков, состоящими из нескольких подразделений пехоты. Целью игры является захват командного пункта противника. Игрок может выбрать простую или сложную стратегию. Если у вас мало времени, в *Tanks of Freedom* можно играть, как в логическую настольную игру, где вы выбираете самый

быстрый и безопасный путь к командному пункту противника. Но куда веселее и интереснее применять танки и самолеты. Когда ваши юниты захватывают заводы (они бывают нескольких типов), можно потратить деньги на производство разнообразного тяжелого вооружения. Вертолет бьет танк, а три пехотинца бьют вертолет. У танков самый большой запас хода, если вы используете дороги с твердым покрытием. Игрок должен сбалансировать разные боевые элементы своей армии, потому что только пехота может занимать здания, а транспортные средства обеспечивают огневую мощь.

«Мы должны признать, что у Tanks for Freedom зрелый геймплей.»



➤ Никакого мародерства и гражданских беспорядков, а только старая добрая битва между профессионалами.

Мы должны признать, что, хотя *Tanks of Freedom* крошечная и достаточно простая игра, от нее невозможно оторваться, и у нее весьма зрелый геймплей, и она доставляет массу удовольствия. Игра распространяется в виде файла BIN, который вам придется сделать исполняемым и запустить из терминала вот так:

```
$ chmod +x ToF_Beta3.071_linux64.bin
$ ./ToF_Beta3.071_linux64.bin
```

Игра невероятно быстрая, ей нужно около 33 МБ на жестком диске, и она отлично будет работать на любой машине, способной использовать игры на SDL.

Симулятор гонок

Dust Racing 2D

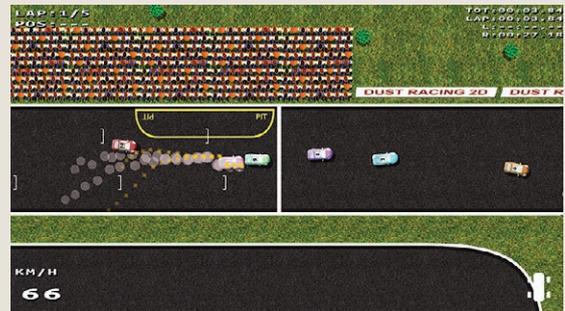
Версия: 1.11 Сайт: <http://bit.ly/DustRacing2D>

Существует множество гоночных симуляторов с видом от первого лица — например, мы ждем появления в этом году *Project CARS*; однако *Dust Racing 2D* — нечто совершенно иное. В игре очень красивые визуальные эффекты, однако они не претендуют на правдоподобие и выполнены в мультяшном стиле.

Игра предлагает аркадный набор уровней, из которых только первый доступен с самого начала. Чтобы разблокировать следующие, надо финишировать в первой шестерке. Так как же игре соответствовать? Управление игрой исключительно сложное, поскольку используется только четыре стрелки на клавиатуре (или другие клавиши, которые вы определили в *Settings*), к чему крайне сложно привыкнуть, когда вы участвуете в гонках с 11 ботами ИИ. Еще одна причина — вид сверху вниз: ваша клавиша ускорения

всегда остается той же самой (стрелка вверх по умолчанию), тогда как боковые клавиши ведут себя по-разному в зависимости от вашего положения на трассе и направления движения вашей машины. Вначале это может показаться сущим кошмаром, поскольку управление машиной очень скользкое, но через некоторое время вы дойдете до должной быстроты реакции, чтобы вписываться в повороты, не сходя с трассы. Вы не можете жульничать и срезать трассу, поскольку за это полагаются дополнительные штрафные круги. Натерев на пальцах мозоли, вы, вероятно, освоите искусство не врезаться в деревья и изящно вписываться

«Начинается в полном экране, играет музыку.»



➤ Нужно немало попыток, чтобы в пух и прах разнести ботов ИИ.

в крутые повороты, и с успехом перейдете на следующий уровень.

Dust Racing 2D сильно прогрессировала за прошлые годы. Она начинается в чудесном полноэкранном режиме, играет весьма прилипчивую музыку, предлагает больше логических настроек сложности, лучшие текстуры и больше не выдает сбоев (у нас их не было). Есть четыре режима игры: для одного или двух игроков (на одной клавиатуре), на время и дуэль. Счет кругов тоже можно изменить по своему желанию.

Игра доступна для разных дистрибутивов Linux, включая Mageia, Ubuntu, Debian, OpenSUSE, Arch и многих других. *Dust Racing 2D* использует пакет SDL и будет работать на любой системе с OpenGL.

Музыкальный плеер

Lollypop

Версия: 0.9.20 Сайт: <http://bit.ly/LollypopPlayer>

Нет, это не мобильная операционная система Android 5.0, а музыкальный плеер с таким же кондитерским названием [Lollypop — англ. леденец]. Он создан, чтобы использоваться в Gnome или другой среде на GTK3, со своей четкой структурой и большим пространством между значками. Lollypop использует части Totem для анализа метаданных трека, создания плей-листов и воспроизведения из разнообразных источников.

При первом запуске приложение автоматически добавляет вашу директорию home в свою базу данных и начинает исследовать ваш жесткий диск, так что приготовьтесь немного подождать, если ваша музыкальная коллекция довольно велика. Lollypop управляет вашей коллекцией, концентрируясь на динамически обновляемой статистике воспроизведения. На левой панели вы найдете набор категорий, таких, как Popular Albums, Recent Albums, Random Album, а также статические

All Albums, Radios и Playlists. Последние две предназначены для заполнения пользовательским контентом, так как Lollypop не предлагает предварительно настроенного набора радиосайтов. Кроме того, Lollypop с успехом употребляет современные функции GTK3, позволяющие буквально на лету менять его цветовую схему.

Здесь имеется также кнопка режима Party, которая гасит свет и окрашивает окно Lollypop в темно-серый цвет. Инструкции Gnome рекомендуют разработчикам использовать автопросмотр музыкальной библиотеки на фоне, но в Lollypop также имеется специальная кнопка Update music (которая находится в меню Tools, в дальнем правом значке на панели инструментов).

«Добавляет home в базу данных и начинает поиск по диску.»



Отличная идея — начать вечеринку единственным щелчком мыши.

Обложка альбома — это еще одна функция, которая потребует ручной работы; просто выберите альбом в любой соответствующей категории, а затем щелкните левой кнопкой по области его обложки. Lollypop отыщет рисунки обложки и даст вам возможность выбрать из нескольких вариантов.

Lollypop прост в использовании и не перегружен функциями и опциями, но уважает контроль пользователя за своей музыкальной коллекцией. Дайте ему шанс на своем стильном рабочем столе на основе GTK3, и, возможно, вы решите, что Lollypop вписывается туда лучше, чем Gnome Music и некоторые другие плееры.

Lollypop доступен в пакетах для Arch, Fedora, openSUSE и Ubuntu/Debian. Более подробную информацию вы найдете на сайте проекта.

Приложение для бизнеса

CVAssistant

Версия: 2.0.0 Сайт: <http://bit.ly/CVAssistant>

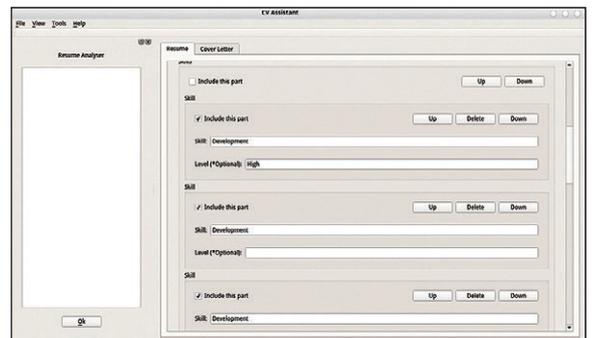
Ощущаете, что работа отправила вас в кокаут? Не сдавайтесь: это удобное приложение поможет справиться со всеми задачами. Не секрет, что большинству людей приходится поддерживать более одного профиля по профессиональным задачам в смежных областях, поскольку отделы кадров, похоже, забывают о том, что один человек вполне способен выполнять не только однотипные задачи. А значит, приходится готовить несколько видов резюме с разной информацией, чтобы лучше соответствовать разным рабочим профилям или определенной вакансии. CVAssistant — специальная программа, которая помогает создать специализированные резюме в формате Word DOCX с меньшими усилиями. Суть в том, чтобы получить образец резюме, где указаны все ваши навыки и опыт работы, и затем, основываясь на требованиях в объявлении о вакансии, экспортировать чистый и хорошо отформатированный файл DOCX

в качестве готового резюме, где указаны только необходимые навыки и опыт.

И это очень полезно: большинство компаний используют Applicant Tracking Software (ATS) для поиска соответствий между своими объявлениями о вакансиях и входящими резюме, так что, вернув нужные фразы в резюме, вы значительно повысите свои шансы попасть на собеседование.

CVAssistant — приложение на Qt, позволяющее создать свой образец резюме, переработать его поля, добавить свое фото и выбрать, какие пункты экспортировать в целевой документ. Самая полезная функция в CVAssistant — Analyzer: она сравнивает ваш образец резюме с ключевыми словами в объявлении о вакансии

«Спецпрограмма поможет создать специализированные резюме.»



Подойдите к поиску работы более профессионально.

и сообщает, соответствует ли им ваше резюме. Помимо основной вкладки Resume имеется также вспомогательная вкладка Cover Letter, где вы должны изложить свой привлекательный подход к работе.

Правильно заполнив все поля, нажмите Ctrl+E для экспорта резюме в формат DOCX, и Ctrl+Shift+E, чтобы экспортировать сопроводительное письмо (в текстовом формате). Образец резюме хранится отдельно во внутреннем формате приложения CVA, на базе XML. CVAssistant пока отсутствует в пакетах дистрибутивов, но его легко скомпилировать; так, в Ubuntu установите зависимости с помощью `$ sudo apt-get install qt5-qmake libqt5-dev` и затем скомпилируйте его, скомандовав `$ qmake && make`. LXF

На диске

Дистрибутивы, приложения, игры, книги и всякое-разное...

Лучшее из Интернета, упакованное в 9 ГБ качественного DVD.



Дистрибутивы

В ранние годы LXF — еще тогда, когда диском был один CD, а дистрибутивы не занимали столько места — к LXF20 мы добавили опцию DVD, а владельцы старого «железа» были рады и двум CD. Установочный диск был обычно и программой установки, и репозиторием: при установке вы выбирали нужные пакеты из имеющихся. Всё изменилось после появления устанавливаемых live CD; на самом первом была PCLinuxOS. Теперь можно ознакомиться с дистрибутивом, не трогая свой жесткий диск, а при желании и установить его — и всё с одного и того же диска. Установочные диски снова уменьшились, и в Ubuntu долго придерживались той политики, что на их базовый дистрибутив должно хватать CD. Потом их диски выросли, но большинство установок остались в размере одного-двух CD. А значит, вы ежемесячно получаете три-четыре классных дистрибутива, и почти каждый можно тестировать с DVD и установить, только если он вам понравился.

Может, CD и ушли в прошлое, но читатели до сих пор просят о 32-битных дистрибутивах, и мы продолжим их включать, пока создатели дистрибутивов будут их выпускать, даже если число тех, кто их поставляет, снизится.

Neil

» Важно ВНИМАНИЕ!

Порченные диски

В маловероятном случае какого-то дефекта вашего LXF DVD обращайтесь, пожалуйста, по адресу disks@linuxformat.ru или телефону +7 (812) 309-0686.

Дистрибутив-впередсмотрящий

Fedora 22

Fedora — важный дистрибутив Linux, даже если вы его не используете. Red Hat является одним из основных разработчиков экосистемы Linux, вносящим свой вклад как в развитие ядра, так и в развитие пространства пользователя, а Fedora — испытательный полигон для новых идей.

Gnome 3 и Systemd сейчас используются весьма широко, и оба появились в предыдущих релизах Fedora. Wayland, графическая система всё откладываемого нового поколения, тоже вошла в предыдущий релиз Fedora. Так что на Fedora стоит взглянуть, если вас интересует будущее — и в некотором смысле она противоположна главному дистрибутиву прошлого месяца, Debian.

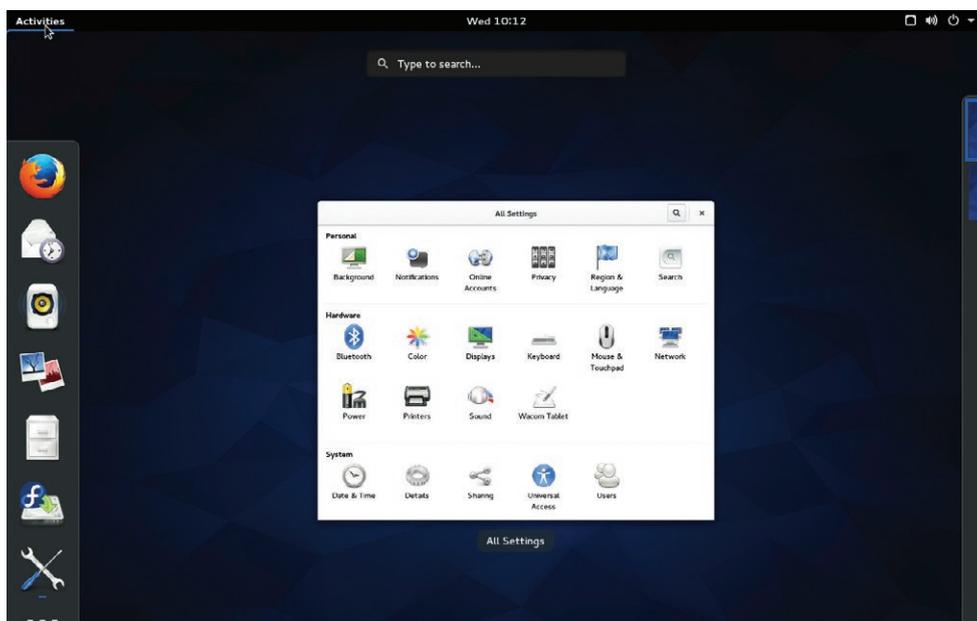
Fedora 22 продолжает тенденцию внедрения новых программ, и Wayland стал графической системой по умолчанию, загружаемую GDM (а если ваше оборудование не поддерживается Wayland, происходит откат к X). Это версия Gnome, рабочего стола Fedora по умолчанию, однако обновлены были все рабочие столы, и в них внесены все основные изменения: так, KDE обновился до Qt5, и на KDE можно перейти, установив Fedora на жесткий диск. Fedora 22 использует ядро Linux 4.0, и за сценой внесены некоторые улучшения пакетов ядра, процессов установки и обновления.

Итак, если вы хотите посмотреть, куда двинется Linux в ближайшем будущем, или, по крайней мере, увидеть некоторые возможности, ознакомьтесь с Fedora 22. Поскольку Red Hat царит в сфере серверов, где так важна виртуализация, Fedora имеет всё необходимое для применения в виртуальной среде, вроде Qemu или VirtualBox, хотя вам, вероятно, не удастся увидеть преимущества Wayland, если вы примените ее именно так.

Непослушные Nvidias

Live-системы Fedora и Sabayon могут столкнуться со сложностями при загрузке, если у вас более старая видеокарта Nvidia. Дело в том, что оба поставляются с новыми драйверами, которые больше не работают с рядом карт. Решением будет подождать, пока не прекратится процесс автоматического определения, что может занять несколько минут, или использовать опцию Безопасной загрузки. Опция Safe часто является оптимальным решением для Fedora, а вот загрузка Sabayon может потребовать терпения.

Это применимо только к некоторым картам и только к дискам live. Как только дистрибутив будет у вас на жестком диске, вы сможете установить нужные драйверы для своей карты. Здесь нет вины дистрибутивов или нашего DVD — причина в том, как Nvidia управляет своими проприетарными драйверами.



» Если вы планируете перепробовать множество виртуальных сред, Fedora 22 — прекрасный выбор.

FEDORA 22 (64-bit) **Рабочая станция**

Встроенная поддержка Docker и крутой рабочий стол GNOME **СЕРВИС РЕЛИЗ**

БЕСПЛАТНО №1 **Ретро!**
Этот классический выпуск ожидает вас больше для

UBUNTU 15.04
Веселая верветка
Теперь на SystemD

LINUX **ЛУЧШЕЕ ПО ДЛЯ LINUX**
ВСЕ, ЧТО ТРЕБУЕТСЯ ДЛЯ СТАРТА В LINUX

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.

32-битная редакция — как будто на дворе 2004-й

Ubuntu 15.04

За годы существования LXF DVD у двух дистрибутивов популярность среди читателей была неизменна. Первым был Mandrake, который превратился в Mandriva и затем, после нескольких приступов агонии, скончался (хотя некая часть его духа продолжает жить в Mageia). Другим был юный выскочка, который удивительно быстро обрел популярность, Ubuntu. Трудно поверить, что первый релиз был в 2004 году, версия 4.04. При двух релизах в год, 15.04 стал 23-м релизом Ubuntu.



Готовый к работе

Sabayon 15.06

Если вы хотите попробовать самое свежее ПО, полезно использовать дистрибутив с возобновляемым релизом; дистрибутивы с фиксированными версиями релизов обычно не обновляют пакетов между релизами, разве что для отладок. Arch и Gentoo популярны как раз из-за этого, но они требуют определенных усилий от пользователя. Sabayon — это дистрибутив, созданный поверх Gentoo, для чего, собственно говоря, Gentoo и предназначен,

поэтому вы получаете преимущества быстрого цикла возобновляемого релиза, не утруждаясь самостоятельной поддержкой основных частей системы. Кроме опций загрузки live и install, у вас есть возможность установить Sabayon в качестве медиа-центра Kodi или машины Steam. Если вы выберете второе, при первой загрузке он загрузит 200+ МБ обновлений Steam; это стоит учесть, если у вас тарифицированное соединение с Интернетом. [LXF](#)



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.

Mement86+ Проверьте ОЗУ на предмет сбоев.

Plop Простой менеджер загрузки для запуска ОС с CD, DVD и USB.

RaWrite Создавайте загрузочные диски в MS-DOS в Windows.

Smart Boot Manager Не зависящий 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 Грегом Кроа-Хартманом [Greg Kroah-Hartman].

Руководство системного администратора Linux Контролируйте свою систему

Обзор инструментов Полный обзор инструментов GNU.



Раритет!
LXF №1
в формате PDF

Пропустили номер?



Закажите его через сайт www.linuxformat.ru в «ГНУ/Линуксцентре»! Журналы доставляются и в печатной, и в электронной форме, так что с момента открытия браузера до получения нужного вам выпуска LXF может пройти всего пара минут!

Прямо сейчас для заказа доступны следующие номера:



LXF197
Июнь 2015

250 руб.

- » Жизнь с Малиной Лучшие проекты для Raspberry Pi
- » Выбор виртуальной реальности Компьютер в компьютере
- » Lego для инженеров Моделирование систем
- » Пламенные звуки Аудио в Linux

LXFDVD: Bodhi 3.0, SteamOS Beta, Zentyal, Antergos, Fedora, Robolinux, руководство по хромбукам, 10 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_197/
PDF-версия: shop.linuxformat.ru/elxf_197/



LXF198
Июль 2015

250 руб.

- » Ubuntu Для кого-то — синоним Linux
- » Из прекрасного далека Удаленная работа
- » Защищайтесь, сударь! Безопасность Linux
- » Хакеры и роутеры Нестандартные прошивки

LXFDVD: Ubuntu 15.04, Android x86, Grml, 4MLinux, Antergos, Chromium, Mint, 10 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_198/
PDF-версия: shop.linuxformat.ru/elxf_198/



LXF199
Август 2015

250 руб.

- » Сделай сам Да, вам по силам собрать ПК!
- » Лети с приветом Быстрый обмен сообщениями
- » Загадки протоколов Мы их разгадаем
- » Система Дух времени

LXFDVD: Debian 8, LXLE 14.04, Elementary OS, Point Linux, ROSA, Zorin OS, 9 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_199/
PDF-версия: shop.linuxformat.ru/elxf_199/

А чтобы не упустить ничего, оформите подписку! Все, кто подписался на печатную версию журнала через www.linuxformat.ru/subscribe или www.linuxcenter.ru, получают электронную версию в подарок! На сайте shop.linuxformat.ru вы также сможете приобрести предыдущие выпуски LXF.

Подписывайтесь на сайте www.linuxformat.ru/subscribe

Телефоны отдела подписки:

- » Санкт-Петербург (812) 309-0686
- » Москва (499) 271-4954



Linux Format VKontakte

Вступайте в нашу
группу vk.com/linuxform

На странице LXF ВКонтакте вы найдете:

- » Новости о Linux
- » Статьи из архива LXF
- » Анонс свежего выпуска LXF и часть статей из него
- » Живое общение и онлайн-консультацию по подписке на наше издание

Страница 1

» Содержание

LINUX
ФОРМАТ

ДИСТРИБУТИВЫ

Fedora 22 (64-битный)
 Redhat 6 20150904 (64-битный)
 OAS 1.4.2 (64-битный)
 Ubuntu 15.04 (32-битный)

СТАВШЕЕ КЛАССИКОЙ

Linux Format 01
 Ответы по Linux

ПОМОЩЬ

Руководство новичка
 Руководства
 Ответы

НОТРІСКС

Classiast! Приложение по работе
 Dust! Naching 2D Имитатор гонок

ГЛАВНОЕ

Клз Аудиоинструмент
 Книга Программа для рисования
 Lolipop Музыкальный плеер
 dritpoint Клиент Vbitpoint
 DiscT Инструмент настройки
 Sefixus Настольная издательская система
 Tanks of Freedom Подлая игра стратегия
 Tmxk Мультиплеер терминала
 Udel Менеджер закладок

Обсуждения
 HandIno
 Kemei
 Mestest66+
 Plo
 SPM
 W/Dial

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

Внимательно прочтите это перед тем, как использовать LXF DVD!

FEDORA 22

64-битный

Рабочая станция

Встроенная поддержка Docker и крутой рабочий стол Гnome

СВЕЖИЙ РЕЛИЗ

БЕСПЛАТНО №1

Ретро!

Этот классический выпуск оживит вам былые дни



UBUNTU 15.04

Веселая верветка

Теперь на SystemD

32-битный

С ПОСЛЕДНИМИ ИСПРАВЛЕНИЯМИ



LINUX DVD ЛУЧШЕЕ ПО ДЛЯ LINUX
 ВСЕ, ЧТО ТРЕБУЕТСЯ ДЛЯ СТАРТА В LINUX

Содержание

LINUX FORMAT

Страна 2

ДОКУМЕНТАЦИЯ: 10 КНИГ О LINUX (НА АНГЛИЙСКОМ ЯЗЫКЕ)

Advanced Bash Scripting Guide Подробное руководство по программированию на Bash для начинающих

Bash Guide for Beginners Руководство по Bash для начинающих

Volume Shell Scripting Начальное руководство по программированию на Bash

The Cathedral and the Bazaar Классический текст Эрика Раймонда [Eric S Raymond] «Собор и базар» Linux Kernel in a Nutshell Описание ядра Linux, созданное одним из его выдающихся разработчиков — Греггом Кроа-Хартманом [Greg Kroah-Hartman]

System Administrators Guide Руководство по базовому администрированию Linux

GNU Tools Summary Руководство по работе в командной строке и обзор основных утилит GNU

The Debian Administrator's Handbook Руководство администратора, написанное разработчиками Debian

Dive Into Python Учебник по программированию на Python

Intro to Linux Начальное руководство по Linux

Linux Dictionary Словарь Linux, объясняющий специфическую терминологию

Tools Summary Обзор утилит

ДИСТРИБУТИВЫ

Linux Lite 2.6 (64-битный)

Sabayon 15.06 (64-битный)

Tails 1.6 (32- и 64-битный)

IPFire 2.17 (32-битный)

Sparky Linux 4.1 (32-битный)

Пожадуйста, перестаньте использовать именем Адного Диска ознакомьтесь с инструкцией, опубликованной в журнале на стр. 109!

КОММЕНТАРИЙ Присылайте ваши пожелания и предложения по электронной почте: info@linuxformat.ru

ДВОЕКЛЕТНЫЕ ДИСКИ В маловероятном случае обнаружения дефектов на данной диске, обращайтесь, пожалуйста, по адресу disk@linuxformat.ru

Настоящий диск тщательно тестировался и проверялся на всех старых платформах, однако, как и в случае с любым новым ПО, мы рекомендуем вам использовать аппаратный сканер. Мы также рекомендуем всегда иметь под рукой актуальную резервную копию данных вашего жесткого диска. К сожалению, редакция Linux Format не в состоянии принимать на себя ответственность за любые повреждения, разрушения или иные убытки, которые могут повлечь за собой использование этого DVD, представленных на нем программ или данных. Прежде чем устанавливать какое-либо ПО на компьютер, пожалуйста, свяжитесь с нами по телефону или по электронной почте.

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 cdfrecord

Самый быстрый способ записать ISO-образ на чистую матрицу — это обратиться к программе *cdfrecord*. Для всех перечисленных ниже действий потребуются права суперпользователя-*root*. Сначала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdfrecord -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:

```
cdfrecord dev=0,3,0 -v /путь к образу/image.iso
```

Чтобы упростить дальнейшее использование *cdfrecord*, сохраните некоторые настройки в файле `/etc/default/cdfrecord`. 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

```
cdfrecord -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 *cdfrecord* 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.



UNIXFORUM.org



Есть вопросы?
Задавайте!

<http://unixforum.org/>

Отдел дистрибьюции
ГНУ/Линуксцентра
приглашает дилеров
и дистрибьюторов
к сотрудничеству!

Широкая сеть представительств
в разных городах России
позволит вам оптимизировать
процессы логистики и доставки товара

ПОДРОБНЕЕ О ПАРТНЕРСКОЙ ПРОГРАММЕ:
WWW.LINUXCENTER.RU/PARTNER/



ДИСТРИБУТИВЫ GNU/LINUX
НА USB FLASH



Linux Mint 16



Ubuntu 14.04

А ТАКЖЕ
версии для юриди-
ческих лиц —
с лицензионным
договором
присоединения

USB Flash 8 Gb

495 рублей

www.linuxcenter.ru/shop/distros/usb/

LINUX
FORMAT

Главное в мире Linux

Реклама в журнале
Linux Format

3-я обложка	200 000
2-я обложка	230 000
4-я обложка	275 000
Разворот	230 000
1/1 полоса (210×297 мм)	140 000
1/2 полосы (197×141, 92×280 мм)	75 000
1/4 полосы (92×113 мм)	45 000

Тел.: (812) 309 06 86

Цены указаны в рублях и не включают НДС (18%)

ГНУ/Линуксцентр
приглашает на работу!



ВАКАНСИЯ: Разработчик систем электронного документооборота

ТРЕБОВАНИЯ:

- » Опыт внедрения и сопровождения систем электронного документооборота на базе Alfresco
- » Умение описать бизнес-процесс
- » Знание SQL, понимание принципов построения и функционирования баз данных
- » Знание СЭД, отличных от Alfresco, приветствуется

ОБЯЗАННОСТИ:

- » Разработка систем с нуля, сопровождение
- » Настройка системы в соответствии с требованиями бизнеса
- » Написание инструкций для конечных пользователей

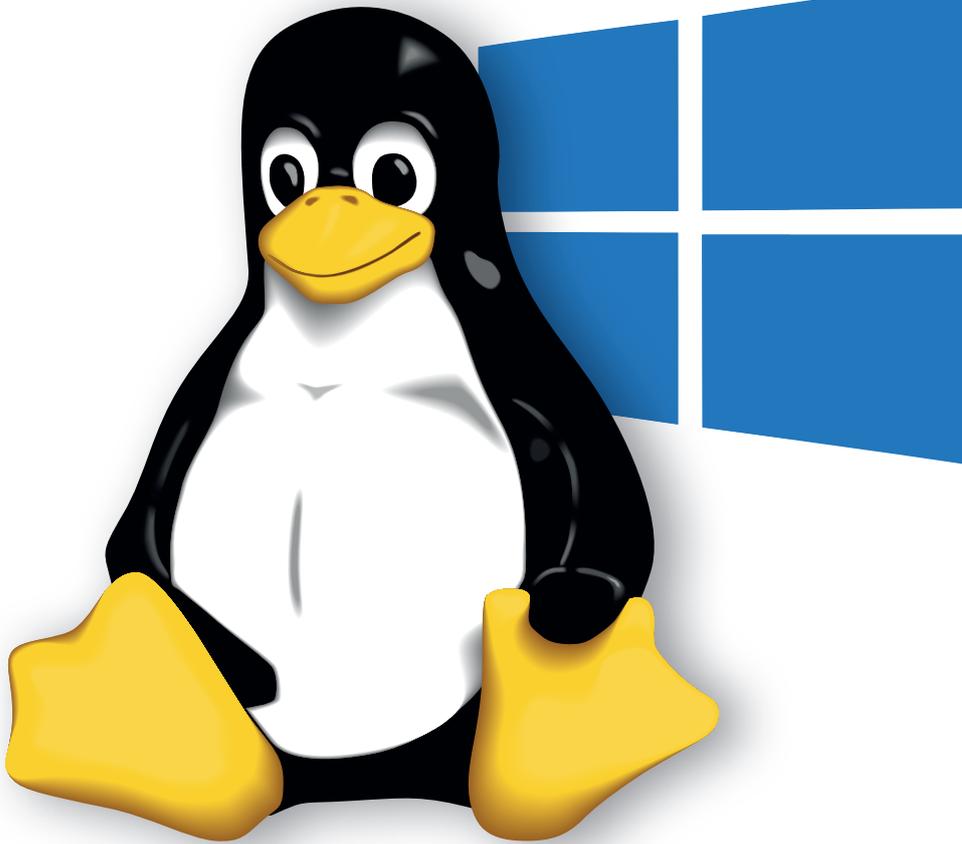
ПОДРОБНЕЕ: www.linuxcenter.ru/vacancy/

Футболки GNU/Linux

БОЛЕЕ 100 ВИДОВ. ВСЕ РАЗМЕРЫ И ЦВЕТА



WWW.LINUXCENTER.RU/SHOP/GIFTS/



В октябрьском номере

Windows 10 vs Linux 4.1

Мы-то знаем, кто победит, но насколько серьезный соперник — новинка от Microsoft для широкого диапазона устройств?

ОС для серверов

Вам нужен пуленепробиваемый, простой в установке и настройке сервер? Тестируем лучшие предложения от Linux.

Вопрос веры

Кому можно доверить ваши данные онлайн?
Опубликованный отчет EFF способен вас напугать...

Изучим LaTeX

Почему с этой системой так носятся в академических кругах?
Освоим лучший инструмент разметки!

Содержание будущих выпусков может меняться — у нас же праздник, вы чего?

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

Литературный и выпускающий редактор

Елена Толстякова

Переводчики

Елена Ессяк, Даниил Кривошеин, Светлана Кривошеина, Валентин Развозжаев, Елена Толстякова

Редактор диска

Александр Баракин

Верстка, допечатная подготовка

Сергей Рогожников

Технический директор

Андрей Смирнов

Директор по рекламе

Владимир Савельев advert@linuxformat.ru

Генеральный директор

Павел Фролов

Учредители

Частные лица

Издатель

ООО «Линукс Формат»

Отпечатано в типографии ООО «ЛД-ПРИНТ»

196644, Санкт-Петербург, Колпинский р-н, пос. Саперный, территория предприятия «Балтика», д. 6/н, лит. Ф
Тел. (812) 462-8383, e-mail: office@ldprint.ru
Заказ 13096

РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ

Редактор Нейл Мор [Neil Mohr] neil.mohr@futurenet.com

Научный редактор Джонни Бидвелл [Jonni Bidwell] jonni.bidwell@futurenet.com

Выпускающий редактор Крис Торнетт [Chris Thornett] chris.thornett@futurenet.com

Художественный редактор Эфраин Эрнандес-Мендоза [Efraim Hernandez-Mendoza] efrain.hernandez-mendoza@futurenet.com

ПОДГОТОВКА МАТЕРИАЛОВ

Джонни Бидвелл [Jonni Bidwell], Нейл Ботвик [Neil Bothwick], Джолон Браун [Jolyn Brown], Крис Браун [Chris Brown], Пол Хадсон [Paul Hudson], Аластер Дженнингс [Alastair Jennings], Хуан Мартинес [Juan Martinez], Бен Наттолл [Ben Nuttall], Лес Паундер [Les Pounder], Фил Сэвидж [Phil Savage], Маянк Шарма [Mayank Sharma], Шашанк Шарма [Shashank Sharma], Александр Толстой [Alexander Tolstoy], Евгений Балдин, Андрей Гондаренков, Денис Силаков, Валентин Синицын, Алексей Федорчук

Иллюстрации Шейн Коллиндж [Shane Collinge], iStockPhoto

КОНТАКТНАЯ ИНФОРМАЦИЯ

UK: Linux Format, 30 Monmouth Street, Bath BA1 2BW
Тел. +44 01225 442244, email: linuxformat@futurenet.com

РОССИЯ: Санкт-Петербург, пр. Медиков, 5, корп. 7
Тел. +7 (812) 309-0686, e-mail: info@linuxformat.ru

По вопросам сотрудничества, партнерства, оптовых закупок:
partner@linuxcenter.ru

Авторские права: статьи, переведенные из английского издания Linux Format, являются собственностью или лицензированы Future Publishing Ltd (Future plc group company). Все права зарегистрированы. Никакая часть данного журнала не может быть повторно опубликована без письменного разрешения издателя.

Все письма, независимо от способа отправки, считаются предназначенными для публикации, если иное не указано явно. Редакция оставляет за собой право корректировать присланные письма и другие материалы. Редакция Linux Format получает эксклюзивное 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 вам материалов, и мы постараемся исправить эту ошибку. Редакция Linux Format не несет ответственности за опечатки.

Ответственность за содержание статьи несет ее автор. Мнение авторов может не совпадать с мнением редакции.

Все присланные материалы могут быть помещены на диски — CD или DVD, поставляемые вместе с журналом, если не было оговорено иное.

Ограничение ответственности: используйте все советы на свой страх и риск. Ни при каких условиях редакция Linux Format не несет ответственность за повреждения или ущерб, нанесенные вашему компьютеру и периферии вследствие использования тех или иных советов.

Linux — зарегистрированный товарный знак Линуса Торвальдса [Linus Torvalds].
"GNU/Linux" заменяется на "Linux" в целях сокращения. Все остальные товарные знаки являются собственностью их законных владельцев. Весь код, опубликованный в журнале, лицензирован на условиях GPL v3. См. www.gnu.org/copyleft/gpl.html

За информацией о журналах, издаваемых Future plc group company, обращайтесь на сайт www.futureplc.com.



© Linux Format 2005

© Future Publishing Ltd 2005

BATH • LONDON • MILAN • NEW YORK • PARIS • SAN DIEGO • SAN FRANCISCO

16+

Добро пожаловать в робототехнику!

ScratchDuino

Электронный комплекс на основе свободного аппаратного обеспечения для школ и вузов

ScratchDuino.Лаборатория

Плата расширения с датчиками ввода-вывода — как установленными на плате, так и подключаемыми. Предназначена для программирования взаимодействия компьютера с внешними устройствами.

Полная интеграция («из коробки») со средой программирования Scratch, предназначенной для детей.

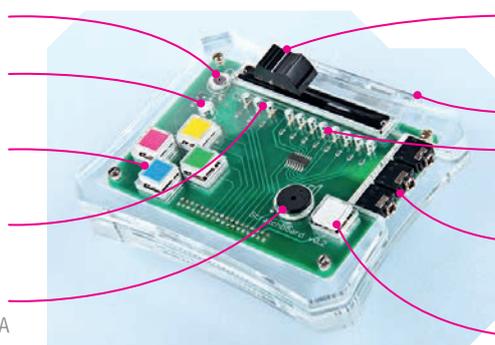
ДАТЧИК ЗВУКА*

ДАТЧИК ОСВЕЩЕННОСТИ*

КРЕСТОВИНА КНОПОК

ЦВЕТНЫЕ СВЕТОДИОДЫ

ДИНАМИК ДЛЯ ВЫВОДА ЗВУКА



ПЕРЕМЕННЫЙ РЕЗИСТОР (ПОЛЗУНОК)*

КАРТРИДЖ ARDUINO

СВЕТОДИОДЫ, УПРАВЛЯЕМЫЕ ЧЕРЕЗ РАЗРЯДНУЮ МАТРИЦУ

РАЗЪЕМЫ ДЛЯ ПОДКЛЮЧЕНИЯ ВНЕШНИХ ДАТЧИКОВ

КНОПКА*

* РАБОТАЮТ В РЕЖИМЕ ЭМУЛЯЦИИ RISCBOARD, НЕ ТРЕБУЯ ДОПОЛНИТЕЛЬНОГО ПРОГРАММИРОВАНИЯ.

ЦЕНА **9500 руб.**

ГАРАНТИЯ 3 ГОДА, ВКЛЮЧАЯ ТЕХНИЧЕСКУЮ И МЕТОДИЧЕСКУЮ ПОДДЕРЖКУ

ScratchDuino.Робоплатформа

Внешний робот-исполнитель, управляемый из среды программирования Scratch. Не требует навыков программирования на языках высокого уровня и может применяться для обучения, начиная с младших классов.

ScratchDuino.Робоплатформа

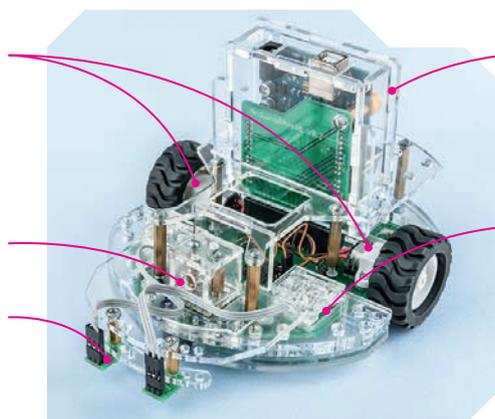
обеспечивает возможность:

- подключения внешних датчиков без механических соединений и пайки;
- расширения через последовательную шину RoboBus;
- установки деталей Lego Technics.

МОТОР-РЕДУКТОРЫ

ФОТОДАТЧИК

ДАТЧИК ЛИНИИ



КАРТРИДЖ ARDUINO

МАГНИТНЫЙ ЦИФРО-АНАЛОГОВЫЙ РАЗЪЕМ ДЛЯ ПОДКЛЮЧЕНИЯ ВНЕШНИХ ДАТЧИКОВ БЕЗ МЕХАНИЧЕСКИХ СОЕДИНЕНИЙ И ПАЙКИ*

* ВСЕГО НА РОБОПЛАТФОРМЕ 5 РАЗЪЕМОВ. ВОЗМОЖНО ПОДКЛЮЧЕНИЕ ДАТЧИКОВ КАСАНИЙ, ИК-ДАТЧИКОВ РАССТОЯНИЙ/ПРЕПЯТСТВИЙ И ДР.

ЦЕНА **19 500 руб.**

ГАРАНТИЯ 3 ГОДА, ВКЛЮЧАЯ ТЕХНИЧЕСКУЮ И МЕТОДИЧЕСКУЮ ПОДДЕРЖКУ

Назначение ScratchDuino

- Изучение взаимодействия компьютера с внешней средой
- Изучение изменения параметров внешней среды
- Изучение процессов передачи информации и принципов ее построения
- Изучение внешних устройств управления
- Моделирование устройств

Среда разработки

Lazarus (язык Pascal) ■ Scratch ■ Arduino IDE

ScratchDuino адаптирован для учебных заведений. Поставляется с комплектами учебно-методических материалов.

Продукт разработан при финансовой поддержке Фонда содействия развитию малых форм предприятий в научно-технической сфере

WWW.SCRATCHDUINO.RU

(812) 309-0686 (Санкт-Петербург) ■ (499) 271-4954 (Москва)

HETZNER DEDICATED SERVERS

HETZNER
— ONLINE —

В ЦЕЛОСТИ И СОХРАННОСТИ!

ХРАНЕНИЕ ДАННЫХ В ГЕРМАНИИ



ВЫДЕЛЕННЫЙ СЕРВЕР PX60

- Intel®Xeon® E3-1270 v3 Quadcore Haswell
- 32 ГБ ECC RAM
- 2 x 2 ТБ 6 Гбит/с SATA 7200 об/м Enterprise класс Software-RAID 1
- 30 ТБ Трафик*
- Операционная система Linux
- Подключение 1 Гбит порт
- Гарантировано 200 Мбит/с
- IPv6 подсеть (/64)
- Без минимального контракта

2600

рублей в месяц
+ Установка 3700 рублей



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.