DVD ДАРОМ! С DEBIAN 8 JESSIE

LINUX FORMAT

Главное в мире Linux

Август 2015 № 8 (199)

PulseAudio
Вещаем потоком

» Управляем звуками
» Заведем звуковой сервер
» Настроим удаленный клиент

+ Поболтаем через лучшие клиенты обмена сообщениями

СТРОИМ LINUX-ПК

Создадим 64-битную систему на Ubuntu

» Настольный ПК » Домашний сервер » Медиа-центр » Игры

ЛЕГКО! Пошаговое руководство

ЕСТЬ ЭЛЕКТРОННАЯ ВЕРСИЯ ДЛЯ ВАШЕГО ПЛАНШЕТА!

Сканируем весь мир

«Мы переписали ваш код Linux, не могли бы вы дать нам доступ на GitHub?»

Том Смит — про сохранение исторических документов с. 42

Освоим Systemd

» Разберитесь в новой системе инициализации — под нашим руководством

Chromium OS
Сами сделаем хромбук
» Поставим ОС от Google на любой ПК

Игрострой
Unity 5 на Linux
» Создадим собственную трехмерную чудо-игру

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

Linux center
www.linuxcenter.ru

iTeleRadio

ИНТЕРАКТИВНОЕ **РАДИО&ТЕЛЕВИДЕНИЕ**

ОГРН 1097847090021

LINUX FORMAT
Главное в мире Linux

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

В этом месяце мы спросили наших экспертов вот о чем. Мы собираем новые Linux-ПК — что бы вы встроили или хотели бы видеть встроенным в ядро или ваш любимый дистрибутив?

Джонни Бидвелл
Меня вдруг стукнуло, что существует масса ПО, дающего нам чуть менее чем почти супер-мощь: социальные медиа почти читают мысли людей, *Git* — почти машина времени (парадоксальные слияния тремя способами), а *Tor* — почти шапка-невидимка. При такой супер-мощи я бы ответил и получше!

Нейл Ботвик
Мне бы подошла система подсказок, управляемая голосом, реагирующая не только на содержание сказанного, но и на громкость и тон. А в идеале — распознавала бы ругань и тогда повышала приоритет проблемы. Реакция на угрозу или реальное применение насилия к ПК тоже приветствовалась бы.

Лес Паундер
Каждый раз, когда я слышу, что Linux можно поставить хоть на дохлого барсука (http://bit.ly/deadbadger), я зову [*Ред.*: — На помощь?] армию кибер-барсуков, пользуясь подспорьем Raspberry Pi. Ну да, это не священное воинство и не мертвые звери, но уж Linux на них точно работает.

Шон Конвэй
Похудение за пребывание в зоне Linux. Сжигание жиров не просто если я сдерживаю и перенаправляю мои эмоции, а когда эмоции положительны, энергичны и сфокусированы на полезной деятельности и изучении Linux, заодно с попиванием пивка. [*Ред.*: — Ответы на открытке по обычному адресу.]

Александр Толстой
Хорошо бы разработчики Linux получили функцию чата для своих комментариев, встроенную в исходники ядра, чтобы не выплескиваться на публичных форумах и в списках рассылки. Можно было бы рисовать отличные графики подсчета ненормативных выражений, с гигантскими пиками.

Валентин Синицын
Во времена MS-DOS у меня в **autoexec.bat** стояла программа, которая печатала на экран какое-нибудь умное изречение, каждый раз новое. В эпоху Интернета можно развить идею — например, научить ее брать умные мысли из Твиттера... Хотя тогда она, наверное, будет в основном молчать.

Эпоха «великого самосбора»

» Тема этого номера не нова, но актуальна — выбор оптимальной конфигурации «железа» для решения тех или иных задач. С начала 1990-х, когда «стандартная архитектура» стала стандартом и в нашей стране, об этом написаны сотни, если не тысячи журнальных статей. Публикации в Сети сосчитать попросту невозможно.

Если вспомнить историю ИТ, обнаружится парадоксальный факт: кроме IBM PC, ни один аппаратный стандарт не позволяет обычному пользователю самостоятельно выбрать оптимальный (по его мнению и в рамках бюджета) набор совместимых компонентов и с помощью единственной отвертки собрать из них вполне работоспособный компьютер.

«До», в эпоху ZX Spectrum и Радио-86РК, требовались паяльник и хорошее знание схемотехники. «После», в эпоху смартфонов, планшетов и «Интернета вещей», системы окончательно стали on-chip и позволяют лишь подключить внешние устройства по стандартным интерфейсам. «Свобода оптимизации» упразднена. Альтернативные прошивки — на свой страх и риск потери гарантии.

И вообще, зачем она вам, эта свобода? Вы же не собираете «под себя» утюг или телевизор. Не устраивает эта модель — осенью выйдет следующая. По той же цене.

Судя по всему, эпоха «великого самосбора» близится к завершению. А жаль...

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

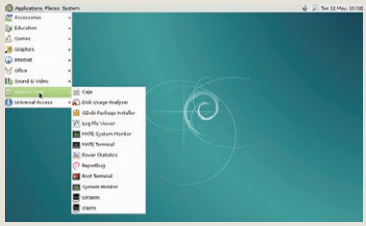
LINUX FORMAT

Содержание

«Если вы это построите, он придет». Поле чудес

Обзоры

Debian 8 “Jessie” 14
Очередной релиз Debian прибыл! Крупнейшее обновление за 22 года, с *Systemd* и прочим.



› Debian 8 во всей красе — с новой системой инициализации, уместной в XXI веке.

Visual Studio Code 16
Адский огонь замерз: продукт от Редмонд — и в *Linux Format*! Нас оправдывает то, что он бесплатен и сделан для Linux.

Облачное хранилище 1&1 17
Как работается свежему облачному сервису от 1&1 на Linux и SSD? Да преотлично.

D-Link и DWA-182 18
Предлагаем выход на случай, если ваш домашний роутер конфликтует с микроволновкой.

Dell Chromebook 11 20
Рынок образования атакован: вот, возможно, идеальный хромбук для школы — прочный, умелый и богатый функциями.

› Новый 2015 Dell Chomebook 11 — крепкий орешек, да и умен к тому же!

Fritz!WLAN Repeater 1750E 21
Хотите расширить беспроводную сеть? Fritz! думает, что у них есть идеальный ответ: работает на 802.11ac, то есть со свистом.

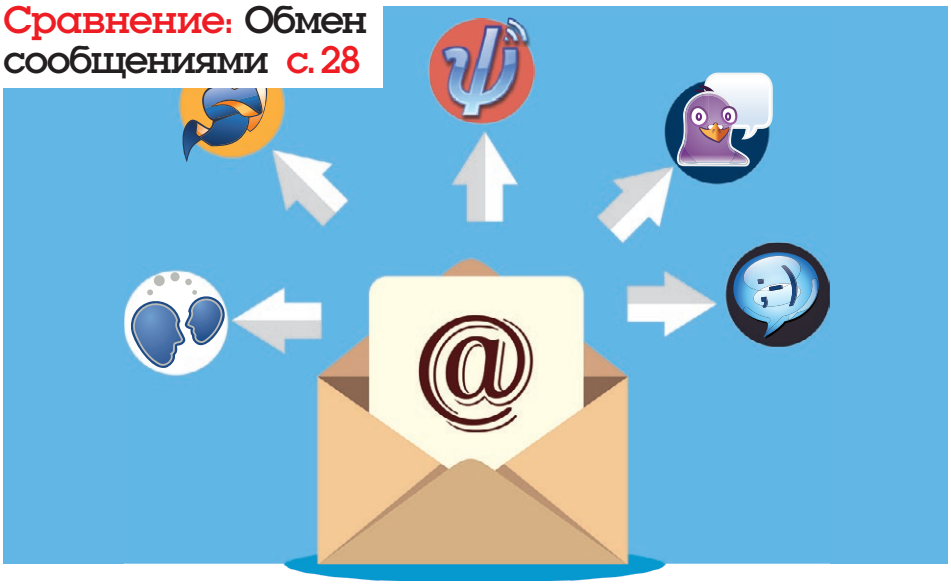
BioShock Infinite 21
Очередная AAA-игра пришла в Linux, причем работает! Там, правда, полно фанатиков, но вам услужливо дадут немаленькие пушки.

Pillars Of Eternity 22
Действительно новая ролевая игра — и с отблесками золотого прошлого.

СТРОИМ ПК!

Контролируйте не только свое ПО, но и свой ПК — сделав собственную супер-систему с. 34

Сравнение: Обмен сообщениями с. 28

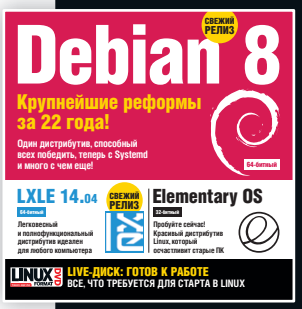


Люди говорят

“Это путь, который может привести вас туда, куда вы и не мечтали.”

Том Смит — о том, как роботы и СПО спасают историю с. 42

На вашем бесплатном DVD



Debian 8 Jessie 64-битный, LXLE 14.04 64-битный, Elementary OS 32-битный

» Каждый месяц — только лучшие дистрибутивы

ПЛЮС: Hotpicks, Сравнение и более того! с. 98

Побалуйте себя и любимых подпиской на LXF!

Доступно в AppStore!

www.linuxformat.ru/subscribe

Пользователям Raspberry Pi

Новости Pi 80
Фонд Pi целится на сотни тысяч педагогов с Picademy. Плюс, Pi надевает скафандр, а модель B+ падает в цене.

Обзор: Принтер Pipsta 81
Микропринтер для Raspberry Pi — Лес Паундер разберется, на что он способен.

Бросаем кубик 82
Объединим GPIO, немного кода и светодиодов ради создания игральных костей.

Строим метеостанцию 84
Объединим температурный датчик Pi и ПО графопостроения ради слежения за погодой.

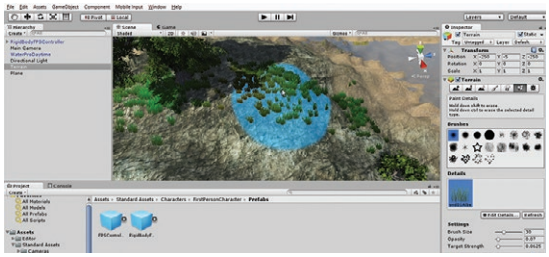
Ищите в номере

Открываем Systemd 54
Спорная система инициализации *Systemd* охватила все основные дистрибутивы, так что пора вам принять ее и научиться любить *Journald*.

› *Systemd* — 'd' значит «доминирует».

Академия кодинга

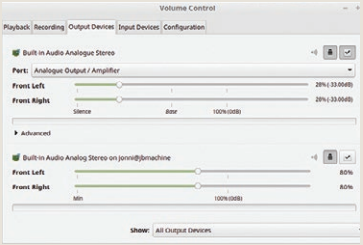
Система: Функции времени 76
Раскрываем тайны времяисчисления и магический смысл 2038 года для компьютеров на UNIX, а также показываем, что конец света — дело поправимое.



Учебники

Chrome OS
Сделаем хромбук 64
Заполучим ОС от Google на своем ноутбуке, для удовольствия и науки! (Рифма получилась случайно.)

PulseAudio
Вещание аудио 66
Как заставить вашу какофо... э-э, сладкозвучную мелодию передаваться через удаленный аудиопоток.



› Сделайте потоковое вещание истинным наслаждением.

Perf_events
Анализ системных событий .. 68
Диагностируем системные события стандартной системой мониторинга Linux, чтобы улучшить производительность.

Tcpdump
Нырнем в TCP/IP 72
Объясняем, как применить *Tcpdump* для погружения в ваши пакеты. И наконец-то разберемся в интернет-протоколе досконально.

Постоянные рубрики

Новости 6
Суперкомпьютеры снижают темп, Blackberry меняет ОС, *Firefox* пересматривается, импорт удаленки замещается, дроны разлетались, КНР усиливает контроль, Fedora обзавелась MIPS, а *VirtualBox* получает drag-n-drop.

Новости Android 26
Android атакуют алчные вирусы, Google платит за дефекты, а обновления надо устанавливать срочно, не то падете жертвой атак!

Сравнение 28
Довольно болтать — клиенты обмена мгновенными сообщениями вас живо дисциплинируют. *Empathy*, *Jitsi*, *Kopete*, *Pidgin*, *Psi*.

Интервью LXF 42, 48
Робот **Тома Смита** умеет бережно обращаться с драгоценными историческими документами, а **Александр Горный** и **Ян Кисель** вовсю развивают проекты в Mail.Ru Group.

Рубрика сисадмина 58
Услышав новости от Microsoft, м-р **Джолион Браун** призадумался, но выманить его из бункера не удалось: он вернулся к чудесам CoreOS.

Ответы 88
ВАШИ ПРОБЛЕМЫ РЕШЕНЫ!
Нейл Ботвик — про резервное копирование при установке Mint, создание множества каталогов, установку *Dracut*, перенаправление e-mail, спецсимволы, первый опыт в Linux.

HotPicks 92
Отведайте горяченького! Лучшие в мире новинки свободного ПО: *DNSCrypt*, *GNU Hurd*, *GNU Nano*, *Guake*, *MegaGlest*, *Minilens*, *Plank*, *QmapShack*, *Tor Browser*, *Vera*, *ZFSonLinux*.

Диск Linux Format 98
Содержимое двустороннего DVD этого месяца.

Пропустили номер? ... 100
Сильно расстраиваться не стоит: в хорошие руки мы его с удовольствием отдадим.

Через месяц 105
У нас выйдет журнал за номером 200 и нам стукнет 10 лет! Предадимся воспоминаниям.

В ЭТОМ НОМЕРЕ: Рейтинги » Смена ОС » Ревизия Firefox » Наше решение » Дроны набирают силу » КНР в глухой обороне » Linux и MIPS » Потяни-и-брось

РАССТАНОВКА СИЛ

Top-500: Темп упал

Но Linux остается ведущей ОС на суперкомпьютерах.

» Рубрику готовил АНДРЕЙ ГОНДАРЕНКОВ

Опубликованный в июне 2015 г. 45-й рейтинг суперкомпьютеров Top-500 неожиданностей не содержит: GNU/Linux, уже традиционно, является преобладающей ОС на самых быстрых компьютерах мира.

В целом, общая производительность всех 500 систем выросла до 363 петафлоп/с (квадриллионов вычислений в секунду). В ноябре 2014 г. было 309 петафлоп/с, а год назад — «всего лишь» 274 петафлоп/с. Производительность измерялась тестом Linpack.

Ситуация на первый взгляд впечатляющая, однако все не столь замечательно, как это может казаться. Такое увеличение производительности на самом деле говорит о заметном снижении темпов роста. И государства в целом, и отдельные компании уже не столь охотно инвестируют в самые большие суперкомпьютеры. Например, единственный новый участник в первой десятке нынешнего Top-500 — занимающая 7-ю позицию Shaheen II, система Cray XC40 из Научно-технологического университета имени короля Абдаллы (KAUST) в Саудовской Аравии (5,536 петафлоп/с). А самый быстрый ныне суперкомпьютер мира, Tianhe-2 из Национального университета оборонных технологий КНР, на базе Intel, остался при тех же 33,86 петафлоп/с, с какими впервые попал в Top-500 еще в 2013 г. На втором месте (17,59 петафлоп/с) — Titan, система Cray XK7 из Национальной лаборатории Оук-Ридж при Департаменте энергетики США. В десятку самых быстрых мира также вошли IBM BlueGene/Q Sequoia (17,1 петафлоп/с), Fujitsu K (10,51 петафлоп/с) и IBM BlueGene/Q Mira (8,58 петафлоп/с).

США остается лидером по количеству компьютерных суперсистем: сейчас их 233 (ранее было 231). КНР теряет позиции: 61 система в прошлом рейтинге и только 37 в нынешнем. За полгода возросло количество европейских систем: 141 вместо 130, а также систем из Японии: 39 против 32. Но влияние китайских компаний в отрасли растет благодаря Lenovo, у которой три системы в Top-500 под своим брендом и еще 20 совместно с IBM.

HP играет ведущую роль в общем количестве систем — 178 (35,6 %), далее следует IBM — 111 систем (22,2 %).

ИМПОРТОЗАМЕЩЕНИЕ

Российская удаленка

Разработана инфраструктура виртуальных рабочих столов.

Российский производитель серверного оборудования «Рикор.ИТ» и ООО «Альт Линукс» расширяют сотрудничество в сфере разработки импортонезависимой высокотехнологичной инфраструктуры виртуальных рабочих столов. Представленный программно-аппаратный комплекс на базе линейки серверов Rikor EcoServer позволит почти в 2 раза сократить расходы компаний на развертывание и управление удаленными рабочими местами (УРМ) по сравнению с использованием серверов x86. Новое комплексное решение для виртуальных РМ на разработанных и произведенных в России серверах — достойная альтернатива VDI. Главное преимущество отечественной платформы — экономия ИТ-бюджета компании за счет низкой стоимости владения, простоты администрирования, быстрого развертывания РМ и надежной системы антивирусной защиты. Решение полностью отвечает повышенным требованиям к информационной безопасности, предъявляемым к продукции в рамках закупок для государственных структур. Протестирована совместимость ПО ALT Linux Desktop и серверов Rikor EcoServer R314. Испытания включали серии тестов общей работоспособности программных продуктов, оценку быстродействия и надежности. Платформа содержит модули системы управления и балансировки нагрузки, коммутационное оборудование, сервис хранения данных и удаленных рабочих столов.

EcoServer — новый класс серверов на архитектуре ARM, обслуживающих облачные и почтовые сервисы, параллельные вычисления, электронные документооборот и коммерцию, распределенные БД, организацию виртуальных РМ. EcoServer сократит расходы на энергопотребление в 10 раз и повысит эффективность использования площадей ЦОД в 9 раз.

Источники: www.top500.org; www.altlinux.ru

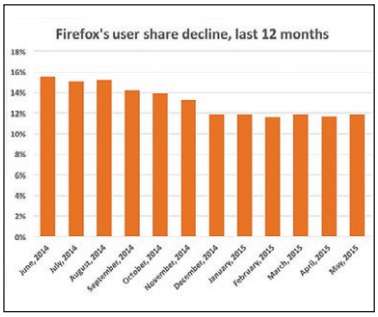
БРАУЗЕР

Отделим мух от котлет

Mozilla пересматривает подход к многозадачности в *Firefox*.

Mozilla активизировала работу по отделению рендеринга страниц в *Firefox* от контента, что улучшает безопасность и стабильность браузера: Крис Петерсон [Chris Peterson], технический руководитель программы по разработке ПО с открытым исходным кодом, объявил о расширении команды, работающей над этим проектом. Технология Mozilla по внедрению многозадачности [multiple processes] в *Firefox* получила название Electrolysis (или кратко "e10s").

Electrolysis, который уходит корнями в 2009 г., когда в Mozilla заговорили о многозадачности *Firefox* и создали первые наработки, предотвращает сбой браузера в случае падения сайта или приложения. А комбинирование с технологией «песочницы» [sandboxing] повышает безопасность: даже если хакерам удастся внедрить эксплойт в определенную страницу или приложение, они не смогут скомпрометировать браузер и получить доступ к базовой системе и файлам.



› На мировом рынке браузеров доля пользователей *Firefox* в течение последних шести месяцев стабилизировалась на уровне 11,9%.

Однако подобный «старт-стопный» характер работы над Electrolysis никогда еще не приводил к законченным изменениям в архитектуре *Firefox*, хотя отдельные наработки по включению многозадачности добавлялись в «ночные» сборки. Согласно предварительным планам, многозадачность будет реализована в Aurora-сборке *Firefox 41* в двадцатых числах июня и в Beta-канале *Firefox 42* в конце сентября. В стабильной же версии Electrolysis появится не раньше ноября.

Конкурирующие браузеры уже располагают многозадачной инфраструктурой: в 2008 г. с радикальной тогда функцией впервые дебютировал *Chrome* от Google. А *Safari* от Apple и *Internet Explorer* (*IE*) от Microsoft также выполняют раздельные процессы, хотя и не всегда для каждой открытой вкладки.

И в то время, как Mozilla расширяет штат сотрудников, работающих над e10s, инициатива по созданию 64-битной версии *Firefox* для Windows, у которой вышел в марте пре-релиз для разработчиков, может быть законсервирована.

Одна из причин такого решения — неоправданные финансовые затраты: несмотря на постоянно вносимые в *Firefox* улучшения, доля пользователей этого некогда популярнейшего браузера продолжает неуклонно снижаться. По данным Net Applications, в конце мая показатель *Firefox* составлял 11,9% (у *Chrome* — 26,4%). Всего же за последние 12 месяцев *Firefox* потерял около 29% своих пользователей.

СТРАТЕГИЯ

BlackBerry меняет ОС?

Осенью не исключено появление их слайдера с Android на борту.

На сообщение Агенства Reuters, что уже осенью этого года компания BlackBerry представит смартфон-слайдер под управлением Android, реакция специалистов была сдержанной: по их мнению, успех такой стратегии представляется маловероятным. Напомним, что в настоящее время на долю собственной операционной системы (последняя версия — BlackBerry 10) канадской компании приходится 1% рынка смартфонов.

«Будет удивительно, если BlackBerry действительно выпустит телефон на Android, — считает Джек Голд [Jack Gold], аналитик J. Gold Associates и колумнист ComputerWorld. — Телефоны дают очень малую доходность бизнесу компании, а доходность от Android будет еще ниже. И смартфоны на BlackBerry 10 уже позволяют выполнять приложения Android. У BlackBerry множество собственных наработок в области безопасности, включение которых в устройство, в частности, в слайдер на Android с физической клавиатурой и сенсорным экраном, сделает его уникальным». В свою очередь, Каролина Миланези [Carolina Milanesi] из Kantar WorldPanel назвала неудачными все попытки BlackBerry сделать свою ОС популярной. По ее мнению, BlackBerry следовало бы сосредоточиться на возможностях менеджмента и сервисов, «предложив корпоративным пользователям высокобезопасное устройство на базе Android».

Аналитик из Moor Insights & Strategy Патрик Мурхед [Patrick Moorhead] уверен, что BlackBerry надо было обратиться к Android еще несколько лет назад. «BlackBerry не обладает ПО, ноу-хау или размерами, чтобы конкурировать с Apple, Google или Microsoft. Телефон на Android от BlackBerry будет началом конца для BlackBerry OS». Скептично настроен и аналитик IDC Рамон Льямас [Ramon Llamas]: «Если это произойдет, BlackBerry ждет битва между ОС, в которых преуспели немногие. Лишь Samsung смогла выпускать смартфоны одновременно на Android и Windows Phone, а ее планы относительно Tizen OS в течение года менялись несколько раз».

Пресс-служба BlackBerry заявляет, что компания сохранит ОС BlackBerry 10, преимущества которой — в «не имеющих себе равных безопасности и производительности». А про выпуск телефона на Android — «Мы не комментируем домыслы».

› ОС от BlackBerry несравненна по безопасности и производительности, но возможности компании не позволяют ей конкурировать с Apple, Google или Microsoft.

Источник: www.computerworld.com

ВОЗГЛАВИТ LINUX

Воздушный флот будущего

В состав проекта Dronecode вошли новые участники.

К Dronecode, инициативе по созданию свободной платформы БПЛА, продолжают присоединяться все новые участники, что содействует превращению этого проекта в стандарт де-факто для потребительских и коммерческих дронов. Список последних новобранцев включает специализирующуюся на выпуске автопилотов компанию Arsov RC Technology (США) и производителя БПЛА Walkera (Китай). Звание спонсоров Dronecode носят компания OpenRelief, предоставляющая открытые краудсорсинговые информационные решения для ликвидации последствий стихийных бедствий, и фонд Open Source Robotics Foundation.

Некоммерческая организация Dronecode была образована под эгидой Linux Foundation в конце 2014 г. За прошедшее время к ее основателям 3D Robotics (3DR) и Yuneec International присоединились еще 27 организаций и спонсоров, заинтересованных участием в «нейтральной, прозрачной инициативе для продвижения технологии БПЛА» — как охарактеризовала ее вице-президент по маркетингу и программам разработки Linux Foundation Аманда Макферсон [Amanda McPherson]. По ее мнению, вливание в состав Dronecode всё новых членов свидетельствует в пользу модели совместной разработки, позволяющей предоставить многим участникам ресурсы и поддержку уже набравшего силу БПЛА-сообщества.

› Умный дрон 3D Robotics Solo: Dronecode взлетает все выше.

В настоящее время в сообщество Droneshare входят приблизительно 1200 разработчиков, пишущих код для новой платформы. Уже была представлена и ее аппаратная реализация — квадрокоптер 3DR Solo с двумя процессорами Cortex-A9 1 ГГц (один непосредственно в дроне, второй в наземном контроллере). Solo управляется автопилотом 3DR Pixhawk от ArduPilot (еще один проект, поддерживаемый Droneshare), а на Cortex-A9 возлагается обработка «живого» видео в полете, что резко снижает вероятность отказа автопилота.

БИЗНЕС

КНР за безопасность

Китай усиливает контроль над данными.

Американские торговые группы обеспокоены политикой безопасности Китая, которая, по их мнению, способна задушить бизнес в стране. 30 июня в Китае принят новый закон о национальной безопасности, который позволяет государству получить контроль над интернет-инфраструктурой и любыми важными данными. Закон охватывает широкий спектр областей, включая оборону, пищевые продукты и технологический сектор. Полный текст окончательного варианта закона пока еще не опубликован, но общая его направленность, согласно сообщению государственного новостного агентства Синьхуа — дальнейшее усиление кибербезопасности. В соответствии с новым законом, ключевые информационные системы и данные в стране также станут «безопасными и контролируемыми».

Проект закона не расписывал в деталях, что именно будет означать такой контроль. Однако американские торговые группы уже выразили озабоченность тем, что текущая политика безопасности Китая заходит слишком далеко и может подтолкнуть иностранные компании к бегству из страны. Ранее в этом году вызвало жалобы антитеррористическое законодательство КНР, поскольку оно может потребовать от технологических компаний США передачи используемых ключей шифрования правительству Поднебесной. Президент Обама даже обратился к Китаю с просьбой внести изменения в законодательство.

В последние годы, однако, кибербезопасность является одним из приоритетов для КНР. Последней каплей, переполнившей чашу терпения, стала утечка информации через Эдварда Сноудена [Edward Snowden], утверждавшего, что США шпионят за китайскими компаниями. Правительство КНР предупредило о намерении блокировать продажи IT-продуктов, которые не смогут пройти контроль.

Ясно, что изменения политики Китая не сулят ничего хорошего американским технологическим компаниям, особенно поставляющим ИТ-продукты для правительства или государственных компаний. Уже сейчас Китай в значительной степени цензурирует свой Интернет и блокирует многие американские сайты, включая Facebook, Twitter и Google.

› Дом Всекитайского собрания — здание китайского парламента на западной стороне площади Тяньаньмэнь в Пекине.

В ответ на критику КНР заявляет, что новые законы последовали после укрепления своей кибербезопасности рядом западных стран. «Китай всегда поддерживал открытую политику в остальных частях света, — заявил в ходе пресс-конференции 30 июня официальный представитель правительства КНР Чжэн Шуна [Zheng Shuna]. — Мы приветствуем в Китае все соблюдающие законы и предлагающие законные услуги зарубежные предприятия».

Источники: www.theregister.co.uk; www.computerworld.com

0+

7 советов быстрого поиска работы от hh.ru

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

И помните, что работа найдется для каждого!

www.hh.ru

hh.ru HeadHunter

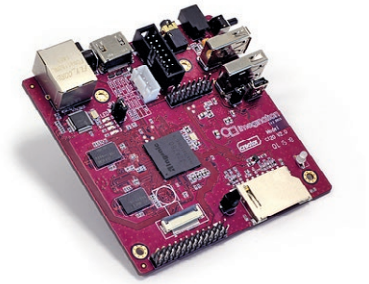
*Данная услуга является платной. Подробнее о ее стоимости и условиях предоставления вы можете узнать по адресу hh.ru/applicant/services

LINUX РАСТЕТ

Освоена MIPS

Fedora готова к тестированию на микрокомпьютере Creator CI20.

Компания Imagination Technologies объявила, что Fedora 22, дистрибутив сообщества, впервые доступен для тестирования на 32-битном процессоре MIPS. Эта версия разработана для микрокомпьютера Creator CI20, с двуядерным процессором MIPS 1,2 ГГц, и содержит все необходимое для запуска Mock — инструментария для компиляции пакетов в системе сборки. Но реализованы не все возможности Fedora: нет GUI, портировано лишь около 7% всех пакетов для родительской версии. Это т.н. образ Stage 3, который загружается в бездисплейном режиме и позволяет войти в систему с помощью SSH или последовательной консоли, а затем использовать DNF для установки и обновления пакетов. Следующим шагом планируется создание, при помощи Mock, полноценного дистрибутива, а финальный этап — развертывание системы сборки Koji-Shadow, используемой для всех других вторичных архитектур в Fedora.



› Макетная плата разработчика Creator CI20 позволит создавать самые разнообразные приложения и проекты.

Комплект макетной платы Creator CI20, выпущенный в конце 2014 г., предназначен для разработчиков Open Source, сообществ производителей и системных интеграторов: его инструментарий допускает создание широкого диапазона приложений и проектов. В основе комплекта — система на чипе (SoC): это центральный процессор, графический стек и ввод/вывод (I/O), а также блок коннектора I/O, совместимый с Raspberry Pi, что допускает подключение ко многим устройствам аналогичных проектов.

ВИРТУАЛИЗАЦИЯ

VirtualBox: Потаскаем!

Обеспечена поддержка Drag-and-Drop между хостом и гостевой системой.

В начале июля, спустя 20 месяцев после выхода предыдущего главного релиза, корпорация Oracle представила первый стабильный выпуск новой ветки *VirtualBox* — мощной и популярной кроссплатформенной системы виртуализации. *VirtualBox 5.0*, безусловно, является одним из самых важных релизов за последние несколько лет: реализовано функционирование гостевых ОС Windows и Linux в режиме паравиртуализации, добавлено множество новых возможностей, не говоря уже о более мелких улучшениях и исправлении ошибок.

Эффективность гостевых ОС улучшена благодаря использованию ими специальных драйверов, а не эмуляции оборудования. Повышена производительность, за счет предоставления гостевым ОС расширенного набора инструкций ЦП — SSE 4.1, SSE4.2, AES-NI, POPCNT, RDRAND, RDSEED, XSAVE, AVX, AVX2. Контроллер xHCI полностью поддерживает интерфейс USB 3.0: гостевая ОС может быть сконфигурирована для работы с USB 1.1, 2.0 и 3.0.

Поддерживается шифрование данных на образах виртуальных HDD в прозрачном режиме, по алгоритму промышленного стандарта AES с 256-разрядными ключами шифрования (DEK), чем гарантируется безопасность данных и их зашифрованность независимо от того, используется в данный момент виртуальная машина или нет. В графический интерфейс добавлена вкладка для настройки параметров шифрования. На всех платформах — Windows, Linux и Oracle Solaris — поддерживается двунаправленный "drag-and-drop" между хостовой и гостевой ОС, чем обеспечивается прозрачное копирование или открытие файлов, каталогов и т.п.

Заявлена поддержка Windows 10, OS X Yosemite, Oracle Solaris и множества дистрибутивов семейства GNU/Linux. LXF

Новости короткой строкой

» Проект Census, выявляющий открытые продукты, нуждающиеся в аудите кодовой базы и оценке защищенности, присвоил самый большой уровень риска (11 из 16) tcpd, whois, ftp и netcat-traditional.
Источник: www.coreinfrastructure.org

» Фонд СПО включил в список полностью свободных предназначенный для использования во встраиваемых системах дистрибутив ProteanOS.
Источник: www.fsf.org

» Звания золотого спонсора проекта OpenBSD (оказание ежегодной финансовой помощи в размере $25–50 тыс.) первой удостоена корпорация Microsoft.
Источник: undeadly.org

» Представлен первый выпуск ориентированной на непрерывную обработку потока данных открытой СУБД *PipelineDB*, основанной на кодовой базе *PostgreSQL 9.4* и совместимой с ней.
Источник: www.postgresql.org

» Очередной openSUSE выйдет в начале ноября под именем openSUSE Leap и с новой нумерацией: 42.x ("x" в номере "42.x" — это номер SUSE Linux Enterprise, основы релиза).
Источник: lists.opensuse.orgl

» Microsoft перевела в разряд свободных компьютерный планетарий *WorldWide Telescope* (*WWT*), чьи источники информации — космический телескоп Хаббл и до десяти наземных телескопов.
Источник: www.dotnetfoundation.org

» Создатель *PackageKit* Ричард Хьюз представил проект Linux Vendor Firmware Service, упрощающий обновление прошивок в дистрибутивах GNU/Linux за счет создания их коллекции.
Источник: blogs.gnome.org

» Oracle прекратила выпуск публично доступных обновлений для Java SE 7, и место лидера проекта OpenJDK 7 заняла компания Red Hat.
Источник: investors.redhat.com

» Компания Code Climate открыла свою платформу статического анализа кода.
Источник: blog.codeclimate.com

» Доступна DVD-сборка релиза LaTeX-дистрибутива TeX Live 2015, с набором установочных файлов для различных ОС, копией репозитория CTAN и документацией.
Источник: tug.org/texlive/

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

softline®

Cloud Software Hardware Services

20+ Years in IT

IT-архитектура вашего бизнеса

корпоративные мобильные решения на платформе Microsoft

SaaS

CLOUD

гибридные облака

SOLUTIONS

SAM

поставка аппаратных решений

юридическая поддержка

remote workplace

PROJECTS

VIRTUALIZATION

LINUX

программное обеспечение

BI & BigData

решения Microsoft

hardware

web-разработка

legal support

education

разработка IT-стратегий

ERP&CRM

защита персональных данных

SERVICES

WinSpace

Mirapolis

CAD CAM CAE

SOFTWARE

публичные облака

САПР и ГИС

информационная безопасность

SECURITY

учебный центр

техническая поддержка

search

flexible

CONSULTING

data leak prevention

разработка информационных систем

корпоративный портал DeskWork

open source

PLATFORM

IaaS

protection

hosting

technical support

PaaS

HARDWARE

частные облака

16+

8 (800) 232-00-23 | +7 (495) 232-00-23 | www.softline.ru | info@softline.ru

Широкоформатая печать

digital print

Графический дизайн

- Полиграфической продукции
- Наружной рекламы
- Фирменный стиль
- Цифровая фотосъемка

Широкоформатая печать

- Наружная и интерьерная
- Стойкая к воде и УФ
- Плоттерная резка, печать на любых материалах

Широкоформатая экопечать

- Печать для рекламы на транспорте
- Печать для метро (стикеры, щиты, световые короба)
- Мобильные рекламные стенды, витрины
- Ширина 1,8 м, разрешение 1440 DPI

Наши преимущества

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

Digital PRINT
(812) 591-79-80
(812) 591-79-81
diprint.spb.ru
Санкт-Петербург,
ул. Смолячкова, 19,
офис 405А
(м. «Выборгская»)

Обзоры

Новинки программного и аппаратного обеспечения в описании наших экспертов

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

LMDE и PPA: еще один довод «за»

Чуть ли не главным преимуществом Mint перед LDME считается возможность подключения к нему PPA-репозиториев Ubuntu, где, как в Греции, все есть. Это правда, чистая правда и одна только правда... но не вся. Ибо PPA-репозитории многих прикладных пакетов можно подключить и к LMDE фирменной утилитой *mintsources* (она же — Источники приложений в секции Администрирование меню Cinnamon и Mate). Хотя штатно в ее LMDE-версии такой возможности нет, это делается через пункты Дополнительные репозитории и Проверка подлинности ключей. Имя репозитория и ключ найдутся, если развернуть пункт Technical details about this PPA на Launchpad-странице пакета. После чего останется обновить локальный кэш и командой **$ apt search [имя-пакета]** убедиться в доступности нового пакета. Что, конечно, не гарантирует его успешной установки в LMDE, но для «простых» пакетов срабатывает — а ведь за разными полезными мелочами и обращаются обычно к PPA-репозиториям: пакеты «серьезные», как правило, есть и в официальных репозиториях Debian, откуда устанавливаются в LMDE штатными способами. И их версии не обязательно старее, чем в репозиториях Ubuntu: часто ситуация оказывается противоположной, особенно с учетом веток Testing и Sid.

Так что «наркозависимость» применителей Mint'а от PPA-репозиториев слегка преувеличена. А возможность ее удовлетворения при «ломке» — еще один довод в пользу равноценности обоих дистрибутивов этого проекта.

alv@posix.ru

Сегодня мы рассматриваем:

Debian 8 Jessie 14
К этому дистрибутиву прочно приклеился эпитет «надежный, как скала». Что же мы получили после двухлетнего ожидания?

Visual Studio Code 16
Microsoft расщедрилась и бесплатно выдала редактор кода. Но, похоже, это не самый лучший подарок.

Сервер от 1&1 17
Масштабируемая облачная платформа, развертываемая по запросу и с оплатой по минутам! Разработчики останутся довольны.

DIR-825/AC и DWA-182 ... 18
За этими загадочными аббревиатурами скрывается тандем роутера и USB-адаптера. Вдвоем они способны побороть микроволновку.

Dell Chromebook 11 20
Молодое поколение стали явно баловать: вот вам еще один хромбук для школьников. Его выносливости позавидовали даже военные.

Fritz! WLAN Repeater 21
Этот ретранслятор расширит зону охвата вашей домашней Wi-Fi-сети, но может послужить и локальным портом для любого устройства.

BioShock Infinite 21
Стрелялка от первого лица, потрясающая зрелищностью и масштабностью сцен, вышла для Linux. Вызволим красавицу из неволи!

Pillars of Eternity 22
Для получения полного удовольствия от ролевки следует прочувствовать ее мир, познав его историю и поняв логику. А на планете Эора это не так-то просто.

› Водо- и пыленепроницаемый, ударопрочный... детям, детям!

Debian 8 Jessie

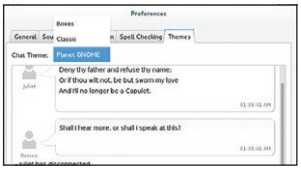
› Рабочий стол Mate навевает спокойствие и умиротворение — настоящая тихая гавань.

Pillars of Eternity

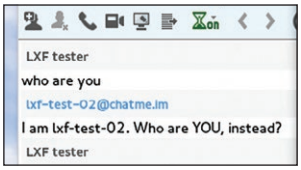
› Действие разворачивается в живописной местности под названием Дирвуд (Олений лес).

Сравнение: Клиенты обмена сообщениями с. 28

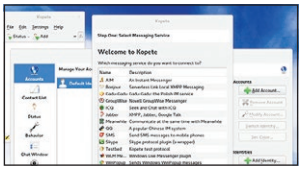
Empathy



Jitsi

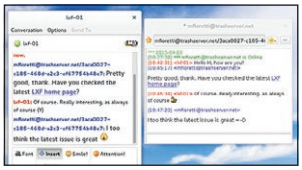


Kopete



Pidgin

Psi



Казалось бы, зачем нам обмен текстовыми сообщениями, если на свете существует *Skype*? Но, как ни странно, у мессенджеров есть целый ряд преимуществ! Не поленитесь разузнать, каких именно.

Debian 8 Jessie

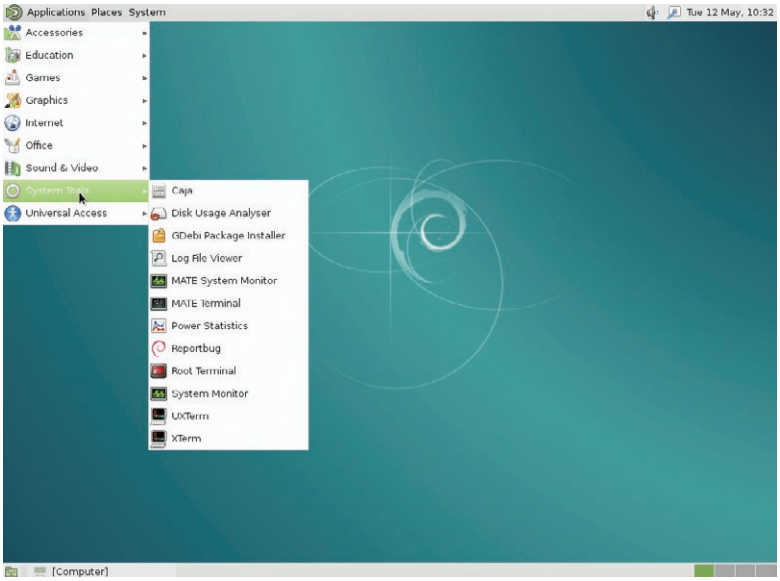
Свежую версию почтенного Debian для классического обзора *LXF* запускает **Джонни Бидвелл**. Кабы он мог, то попробовал бы ее и на зуб.

Вкратце

» Свежая итерация одного из самых уважаемых дистрибутивов. Делался два года, и теперь с витамином D, т.е. *Systemd*.

Неужели с последнего стабильного релиза Debian, с эпохи ядер 3.2, SysVinit и Gnome 3.4, на самом деле прошло два года? Да, прошло, и мир за это время (подобно пальцу Омара Хайяма [отсылка к изречению Хайяма «Двигающийся палец пишет, и, написав, движется далее», — *прим. пер.*]) отнюдь не стоял на месте. Когда Debian Wheezy перешел из ветки тестирования (Testing) в стабильную ветку (Stable) (тем самым став Debian 7.0), началась работа над тестовой версией нового дистрибутива (Jessie). Jessie, в свою очередь, был основан на нестабильной версии другого дистрибутива Debian, известного как Sid. Несмотря на то, что ветка тестирования наделяется новым именем из «Истории игрушек» с каждым релизом (и впоследствии это название переходит на стабильный релиз), нестабильная ветка всегда называется Sid. Новая тестовая ветка называется Stretch, а следующая получит название Buster.

Чтобы насладиться дистрибутивом, мы установили его на компьютер, который в 2006 г. был бы впечатляющим — двуядерный Athlon 64 2.4 ГГц с 2 ГБ ОЗУ и видеокартой Radeon X1650 AGP. Начало установки пролетело с ветерком: часовые пояса, пользователи, и все остальное как обычно. Ручное разбиение диска на разделы отпугнет новичков, но весьма функционально; кроме того, есть опция Guided Partitioning [Разбиение под руководством мастера].



› Рабочий стол Mate навевает воспоминания о безмятежных временах Gnome 2 (потому что именно он некогда был на нашем компьютере).

Если принять параметры по умолчанию, вы получите один огромный раздел ext4 и один маленький раздел подкачки. Этот вариант прекрасно подойдет тем, кому не хватает смелости определить собственную схему разбиения диска. Программа *tasksel* предлагает несколько вариантов рабочих столов и настройки SSH, web-серверов, серверов печати и т.д. Рабочий стол по умолчанию — Gnome 3.14, но его можно изменить на *Xfce*, Cinnamon, Mate, KDE 4 или *LXDE* одним щелчком мыши. Если во время установки у вас будет сетевое подключение, надо будет загрузить только образ первого диска. Оставшиеся пакеты можно загрузить позже, когда базовая система будет готова. Если этой роскоши у вас нет, придется загрузить до семи дисков для установки пакетов оффлайн.

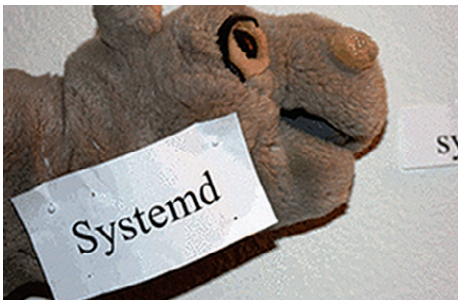
Сначала мы списали черепашью скорость установки базовых пакетов (около часа) на древность «железа», но, похоже, те же симптомы наблюдались и у других пользователей (у части которых «железо» было явно лучше). Мы решили подробнее заняться этой проблемой и установили дистрибутив на виртуальную машину. Все было точно так же — пакеты мигом загружались и медленно устанавливались. Хм-м. Ладно, авось вам предстоит пройти эту процедуру всего один раз. Когда процесс, наконец, закончился, мы обнаружили, что все работает, за исключением Cinnamon, который не поладил со старой видеокартой и утешился программным рендерингом.

Вся базовая установка с Mate и Cinnamon занимает чуть меньше 4 ГБ и включает разумный (хотя и не изобильный) набор настольных приложений: *LibreOffice*, *GIMP*, *Inkscape*, *Rhythmbox*, *Transmission*, *Iceweasel* и *Icedove* — лишь некоторые из них. Последние два — *Firefox* и *Thunderbird* без официального бренда, который

Свойства навскидку

Прочный, как скала

Беспрецедентный объем тестирования гарантирует, что стабильный релиз Debian максимально свободен от ошибок, на каком бы компьютере ни запускался.



Systemd

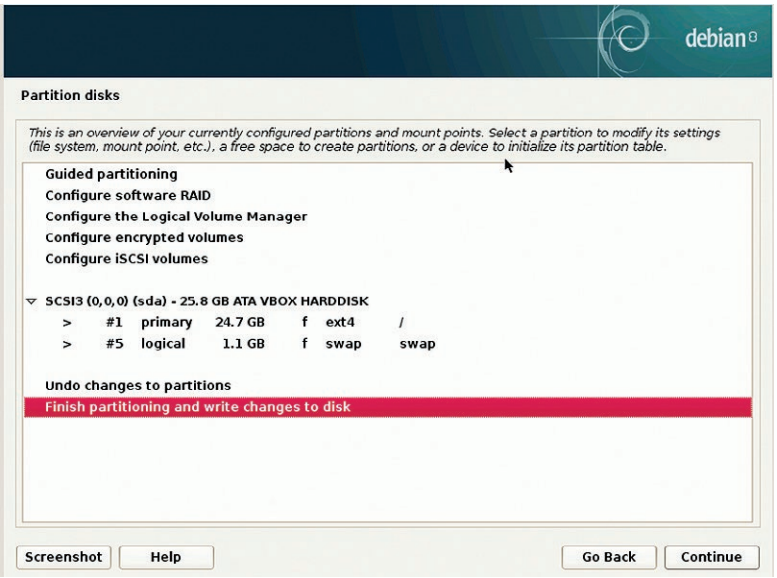
Жребий брошен, и кардинально новая система init теперь принята. Можно вернуться к старому доброму SysVint с помощью утилиты preseed или примириться с 'd'.

не подлежит распространению. У различных окружений есть еще собственные приложения и утилиты.

Номера версий тоже приятно «подросли»: Python 3.2 (но по умолчанию все еще используется Python 2.7), Perl 5.20, PHP 5.6 и Mesa 10.3. При установе пакета *build-essential* вы получите *GCC* версии 4.9 и всё необходимое для компиляции из исходников.

В отличие от Fedora, Arch и Mageia, в Debian не употребляются постоянные имена сетевых интерфейсов, и поэтому у вас по-прежнему будут eth0 и wlan0; многие предпочитают именно такую схему. Поддержка SSL 3.0 была отключена по умолчанию во всей экосистеме Jessie.

› Графический установщик, возможно, и неказист, но дело свое делает… хотя и очень медленно, как показала практика.



Systemd на заре

Мы еще не рассказали об огромном двоичном слоне в нашей системной лавке init. Переход на *Systemd* — это впервые, когда Debian вообще изменил свою систему init. Она была реализована с типичной для этого дистрибутива тщательностью и должна иметь полную обратную совместимость со скриптами *SysVinit* — прежняя структура каталогов **/etc/rd*.d/** сохраняется — поэтому в теории обычный пользователь вообще не заметит перемены. Однако запуск команды **$ pstree** покажет горькую правду: власть перешла к *Systemd*. Есть соблазн пофилософствовать о том, что произошло бы, если бы дебианцы проголосовали не за *Systemd*, а за *Upstart* (тогда, во-первых, Ubuntu ни за что не перешел бы в лагерь *Systemd*), но это лишь фантазии. У нас в свежей системе все прошло гладко, но некоторые пользователи сообщали о различных несовместимостях с *Systemd*; большинство этих ошибок незначительны, и их легко решить, перенеся все пользовательские скрипты инициализации SysV в модульные файлы *Systemd*. Это монотонная работа, но синтаксис *Systemd* гораздо четче, и это сильно упростит вам жизнь в перспективе.

> «Debian славен настраиваемостью, и вы не прикованы к старью.»

Правда, рекомендовать Debian новичкам трудно: даже установив один из главных рабочих столов, вы не найдете в нем столько лоска, сколько, скажем, в Linux Mint или elementary OS. Вики Debian — превосходная документация, но местами написана на языке, отталкивающем непосвященных. Это напоминает активную рекламу для пользователей, тяготеющих к современному: ядро 3.16, используемое в этом релизе, было выпущено год назад, рабочий стол по умолчанию — Gnome 3.14, релиз *Firefox* с расширенной поддержкой (ESR) для *Iceweasel* — 31.6, список можно продолжить. Если все это не выглядит достаточно старым, вспомните, что основные версии останутся такими еще два года. Но Debian славен настраиваемостью, и вы вовсе не прикованы к версиям из старья. Для тестовой версии доступны бэкпорты, которые можно включить добавлением нескольких строк в **/etc/apt/sources.list**, чтобы воспользоваться пакетами поновее.

Если вы охотник до авангардных версий приложений, перейдите на тестовую ветку (Stretch) или даже на нестабильную ветку, если наберетесь смелости. Хотя такое «обновление» (чисто в смысле номеров) проходит отлично, это операция строго однонаправленная. «Обратный апгрейд» с тестовой версии на стабильную или с нестабильной на тестовую сделать невозможно.

Даже если вы предпочитаете более современные дистрибутивы, установку Debian неплохо сохранить на другом разделе диска до момента, когда зависнет ваш суперновый Linux. Debian будет заботиться сам о себе как минимум три года, и, возможно, вы даже решите, что его не стоит держать на полке, или неожиданно обнаружите в себе внутреннего сисадмина, которому захочется взять да поиграть с этим превосходным дистрибутивом. LXF

Debian: 22-летнее наследие

За 22-летнюю историю Debian доказал свою надежность, и его выбирают многие пользователи. Эта надежность — отчасти следствие двухлетнего цикла тестирования и хорошо организованного контроля со стороны Debian Foundation, но отчасти и следствие заморозки ядра и других крупных пакетов перед релизом. Команда ядра Linux поддерживает ряд (сейчас — 7) долговременных веток до версии 2.6.32; исправления ошибок и безопасности портируются в предыдущие версии ядра. Аналогично поддерживаются пакеты ядра, и в огромный труд, производимый командой разработчиков Debian, входит поддержка около 40 000 официальных пакетов стабильными, защищенными и синхронизированными друг с другом.

Помимо возраста и надежности, Debian славен своим Общественным договором [Social Contract], который раскрывает принципы главенства пользователей, обязательств перед сообществом, открытой разработке и верности свободному ПО. Программы, лицензируемые по условиям, по Договору не считающимся свободными, например, плагин Flash, переносят в несвободный репозиторий. Ради них измените настройки *apt* (или обратитесь к утилите вроде *Synaptic*, она все сделает за вас). Это потребуется, например, для установки проприетарных видеодрайверов.

LINUX FORMAT Вердикт

Debian

Разработчик: The Debian Foundation
Сайт: https://debian.org
Лицензия: Разные

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

» Может, здесь есть старые пакеты, но пользователи Debian явно будут смеяться последними, поскольку их дистрибутив останется в седле как минимум до 2018 г.

Рейтинг 8/10

Visual Studio Code

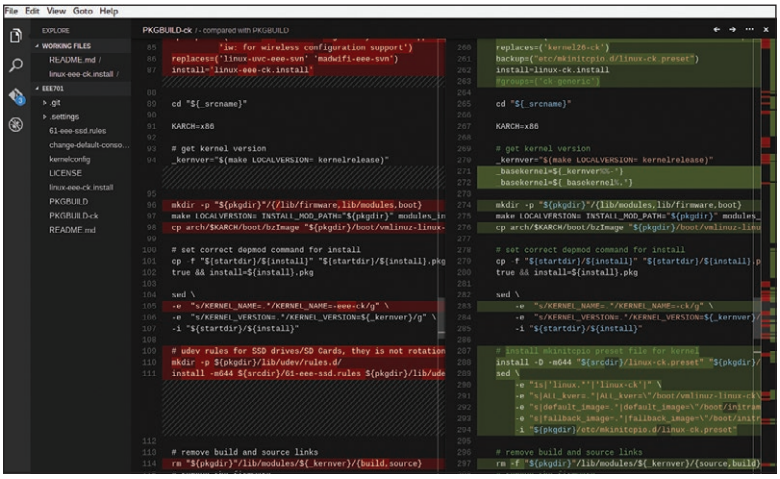
Джонни Бидвелл не скрывает ухмылку: Microsoft сделала что-то для Linux.

Вкратце

» **Кроссплатформенный редактор кода, сочетающий открытую платформу (*Chromium*, io.js, Typescript и Electron) и систему дополнения кода IntelliSense. Немного более продвинутый, чем *Geany* или *JuffEd*, но далек от полноценных сред разработки, таких как Eclipse или полноценный Visual Studio. См. также: *Light Table*, *vi*.**

Под руководством Сатьи Наделлы [Satya Nadella] Microsoft явно заняла гораздо более открытую позицию насчет Linux (который предшественник Наделлы окрестил «коммунизмом»). Мы наблюдали, как компания внесла вклад в Debian, открыла код ряда библиотек .NET, поддержала Linux в своем облаке Azure и анонсировала поддержку *Docker*. Конечно, все это Microsoft делает не в ущерб своим интересам, но все равно приятно. Последний жест компании — кроссплатформенный (с поддержкой Linux) редактор кода *Visual Studio Code*, анонсированный на конференции Build в апреле 2015 г. В Редмонде утверждают, что с данным релизом (и с сообщением о том, что Windows 10 будет поддерживать приложения для Android) компания ублажит разработчиков «там, где они есть».

Все, кто ожидал полной переработки «чудовища» *Visual Studio*, будут разочарованы. *Visual Studio Code* — просто легкий редактор кода, и, откровенно говоря, не самый лучший. При первом запуске вас поприветствует пасмурно-серая рабочая область, где вверху находится строка состояния, выполненная особенно странным оттенком фиолетового. Это — «темная» тема [Dark Theme], которую можно переключить на «светлую» [Light Theme] одним нажатием кнопки. Выбор User Settings [Настройки пользователя] позволит изменить и другие параметры, но в совершенно не-Microsoft'овской манере: откроется пустой файл **settings.json**, а рядом с ним — файл настроек по умолчанию, и вы сможете перегрузить настройки соответствующим образом. На сайте есть хорошие руководства плюс документация о более продвинутых функциях редактора.



› **Сравнить файлы лицом к лицу гораздо удобнее, чем искать различия вручную.**

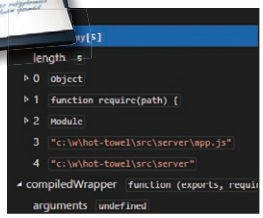
Функции примерно те, каких и ждут от редактора кода: подсветка синтаксиса, вид проекта, автозамыкание скобок. Но есть и неожиданности: поддержка системы автодополнения кода Microsoft IntelliSense и нескольких курсоров, отладка и интеграция с *Git*. Самое же удивительное то, что этот редактор, по сути, измененная версия браузера *Chromium*: он основан на разработанном GitHub открытом фреймворке под названием Electron, использующем io.js и другие web-технологии для создания подлинно кроссплатформенных приложений. Как уже заподозрили скептики, *Code* проприетарен (Стив Балмер [Steve Ballmer] однажды назвал GPL «раковой опухолью»). Как иногда бывает, Adobe тоже выпустила свой редактор *Brackets* [см. HotPicks, стр. 102 **LXF198**], в основном предназначенный для web-разработчиков, но с открытым кодом.

Легковесность функций

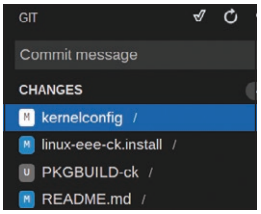
Подсветка синтаксиса в *Code* поддерживает около 30 языков — гораздо меньше, чем *Kate*, *Gedit*, *JuffEd*, *Geany* или любой другой редактор кода, какой ни назови. Более продвинутые возможности доступны лишь для части этих языков. Правда, это лишь пре-релиз, и в будущем ситуация должна улучшиться. Возможность сравнения кода (на рис. вверху) и поддержка *Git* удобны, и их нет ни в одном из других легких редакторов кода. Но в Linux многие разработчики предпочтут запуск **diff** и **git** из командной строки. По правде говоря, при наличии выбора многие предпочтут *vi* или *Emacs*. Слово «легкий» здесь относится скорее к функциям, чем к самому редактору: после распаковки он «весит» около 220 МБ, хотя в основном это объясняется огромной библиотекой *Chromium*.

Поддержка *Visual Studio Code* вошла в версию 0.7 Ubuntu Make, и все пользователи ПО от Canonical могут его пробовать. На момент написания статьи отладчик поддерживает только приложения Node.js и ASP, но обладает всеми должными возможностями по установке контрольных точек, просмотру переменных и пошаговому выполнению программы. Их маловато, чтобы выложить *Code* на «шведский стол» открытых редакторов кода. Если вы жаждете новизны, попробуйте *Light Table*. LXF

Свойства навскидку



Отладчик
Отладка прямо в редакторе здорово облегчает разработку — но поддерживаются только ASP и Node.js.



Поддержка Git
Откройте репозиторий Git, и Visual Studio Code покажет все изменения, не отосланные в репозиторий.

LINUX FORMAT Вердикт

Visual Studio Code

Разработчик: Microsoft
Сайт: https://code.visualstudio.com
Лицензия: Бесплатная (проприетарная)

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

» Неплохой редактор, но есть масса гораздо лучших бесплатных открытых решений, перекрывающих все ваши потребности.

Рейтинг 6/10

Облачный сервер 1&1

Масштабируемая и развертываемая по запросу облачная платформа с поминутной оплатой заставила **Алана Стивенса** азартно двигать ползунки.

Вкратце

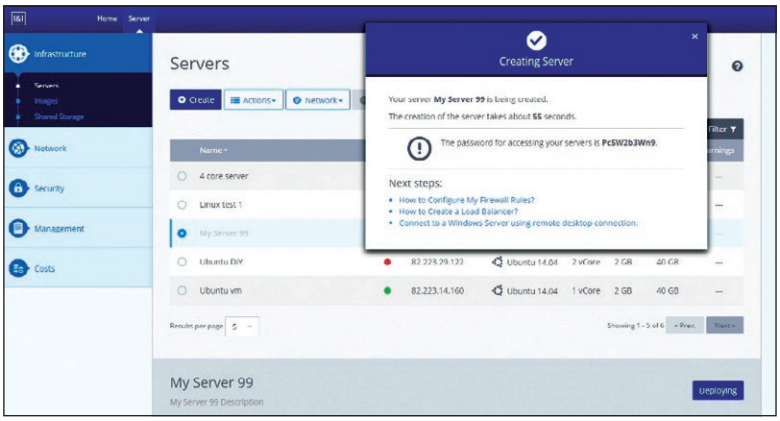
» **Надежный и простой облачный хостинг с доступом к виртуальным машинам *VMware* различных типов, балансировкой нагрузки и распределенными хранилищами.**

Cloud Server от 1&1 предназначен заполнить пробел между дешевыми и простыми хостинг-сервисами и облачными платформами высокого класса от Amazon, Rackspace и иже с ними.

Он делает это не без апломба, предоставляя доступ к виртуальным машинам для самых разнообразных дистрибутивов, к дополнительным приложениям, средствам балансировки нагрузки и распределенному хранилищу — и все это по разумной цене и без контрактной привязки. Облачный сервер несовершенен, в нем есть недостатки, но он вполне доступен, масштабируем, очень удобен и превосходит все ваши ожидания.

Для поддержки нового сервиса 1&1 развернула специализированные дата-центры в США и, что важно для европейских клиентов, один дата-центр в Европе. Основное оборудование для всех дата-центров предоставлено HP — это сервера с гипервизорами *VMware* и хранилищами all-flash от SolidFire, что позволяет компании по умолчанию предоставить своим клиентам быстрые твердотельные диски вместо традиционных жестких дисков. Недостаток этих дисков — ограничение размера до 500 ГБ; но в каждом облачном сервисе можно использовать до пяти таких дисков.

Для жадных до жестких дисков приложений есть возможность смонтировать разделяемые тома SAN размером до 2 ТБ. Кроме того, если понадобится дополнительное пространство, вы всегда можете подключиться к выделенному облачному сервису. Сеть поддерживается преимущественно оборудованием Cisco с заявленной пропускной способностью 300 Гбит/с при подключении к дата-центру, где каждый клиент получает виртуальный ресурс. За безопасностью следят брандмауэры, а благодаря аппаратным устройствам от F5 воможен баланс нагрузки.



› **Точную стоимость своей конфигурации всегда можно увидеть на панели управления.**

Твердые основы

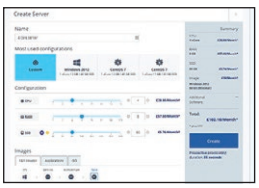
Начните со входа в свою учетную запись 1&1 через панель управления, которая в случае с Cloud Server называется — вы удивитесь — облачной панелью (Cloud Panel). Ползунки управляют тремя основными параметрами — количеством виртуальных процессорных ядер, объемом оперативной памяти и количеством и размером твердотельных жестких дисков; и цена при перемещении этих ползунков вперед-назад соответственно меняется. У каждой виртуальной машины может быть до 16 виртуальных процессоров и до 128 ГБ оперативной памяти, хотя память довольно дорога.

Виртуальные машины для Windows и Linux предоставляются с ОС Ubuntu, CentOS или Debian. Мы создали одноядерную виртуальную машину с 1 ГБ оперативной памяти, твердотельным жестким диском объемом 20 ГБ с CentOS 7, и получилось всего £16 без НДС в месяц. С Windows стоимость возрастает до £28, ибо минимальный объем диска ограничен 40 ГБ, к тому же приходится платить £10 в месяц за лицензию Windows Server. Кроме того, облачные серверы 1&1 тарифицируются только тогда, когда виртуальные машины запущены, с поминутной оплатой. Это может существенно снизить расходы, особенно на серверы для разработчиков, которые не обязаны быть включенными постоянно. Также можно указать набор разнообразных приложений, включая SQL Server и разнообразные CMS-платформы. Другая ключевая особенность — скорость создания новых экземпляров серверов. В прежнем продукте 1&1 Dynamic Cloud Server это могло отнять несколько часов, но теперь от нажатия кнопки Create [Создать] до запуска большинства виртуальных машин проходит всего 55 секунд, что действительно впечатляет.

Наконец, на выделение нескольких распределенных хранилищ SAN и для подключения к нашим тестовым серверам ушло всего несколько минут, хотя стоит отметить, что за эту опцию придется заплатить дополнительно. Ну и последняя полезная особенность — возможность всегда проверить, за что именно вы платите, с помощью интерактивного счета. LXF

Свойства навскидку

Быстрая установка
Создайте учетную запись Cloud Server с оплатой за фактически использованное время, и сможете развернуть сервер за минуты.



Тарификация по минутам
Гибкая конфигурация с поминутной тарификацией позволяет сделать расходы предсказуемыми.

LINUX FORMAT Вердикт

Облачный сервер 1&1

Разработчик: 1&1
Сайт: www.land1.co.uk
Стоимость: От £16 в месяц без НДС

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

» Быстрое и удобное в настройке, масштабируемое облачное решение прекрасно подойдет малому бизнесу.

Рейтинг 9/10

DIR-825/AC и DWA-182

Михаил Чеботарев увлекся построением беспроводной сети на базе роутера от D-Link, невзирая на соседей, злорадно разогревающих обед.

Вкратце

» **Многофункциональный роутер D-Link работает в двух частотных диапазонах и позволяет не только подключиться к Интернету, но и создать на своей базе локальную сеть, а также подключить в USB-порт накопитель, принтер или 3G/4G/LTE-модем.**

Появление смартфонов, планшетов, телевизоров, игровых приставок и многих других устройств, поддерживающих технологию Wi-Fi, позволило сделать нашу жизнь более мобильной. Теперь в Интернет стало можно выходить не только с компьютера, а для подключения нового пользователя больше не надо сверлить стену и тянуть все новые и новые кабели. Технология Wi-Fi-сетей так прижилась, что со временем этот интерфейс появился даже у утюга. Массовое развитие беспроводных сетей требовало качественного «железа», способного выдерживать большую нагрузку и предоставлять устойчивую высокоскоростную связь.

Роутер DIR-825/AC

В январе этого года компания D-Link выпустила двухдиапазонный роутер DIR-825/AC, который поддерживает стандарт 802.11AC, а также имеет в своем арсенале две внешние антенны, USB-порт и четыре гигабитных порта.

D-Link DIR-825/AC поддерживает одновременную работу в двух диапазонах частот, позволяя пользователям подключаться по стандарту 802.11n в диапазоне 2,4 ГГц (до 300 Мбит/с) или в диапазоне 5 ГГц по стандарту 802.11ac (до 867 Мбит/с). Скорость выхода в Интернет, предоставляемая большинством провайдеров, не превышает 100 Мбит/с, и остается большой запас скорости, который можно использовать для своей локальной сети.

Комплектация роутера стандартная: блок питания, сетевой патч-корд (1 метр), инструкция по настройке и сам роутер.

› **Рис. 1. Комплектация роутера.**

Внешний вид роутера сильных отличий от других типовых устройств не имеет. Корпус размером 174×124×33 мм сделан из черного пластика, на нижней панели есть резиновые ножки и крепление на стену; на передней панели — 8 индикаторов и кнопка WPS для быстрого подключения устройств. В роутере установлен процессор с частотой 660 МГц и 64 МБ оперативной памяти, плюс чип флэш-памяти на 8 МБ, хранящий прошивку. Одновременная работа в двухчастотном диапазоне достигается путем использования двух независимых чипов. При тестировании использовалась прошивка 2.5.23, устанавливаемая на устройство изготовителем.

Интерфейс

Роутер имеет многофункциональный web-интерфейс, включающий мастер настроек сети, виртуального сервера и IPTV, а также инструменты тонкой настройки.

Мастер настройки беспроводной сети позволяет быстро ввести необходимые параметры и начать работу с устройством. Поддерживаются русский и английский web-интерфейс. Настройка роутера заняла около минуты, отклик web-интерфейса практически мгновенный. Как и в большинстве роутеров, отображается сетевая статистика. Отдельная вкладка «Мониторинг» отображает полную графическую карту сети, от которой можно перейти к настройкам межсетевого экрана, брандмауэра, IP-адресов и пр.

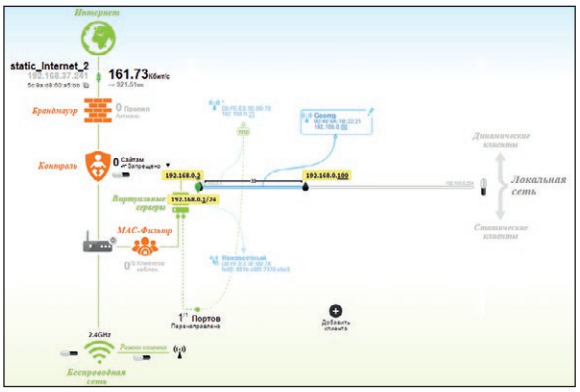
Роутер имеет возможность подключения к Интернету через провод, по WI-Fi или посредством USB-модема (его необходимо приобретать отдельно). Для защиты Wi-Fi-сети можно использовать функцию «MAC-фильтр», создав список пользователей, имеющих доступ, либо же список тех, кому доступ закрыт. Роутер имеет 2 независимые точки доступа; для каждой можно задать свои настройки. Также поддерживается функция мультивещания.

Отдельно стоит отметить встроенную функцию «Безопасный интернет от Яндекс» — бесплатный DNS-сервис, позволяющий в зависимости от режима автоматически блокировать вредоносные сайты и нежелательный для детей контент.

Роутер DIR-825/AC оснащен одним портом USB 2.0. Посредствам USB-хаба через порт к роутеру могут быть подключены несколько устройств, например, USB-модем (список поддерживаемых модемов можно уточнить на сайте производителя; среди поддерживаемых устройств — 3G/4G модемы и некоторые модемы LTE). Возможно подключение накопителей с файловыми системами NTFS, EXT2/3, FAT32, а в случае подключения нескольких устройств хранения будет сформирован один диск. Однако ждать высокой скорости от USB 2.0 не стоит. Для каждого пользователя можно установить индивидуальные права доступа. Для организации доступа к файловому хранилищу через Интернет предлагается использовать FTP, DLNA или Samba.

Адаптер DWA-182

Значительную помощь вам окажет D-Link Wireless AC1200 Dual Band USB Adapter — беспроводной USB-адаптер стандарта



› **Рис. 2. Мониторинг сети.**

› Рис. 3. USB-адаптер DWA-182 избавит сигнал от шумов.

802.11AC, обеспечивающий повышенную скорость передачи данных по беспроводной сети. Проще говоря, DWA-182 позволит «прислушаться» к роутеру более внимательно и выделить информационный сигнал среди множества других, то есть среди шума.

У вас есть СВЧ-печь?

Наверняка вам известно, что исходящие от СВЧ-печки микроволны не только позволяют подогревать пищу, но и создают достаточно сильные помехи для беспроводной сети. Всего одна «микроволновка» выдает электромагнитные волны мощностью около 800 Вт и частотой 2,4 ГГц. Только представьте: всего одна СВЧ-печь вырабатывает такое же излучение, как 10 000 роутеров! Вообразите, что в вашей комнате несколько тысяч роутеров и вы пытаетесь подключиться и работать с одним из них (обратите внимание на то, что частота работы «микроволновки» 2,4 ГГц)... Конечно, мы не греем пищу круглые сутки, и в СВЧ-печи используется двойной защитный экран из стали, но создаваемое зашумление легко почувствовать, даже если роутер отделен от «микроволновки» бетонной стеной, поэтому вы, или ваши соседи, или даже соседи соседей, включив печь, сильно понижают качество сигнала.

Как побороть шум? Никак, разве что обить стены экранирующим материалом и не использовать СВЧ-печи вам и вашим соседям. Конечно, наиболее разумный способ повышения стабильности соединения — повышение качества сигнала от роутера и качества приема сигнала.

Взглянем на цифры: адаптер работает в двух частотных диапазонах и позволяет подключаться к Wi-Fi-сетям на частотах 2,4 и 5 ГГц. Поддерживаемая версия порта — USB 3.0, следовательно, реальная скорость передачи при передачи по USB-порту занижена не будет. Адаптер может работать в диапазоне 5 ГГц, а это означает свободный чистый диапазон, в отличие от зашумленного диапазона 2,4 ГГц. Конечно, у Wi-Fi-сетей в диапазоне 5 ГГц не такой большой радиус действия, зато скорость намного выше (результаты испытаний «в полевых» условиях будут приведены далее).

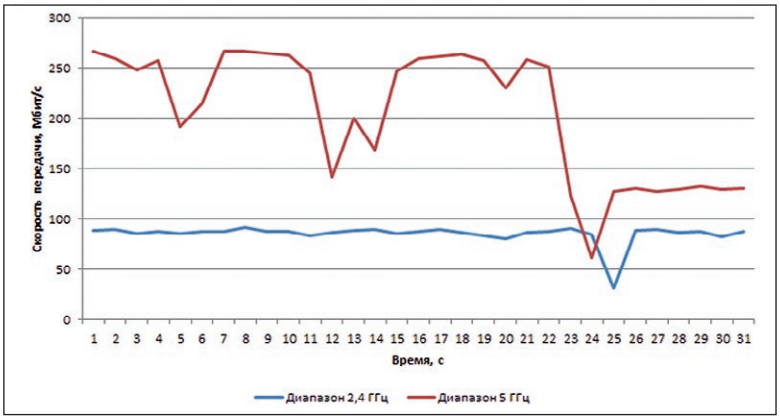
Установка драйверов на Windows сложностей не вызывает. В комплекте с адаптером прилагается диск с драйверами под Windows 8 (для Win7 они тоже подходят); в случае отсутствия дисковода драйверы можно скачать с сайта производителя. Под Linux ситуация чуть сложнее (тестирование производилось в Ubuntu 14.04.3): на сайте производителя драйвера отсутствуют, однако с GitHub можно загрузить репозиторий с исходным кодом и установить их самостоятельно. Установку можно выполнить следующим образом:

```
sudo apt-get update
sudo apt-get install linux-headers-generic build-essential git
git clone https://github.com/abperiasamy/rtl8812AU_8821AU_linux.git
cd ~/rtl8812AU_8821AU_linux
make
sudo make install
sudo modprobe 8812au
```

С помощью перечисленных команд адаптер был установлен и готов к работе без дополнительных настроек.

Вкратце

» DWA-182 — беспроводной USB-адаптер стандарта 802.11AC, обеспечивающий повышенную скорость передачи данных по беспроводной сети. Позволяется подключаться к Wi-Fi сетям в диапазонах 2,4 ГГц и 5 ГГц.



› Рис. 4. Результаты тестирования.

Тестирование

Чтобы исключить влияние оборудования провайдера, а также уйти от ограничений по скорости доступа к сети Интернет, тестирование USB-адаптера DWA-182 и роутера DIR-825/AC проводилось путем замера скорости передачи файлов в локальной сети по Wi-Fi в диапазонах 2,4 ГГц и 5 ГГц.

Выводы

Роутер обладает понятным интерфейсом; рядовой пользователь сможет справиться с его настройкой без особых затруднений.

Конечно, это не самая мощная модель D-Link, однако если вы захотите купить роутер для своего дома, где стоит 1-2 компьютера и имеется несколько телефонов, планшетов и вдобавок, например, телевизор или XBox, то он вполне потянет такой парк устройств. Также скорости его работы будет достаточно для небольшого офиса.

За время тестирования сбоев и зависаний не наблюдалось, все работало качественно и стабильно. Да, он не самый мощный, и заметно греется, но при этом не тормозит и для среднестатистического пользователя вполне подойдет. Перед покупкой важно правильно оценить, насколько большую площадь вы хотите покрыть Wi-Fi-зоной и как много будет одновременных подключений.

USB-адаптер пригодится для использования со стационарным компьютером, если прокладывание витой пары затруднено, либо для ноутбука, если необходимо обеспечить подключение к сети в диапазоне 5 ГГц.

Как видно из результатов теста, скорость передачи в частотном диапазоне 5 ГГц практически сравнима со скоростью передачи по витой паре. Однако дальность связи в этом случае будет чуть меньше, чем при работе в зашумленном диапазоне 2,4. LXF

LINUX FORMAT Вердикт

D-Link DIR-825/AC

Разработчик: D-Link
Сайт: www.dlink.ru
Цена: ~3600 руб.

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

» Надежный многофункциональный агрегат по доступной цене, работающий в двух частотных диапазонах одновременно и притом на высокой скорости.

Рейтинг 9/10

LINUX FORMAT Вердикт

DWA-182

Разработчик: D-Link
Сайт: www.dlink.ru
Цена: ~2000 руб.

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

» Качественный адаптер, отличный сигнал и высокая скорость. Минус: сложности с установкой драйверов.

Рейтинг 9/10

Dell Chromebook 11

Хромбука прочнее **Чуонг Нгуен** еще не видел, но в этом ли счастье?

Спецификации

- **Процессор** 2,6 ГГц, двуядерный Intel Celeron Bay Trail-M N2840
- **Графика** Intel HD Graphics
- **Память** 4 ГБ (DDR3L, 1,6 ГГц)
- **HDD** 16 ГБ SSD
- **Экран** 11,6-дюймовый HD, тачскрин 1366×768
- **Камера** 720p web-камера
- **Связь** Bluetooth 4.0, 802.11ac, dual-band Wi-Fi
- **Порты** HDMI 1.4, 1×USB 3.0, 1×USB 2.0, картридер, дополнительный картридер с замком, 3,5-мм комбо джек
- **Габариты** 297×218×2 мм
- **Вес** 1,32 кг

Заточенный на выживаемость в суровых условиях образовательной среды, Dell Chromebook 11 этого года являет собой надежную систему с довольно скучным дизайном. Своим неказистым видом он напоминает скорее казенный ноутбук для офиса; но в Dell заявляют, что он таким и задуман, чтобы противостоять брызгам, пролитым жидкостям, грязи, пыли и всему прочему, на что горазда школьная среда.

В отличие от Asus, пытающихся переосмыслить идею хромбука своим разворачивающимся на 360 градусов экраном, Dell действует более обстоятельно, создавая долговечный продукт, способный изо дня в день выдерживать и хорошее, и дурное обращение со стороны учеников.

› Не часто увидишь Chromebook, который пригоден к работе в школе аж по оценке военных.

Прочность конструкции Chromebook 11 делает его немного больше и тяжелее своих конкурентов. Весит он практически наравне с ноутбуками 11-дюймовой категории, но его вполне посильно таскать с собой весь день.

На рынке он выделяется своим кондовым дизайном. По заявлениям Dell, эта новая модель прошла типовые испытания на соответствие требованиям военного ведомства США по прочности, сопротивлению грязи, пыли, давлению, температуре, влажности, ударам и вибрации, «так что для школьной жизни пригодна».

Чтобы соответствовать этому, Dell облачили Chromebook 11 в прочный матово-черный пластиковый корпус, на котором не видно царапин, грязи и отпечатков пальцев. Обрезиненные края Chromebook 11 защищают его на случай падения.

В качестве опционного обновления, для хромбука предусмотрен сенсорный экран, предоставляющий ученикам еще одно средство для работы с его содержимым, помимо клавиатуры и трекпада.

Еще одна функция, дающая этому новому Chromebook 11 конкурентное преимущество — усиленное крепление крышки. Помимо того, что оно позволяет разворачивать экран на полные 180 градусов для удобства совместной работы, в Dell его называют особо прочным. Поскольку школьники частенько хватают и таскают Chromebook за крышку, дизайн шарнира позволяет уменьшить давление на петли и препятствовать их слому. Также на шарнире спереди расположены громкие стереоколонки.

Суровая школа

Неожиданно удобной оказалась клавиатура без подсветки. При почти стандартных клавишах — примерно на 1/16 дюйма у́же, чем на 15-дюймовом MacBook Pro — на ней удобно печатать. Клавиши сильно пружинят, но не так, как на полноразмерной клавиатуре. Как и трекпад, она водонепроницаема.

Несмотря на то, что двуядерный процессор Celeron Bay Trail-M не отличается такой производительностью, как у другого ноутбука на чипе Intel Broadwell — гораздо более дорогого Google Chromebook Pixel, работает Chromebook 11 вполне себе ровно.

На практике, по мощности он сравним с Toshiba Chromebook 2. У обеих моделей один и тот же процессор, объем памяти и разрешение экрана. При 15 одновременно открытых вкладках Dell ничуть не тормозил и в целом демонстрировал скорость, отзывчивость и отсутствие задержек. В этой модели Dell, в отличие от других хромбуков, таких как Toshiba, дисплей — не IPS, отчего угол обзора хуже. Да и яркости в нем всего 200 нит, что создает трудности при солнечном свете. Зато звук очень громкий и четкий.

На практике, при яркости экрана 60%, заряда Chromebook хватило чуть больше, чем на 8 часов нормальной повседневной нагрузки, то есть работы в Интернете и электронной почте, редактирования документов и просмотра YouTube.

Примечательно, что у Dell есть простое решение для управления мобильностью, Dell KACE K1000 для управления инвентаризацией, а также передвижные компьютерные столы, где надежно хранятся и подзаряжаются до 30 устройств. LXF

LINUX FORMAT Вердикт

Dell Chromebook 11 (2015)

Разработчик: Dell
Сайт: www.dell.com/uk
Цена: £199

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

» Dell вложили бизнес-функциональность в ноутбук для образовательной среды; вышло лучше всех.

Рейтинг 9/10

Fritz!WLAN Repeater

Нейл Мор нашел легкий способ увеличить зону действия домашнего Wi-Fi.

Беспроводные сети — звучит здорово, хотя на практике это не всегда так. Даже с появлением 802.11ac диапазон «не так уж сильно» расширился, хотя пропускная способность значительно возросла. Помимо проводных сетей, одной из альтернатив является приобретение ретранслятора для вашей домашней или офисной сети, и у Fritz! — известного немецкого производителя — имеется решение для 802.11ac.

› Этот изящный приборчик увеличит зону действия вашей сети.

WLAN Repeater 1750E включает ряд заманчивых возможностей, но ключевой является поддержка двух диапазонов, для расширения существующей 2,4-ГГц и 5-ГГц сети до полноценной 802.11ac, и, поскольку как Virgin, так и BT в настоящее время выпускают маршрутизаторы с поддержкой AC, их доля в своей сфере растет.

Мы испытали Asus EA-N66 dual-band 802.11n 3×3, и он блестяще справился. Подсоединив его к Fritz!, мы тут же отметили увеличение диапазона за пределы зоны действия Asus.

Установить Fritz!, благодаря традиционному web-интерфейсу, легко и просто — правда, клиентское ПО поддерживает только Windows или Mac. Выберите для расширения сети 2,4 и 5 ГГц, введите пароль, и готово. Также поддерживаются WPS-соединения. С подключением Gigabit Ethernet он также будет действовать как локальный порт для любого устройства или маршрутизатора.

Ретранслятор Fritz! имеет встроенный штекер, 155×76,5 мм ширины и 62 мм глубины, что может стать проблемой. Имеется также измеритель уровня подключения, но выключить его нельзя. Но в ценовом плане он вполне выдерживает конкуренцию с проводным соединением и даже другими ретрансляторами. LXF

LINUX FORMAT Вердикт

Fritz!WLAN Repeater 1750E

Разработчик: AVM
Сайт: http://bit.ly/WLANRepeater1750e
Цена: £67

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

» При том, как он увеличивает скорость и существенно расширяет диапазон, мы поражены его производительностью.

Рейтинг 9/10

Спецификации

» **ОС** Ubuntu 14.10, Mint 17.1
» **Процессор** Intel Core 2 2,4+ ГГц, AMD Athlon X2 2,7+ ГГц
» **Память** 4 ГБ
» **HDD** 20 ГБ
» **Графика** Nvidia/AMD, с поддержкой OpenGL 4.1, 1 ГБ VRAM

BioShock Infinite

Нейл Мор ловит кайф — разумеется, законнейшим способом.

BioShock Infinite — потрясающе красивая игра, чего не скажешь о ее сюжете. Это игра, в которую должны играть все, кто может. Мало игр, где аж дух захватывает от масштабности сцен, порой доходящей до абсурда, и все это теперь доступно пользователям Linux. Во вселенной *BioShock* она поднимает вас с морских глубин до самых небес в одержимый культом, ультра-расистский облачный мир, где вы можете спасти положение, а также томящуюся в неволе страдалицу-девицу. Игра старая, вышла в марте 2013 г., и вот, два года спустя, она вышла для Linux, на базе обновленного движка eON Wrapper, создавшего столько проблем для Linux-релиза *Witcher 2* [Обзоры, стр. 21 **LXF191**]. К счастью, с *BioShock Infinite* таких проблем нет: на самом деле, это одна из безупречнейших игр среди тестированных нами за последнее время. Она вдвойне впечатляет своей зрелищностью, даже с ущербной картой Radeon HD 7750. Честно говоря, невозможно отдать должное этой игре в таком кратком формате, так что прочитайте те две с лишним тысячи слов, что ей уделили наши собратья в *PC Gamer* тогда, в 2013 г.: http://bit.ly/LXFbioshock. Осторожно, спойлер: им понравилось. Дождитесь подходящей Интернет-распродажи и урвите одну из лучших стрелялок от первого лица этого десятилетия. LXF

› Bioshock Infinite — первоклассная игра для Linux, которую должен иметь каждый.

LINUX FORMAT Вердикт

Bioshock Infinite

Разработчик: Virtual Programming
Сайт: http://store.steampowered.com
Цена: £20

Геймплей	9/10
Графика	10/10
Продолжительность	8/10
Оправданность цены	9/10

» Развернутый, красивый, немного путаный сюжет доставляет массу удовольствия, в добавление к отличному DLC.

Рейтинг 9/10

Pillars Of Eternity

Евгений Балдин сыграл в действительно новую ролевую игру и заметил отблески из золотого прошлого.

Вкратце

» **Современная партийная ролевая игра «старой школы» с видом от третьего лица.**

› **Путешествовать придется по весьма красивым локациям. Как и в BGI для того, чтобы перейти к следующей, нужно дойти до конца предыдущей.**

В 2012 году американская компания по разработке игр Obsidian Entertainment через сервер Kickstarter предложила окружающему миру сброситься на изометрическую компьютерную ролевую игру с рисованными двумерными задниками. Мир посмотрел на этот призыв и скинулся на более чем $4 (четыре!) миллиона. Причина для этого была. Клич кинули люди, которые участвовали на первых ролях в разработке *Fallout*, *Icewind Dale*, *Planescape Torment* и *Arcanum*, то есть того, что в каком-то смысле составляет прекрасное лицо компьютерных ролевых игр старой школы. Вначале поддержка GNU/Linux из коробки была гарантирована после того, как будет собрано $2,2 миллиона, то есть не сразу, но результат того определенно стоил!

В качестве основы игрового движка был выбран Unity3D, доработанный напильником разработчиков специально для *Pillars of Eternity*. С запуском и работой самой игры не было никаких проблем. Баги в игре все-таки случаются. Я хорошо помню один такой с застреванием партийца в текстуре стены. Поэтому лучше периодически сохраняться, но происходит это крайне редко. Мне сейчас припоминается только один баг. Хотя, может, память меня и подводит, ибо глаза застилает хорошая история и красивая картинка.

Для игры была создана новая вселенная с планетой Эора в главной роли. Дело происходит в местности под названием Дирвуд, который на момент игры официально имеет 200 лет истории. Дирвуд был колонизирован аэдирцами, и, как это случается с имперцами, начало колонизации и было объявлено началом истории. Внезапно выяснилось, что Дирвуд вообще-то был уже заселен теми, кто называют себя глафатанцами, которые вообще-то не особенно против соседей, но крайне негативно реагируют, если эти соседи лезут в местные руины цивилизации, погибшей задолго до того, как глафатанцы сформировались как народ. По этому поводу случились две Войны расколотых камней, которые отличались крайней жестокостью с обеих сторон. В итоге был установлен мир, где побежденные глафатанцы все-таки смогли настоять на некоторых из своих прав. В частности, в Дирвуде было официально запрещено рабство, а всех имеющихся на тот момент рабов выпустили на свободу. Кроме того, был наложен абсолютный запрет на разграбление старых руин. Не всем имперцам из Аэдира это понравилось, и они начали строить козни, что в конце концов надоело официальному победителю Войны расколотых камней Адмесу Хадриету, и он сделал Дирвуд независимым. В этой войне дирвудцы и глафатанцы воевали плечом к плечу уже с имперцами.

Совсем незадолго до повествования в Дирвуд пришла еще одна война — Война Святого. Живое воплощение бога перемен Эотаса, святой Вайдвен, вместе с армией из Редсераса (страна на севере от Дирвуда) пришел, чтобы, по его словам, «освободить Дирвуд». После нескольких поражений на поле боя жрецы богини войны Магран сконструировали специальную бомбу и взорвали святого, что, как тогда казалось, решило проблему. Через какое-то время дети в Дирвуде стали рождаться без душ или пусторожденными. Это явление стали называть наследием Вайдвена и обвинили в нем анимансеров.

К чему был этот экскурс? Чего от компьютерной игры хотят фанаты *Planescape Torment*, помимо одевания и вооружения персонажей, любования задниками,

› **Можно стать «человеком», «островитянином», «дварфом», «эльфом», «лесовиком» или «божественным отродьем».**

уничтожения нечисти и получения за это экспы? Правильно: истории и моральных дилемм. А этого в пустом новеньком мире сделать не получится. Поэтому для получения полного удовольствия придется прочувствовать историю мира, понять его логику и принять его язык. История подается через диалоги и книги. И в начале много непонятного. Собственно, главный герой идет по сюжетной линии не из ненависти и мести, а для того, чтобы получить ответы на вопросы. Коротенькие рассказы из жизни Эоры можно почитать, щелкая на персонажах, отмеченных именами на золотой подложке — это не обязательно.

У каждого, даже у главгада, особенно у главгада, тут своя правда. В каждом поселении и «клубах по интересам» имеется счетчик вашей репутации. В зависимости от его уровня к ГГ меняется отношение в соответствующей местности или штаб-квартире сообщества. Кроме этого, если говорить правду невзирая на потенциальную выгоду, можно прослыть честным человеком, что, в свою очередь, добавляет к вам доверия и позволяет временами избежать ненужной конфронтации.

Уровень развития мира Эоры можно с некоторыми оговорками сопоставить с европейским Ренессансом. Огнестрельное оружие только-только начало распространяться и по сравнению с другими средствами уничтожения отличается избыточной неторопливостью. В отличие от реального мира, наличие душ в Эоре является бесспорным фактом. Их влияние отзывается в материальном мире и наоборот; соответственно, с душами можно экспериментировать.

Анимансия, или изучение душ, в Дирвуде в момент повествования является без всяких «но» передним краем науки. У сообщества анимансеров есть свои рецензируемые журналы, ради статьи в которых не жалко будет и протагониста к креслу привязать, и электрический ток через него пропустить...

› Бои динамические, с возможностью в любой момент поставить паузу, чтобы было время приказать чародею наколдовать магических ракет.

Первое, что надо сделать — это выбрать уровень сложности: Легкий, Нормальный (рекомендуется для начала), Тяжелый и «Путь проклятых». Искусственный интеллект не изменяется с изменением уровня, но меняется число и уровень противников на локациях. Дополнительно можно отключить основное орудие игрока, а именно возможность загрузить игру после смерти персонажа («Испытание железом»).

> «Первое, что надо сделать — выбрать уровень сложности.»

Игра начинается с создания главного героя. На выбор имеются 6 игровых рас (именно они населяют Дирвуд), каждая из которых делится на подрасы:

» **Люди** Привычный середнячок, бонус к силе и решительности.

» **Аумауа** Здоровяки-островитяне с пилообразными зубами и добавочным бонусом к силе.

» **Дварфы** Классические рудокопы, с бонусом к силе и телосложению, но имеют проблемы с ловкостью.

» **Эльфы** Эльфы с бонусом к ловкости и восприятию.

» **Орланы** Лесовики-коротышки с бонусом к решительности и восприятию, но с серьезным недостатком в силе.

» **Богоподобные** Могут быть любой из перечисленных рас; несут на себе уникальный отпечаток богов с соответствующими бонусами и недостатками.

На сегодня доступно 11 классов:

» **Варвар** Классический сильный воин, созданный для быстрых разборок с толпой врагов.

» **Воин** Танк, специализирующийся на выживании в бою, каким бы этот бой ни был.

» **Монах** Специалист по «кун-фу».

» **Паладин** Ослабленная версия воина с аурой, усиливающей сопартийцев.

» **Авантюрист** Классический вор со слабым здоровьем и исключительно сильной атакой.

» **Рейнджер** Специалист по стрельбе из лука с полезным питомцем.

» **Друид** Квалифицированный маг и перевертыш.

» **Священник** Классический клирик: может и порчу навести, и здоровья подкинуть, и посохом приголубить.

» **Волшебник** Носит книгу с заклинаниями, которые, как правило, приходится сначала снять с трупа убитого мага, а затем покупать, но они того стоят.

» **Сайфер** Псионик, манипулирующий непосредственно душой, поэтому на защиту ему плевать.

» **Певчий**, или **Чантер** Воин-бард, его песни дают полезные бонусы.

На самом деле, вовсе нет необходимости следовать «общепринятому» расовому представлению, кем следует быть. Дварф-рейнджер (для одной из подрас дварфов эта специализация является основной), аумауа-интеллектуал (вы встретите и такого) или орлан-варвар (во время Войны расколотых камней главным и самым чтимым полководцем глафатанцев был орлан Регд) — это вполне реальные комбинации.

Основные характеристики:

» **Сила** Отвечает за урон (даже если это урон от выстрела из пистолета), защиту и восстановление выносливости во время боя; иногда используется в диалогах, если нужно кого-то припугнуть.

» **Телосложение** Отвечает за выносливость и стойкость.

» **Ловкость** Увеличивает скорость атаки и реакцию.

» **Восприятие** Добавляет реакцию, уклонение и упреждение в бою, позволяет замечать мелкие детали, что часто используется в диалогах.

» **Интеллект** Используется при разговорах, отвечает за волю, увеличивает область и время действия специальных »

Свойства навскидку

Начало истории
С этого странного устройства и начинается история. В начале ничего не понятно, но в конце все кирпичики повествования найдут свои места.

Их быт, их нравы
Понятие толерантности до этого мира еще не дошло, и развешанные на дереве иноверцы — явление вполне себе ординарное.

› После убийства какого-либо живого существа информация о нем заносится в карманный аналог «Википедии».

эффектов, то есть может быть полезен в том числе и варвару.

» **Решительность** Усиливает волю, уклонение, повышает концентрацию и используется в диалогах.

Следует осознавать, что общение с окружающим миром происходит на основании исключительно характеристик и навыков протагониста, поэтому не стоит создавать глупого или невнимательного персонажа, так как в противном случае число возможных веток диалога весьма значительно уменьшится.

При росте в уровнях можно будет увеличивать уровень навыков:

» **Скрытность** Возможность увидеть противника раньше, чем он тебя, а также начать бой на своих условиях.

» **Атлетика** Уменьшает утомляемость и используется при проверке на умение карабкаться, плавать и продираться сквозь колючие кусты.

> «Не стоит создавать глупого или невнимательного персонажа.»

» **Знание** Используется в диалогах, а также позволяет читать заклинания со свитков, что в сложных боях может оказаться весьма полезным умением.

» **Механика** Обезвреживание ловушек и вскрытие замков.

» **Выживание** Применяется в диалогах и продлевает длительность полезных эффектов от еды и зелий.

Хотя бы один персонаж в команде должен владеть навыком Механика, так как ловушек и запертых сундуков в игре предостаточно. Максимальный уровень встреченных мною ловушек 12. Обратите внимание, что Авантюрист имеет бонус к Механике.

В игре в специальных местах, почти как в *Icewind Dale*, можно нанять (фактически сделать с нуля) напарников, но лучше использовать готовых для путешествия персонажей, которых в достижимой сейчас области действий можно найти 8 штук. Они не клеятся к протагонисту, но активно общаются друг с другом, и каждый имеет свой персональный квест. Во время путешествий можно подружиться с человеком-воином, эльфом-волшебником, человеком-священником, аумауа-певчим, дварфом-рейнджером, богоподобным паладином, человеком-сайфером и орланом-друидом. Будущее расширение *The White March — Part 1* (выйдет этой осенью) добавит голема-авантюриста и человека-монаха.

Каждый персонаж, кроме глобальной линейки здоровья, имеет линейку выносливости. Именно выносливость показывает в бою, дерется ли персонаж или валяется без чувств. После боя выносливость восстанавливается, а здоровье можно улучшить только отдохнув. Если персонаж не мертв, то все повреждения и последствия усталости «печатся» отдыхом. Отдыхать можно в гостиницах или использовать ограниченные припасы для оборудования полевого лагеря.

Как только возникает возможность, следует улучшать оружие. Уровень улучшения регулируется исключительно уровнем партии и наличия подходящих ингредиентов. Кроме улучшения оружия, можно создавать свитки и еду. В бою партию может усилить призыв существ из фигурок, собираемых по чужим сундукам или купленным за очень большие деньги. Десяток дополнительных существ хоть и не самого высокого уровня может кардинально изменить повестку боя.

С приобретением новой обновки обязательно изучайте, кому ее лучше передать. Улучшения от разных элементов одежды не складываются. Если кольцо добавляет +2 силы, а шапка +1, то итоговое улучшение будет +2, а не +3. В результате подобных правил получается своеобразный пасьянс из вещей и героев.

В неизвестной местности нужно постоянно передвигаться в режиме скрытности. Кроме бонусов за заставание врагов врасплох, в этом режиме включается поиск ловушек. В любой момент можно увеличить/уменьшить скорость течения времени, чтобы не ожидать при перемещении из одного конца в другой по изученной локации. Полезно изучить и подстроить клавиатурные команды под себя до начала игры, ибо мышью всего не накликаешь.

Если не хватает уровня атлетики, чтобы влезть на стену, можно воспользоваться крюком и веревкой. Также временами может помочь ломик. От использования эти предметы исчезают, поэтому следует всегда иметь запас — мало ли что.

Основная сюжетная линия занимает едва ли пятую часть от всех доступных приключений. Почти в самом начале придется договориться о владении настоящим замком, который можно улучшить, что не обязательно, но результат хозяйственного отношения к нему отмечается в заключительном эпилоге. Здесь интересно просто исследовать имеющиеся локации из конца в конец с целью отыскания всех секретов и возможностей. Как и в первом *Baldur Gates*, чтобы перейти в следующую локацию, нужно добраться до края предыдущей.

Опыт выдается почти что исключительно за выполнение заданий. Даже при убийстве существ опыт набавляют вовсе не за убийство, а за заполнение соответствующих страниц в местном аналоге Википедии. После заполнения страницы целиком опыт за убийство существа этого вида выдаваться перестает. За убийство людей и им подобных опыта не полагается, зато от разумных остается оружие и броня, которые можно продать. Вещи на продажу можно складывать в бездонный походный сундук. Деньги в этом мире, с одной стороны, не проблема для приключенцев, а с другой — их всегда есть на что потратить. Опыт выдается не на партию, а на героя, поэтому нет необходимости ходить одному. Интересно наблюдать, как герои переругиваются... LXF

Pillars Of Eternity

Разработчик: Obsidian Entertainment
Сайт: eternity.obsidian.net
Цена: 799 руб. в Steam или gog

Сюжет	10/10
Графика	10/10
Увлекательность	10/10
Оправданность цены	10/10

» Современный Baldur Gates, только другой и во многом лучший.

Рейтинг 10/10

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!

НОВОСТИ Вирусы любят деньги » Дефекты оплачиваются » Android влечет злодеев » Обновления отстают

БАНКИНГ

Мобильные угрозы лета

Вирусописатели покушаются на хищения денежных средств.

В июне специалисты компании «Доктор Веб» обнаружили и проанализировали весьма любопытного банковского троянца *Android.BankBot.65.origin*, которого злоумышленники внедрили в официальное приложение для доступа к мобильному банкингу от Сбербанка России. Модифицированная вирусописателями программа рассылалась ими через популярный сайт, посвященный мобильным устройствам, и предлагалась пользователям для установки в качестве «новой» версии приложения. Главная опасность в том, что троянская копия банковского ПО сохраняет все свои штатные функции, и после ее установки на мобильное устройство потенциальная жертва не должна заподозрить какой-либо подвох. В действительности же вредоносная программа после своего запуска незаметно для владельца зараженного Android-устройства загружает на удаленный сервер различную конфиденциальную информацию и по команде вирусописателей способна перехватывать и отправлять SMS-сообщения, что может использоваться киберпреступниками для кражи денег с банковских счетов.

Появляющиеся с завидной регулярностью разнообразные банковские троянцы в настоящее время представляют одну из главных угроз для пользователей ОС Android. Помимо *Android.BankBot.65.origin*, в июне отмечена заметная активность других аналогичных троянцев. В частности, злоумышленники из разных стран продолжили распространять Android-банкеры при помощи нежелательных SMS, содержащих ссылку на загрузку вредоносного приложения. Например, в России под видом поступивших MMS-сообщений вновь распространялись различные модификации троянца Android.SmsBot.291.origin, управляемого злоумышленниками удаленно и предназначенного для кражи денежных средств пользователей.

Другой обнаруженный в июне троянец, *Android.DownLoader.157.origin*, с виду безобидная утилита, которая во время телефонного разговора демонстрирует на экране различную информацию о собеседнике (в частности, его местонахождение, а также название используемого им мобильного оператора). Однако установивших эту утилиту пользователей ожидает малоприятный сюрприз: вскоре после ее запуска в информационной панели ОС Android начинают появляться извещения, внешне похожие на уведомления о входящих сообщениях. Нажатие на эти «уведомления» приводит к загрузке различного ПО, в основном вредоносного.

› Троянец *Android.BankBot.65.origin* встроен в одно из официальных приложений-клиентов для доступа к онлайн-банкингу и распространяется под видом обычного ПО.

БУДЬТЕ БДИТЕЛЬНЫ

Лакомая мишень

На долю Android приходится 97 % всего вредоносного мобильного ПО.

Отчет Mobile Threat Report компании Pulse Secure, в котором на основе исследования 2,5 млн приложений отображено современное положение с мобильным malware, утверждает: ситуация с вредоносным ПО для мобильной ОС от Google продолжает ухудшаться, в 2014 г. количество такого ПО выросло в 4 раза. В настоящее время именно платформа Android является крупнейшей жертвой мобильного вредоносного программного обеспечения, на ее долю приходится 97 % этого ПО.

По данным Pulse Secure, только за 2014 г. стали известны 1268 видов вредоносных программ для Android (464 в 2013 г. и 1030 в 2012 г.). При этом без малого миллион прошлогодних вредоносных приложений по сравнению с 2013 г. составляет увеличение на 391 %. По мнению исследователей, ранее выявивших ошибку Heartbleed (переполнение буфера в OpenSSL), не менее половины вредоносных приложений для Android используют унаследованную уязвимость системы безопасности, вызванную непродуманным повторным использованием кода, в частности, библиотек программ.

Позиции платформы iOS пока гораздо лучше: в 2014 г. на нее были предприняты всего 4 целевые атаки (причем большинство из них были разработаны для открытия на устройствах доступа к системному разделу [jailbreak]). Тем не менее, вирус *WireLurker*, атаки которого на iPhone и iPad начались в ноябре, стал первым примером заражения гаджетов iOS, ограничивающего свой интерес только устройствами Mac.

В связи с резким развитием мобильного malware Трой Веннон [Troy Vennon], директор Mobile Threat Center компании Pulse Secure, обращает особое внимание на риски, возрастающие для корпоративных сетей, в которых все большую популярность получает политика BYOD (Bring Your Own Device, «принеси свое устройство»). Акцент разработчиков мобильного malware на Android и iOS ясно показывает, что они активно пытаются использовать мобильные устройства в качестве слабого звена в безопасности предприятия.

Интересно, что как раз в день выхода отчета Pulse Secure специалисты компании Bitdefender обнаружили уязвимость в Android-версии приложения *Instapaper*, что подвергает опасности персональные данные сотен тысяч пользователей.

Источники: news.drweb.ru; www.pcworld.com

БЕЗОПАСНОСТЬ

Плата за дефекты

Google запускает программу выплат хакерам за выявленные дефекты безопасности Android.

Сообщество Android отличается индивидуальной ориентированностью. Используя этот аспект, Google пытается сделать свою мобильную платформу еще более безопасной. Имидж Android'а не лишен проблем. Среди некоторой части пользователей этой платформы укоренилось (не без помощи конкурентов Google) мнение о ее небезопасности. На протяжении многих лет Google делал многое, чтобы рассеять этот миф. Очередным шагом стала программа, в рамках которой хакерам, исследователям в области безопасности и разработчикам предлагается находить дефекты безопасности в Android.

Бонусная программа безопасности Android [Android Security Rewards program] предусматривает вознаграждение за вклад времени и усилий в процесс повышения безопасности Android. В обмен на отчеты об уязвимостях системы безопасности и тому подобных вещах Google будет предоставлять денежные награды и общественное признание. С одной стороны, разработчики и исследователи получают возможность приработка, с другой — это еще одна возможность для Google продемонстрировать пользователям Android процесс безостановочного обеспечения безопасности и неуязвимости этой свободной мобильной ОС.

Концепция программы Android Security не нова. Google уже запускал подобные инициативы для Chrome OS и других своих продуктов, по результатам которых компания выплатила сотням исследователей в виде премий более $4 млн. «Нас больше интересуют решения, чем поиск ошибок, — пояснил Адриан Людвиг [Adrian Ludwig], ведущий инженер по безопасности Android. — Уровень премии основывается на серьезности ошибок, о которых нам сообщают, и повышается, если участник предлагает полную информацию о том, как он воспроизвел код, что он сделал, чтобы протестировать его, и как он его исправил».

Сейчас программа распространяется на устройства Google Nexus 6 и Nexus 9 под управлением Android 5.1.1 и выше. Что касается более старых устройств, то, по словам Людвига, Google намеревался сосредоточить усилия разработчиков на устройствах, для которых актуальны предложения Google Play Store. Тем не менее, программа не будет затрагивать Android Wear, Nexus Player, а также устройства проекта Tango. Заинтересовавшихся отсылаем за дополнительной информацией к блогу Google (https://www.google.com/about/appsecurity/android-rewards/index.html).

› Плата за страх: Google вознаграждает за вклад времени и усилий в процесс повышения безопасности Android.

НЕ ТОРМОЗИ

Задержка обновлений

Производителям телефонов на Android надо улучшить ситуацию.

Слишком много телефонов на базе Android имеют устаревшее программное обеспечение с дефектами системы безопасности: к такому выводу пришла голландская ассоциация потребителей Consumentenbond. За 2 года в ее испытательной лаборатории на 171 Android-смартфон устанавливали последнее на тот момент обновление программного обеспечения, предоставляемое производителем.

Лишь каждая шестая из протестированных моделей обновилась до версии Android 5.0, выпущенной в ноябре 2014 г., или до более новой. На половине телефонов, даже после обновления, все еще выполнялась версия двухлетней давности Android 4.3 или более старая, известные недостатками системы безопасности.

«Потребители вправе рассчитывать на устройства надежные и безопасные. Производители обязаны предоставлять обновления для своих устройств в течение, по крайней мере, двух лет после их выпуска», прокомментировал данную ситуацию Барт Комбе [Bart Combée], президент Consumentenbond.

У разных производителей значительно различаются подходы к выпуску обновлений. Например, более половины протестированных телефонов от Huawei Technologies работали на версии Android 4.3 или ниже, остальные — на версии 4.4. А пять из восьми смартфонов Motorola Mobility, протестированных ассоциацией, получили Android 5.0 или 5.1. Данных о связи между финальной версией ПО и возрастом телефона Consumentenbond не приводит, однако даже в номенклатуре изделий одного производителя корреляция может отсутствовать: двухлетняя HTC One обновляется до Android 5.0, а вышедшая в мае 2014 г. HTC One Mini 2 обновлений не получит. У Samsung Electronics, крупнейшего производителя телефонов на базе Android, также много моделей, которым актуальное ПО недоступно.

Поддержка конкурирующих ОС, включая iOS и Windows Phone, организована куда лучше. Устройства на iOS поддерживаются в течение приблизительно четырех лет, а Microsoft обещает обновить Windows Phone 8 до Windows 10 (вышла в октябре 2012 г.) на всех телефонах Lumia.

Доклад Consumentenbond дополняют данные от издания Computerworld. На 1 июня 12,4% устройств с доступом к Google Play Store имели Android Lollipop (5.0 или 5.1, спустя 6 месяцев после его выхода), а 39,2% — Android 4.4 KitKat, вышедший в октябре 2014 г. Для обновления Lollipop на своих флагманских моделях основным производителям потребовалось до 147 дней. Компания Google поставляла модель Nexus 6, произведенную Motorola Mobility, с уже предустановленным Android 5.0, а свой флагман Moto X 2014 Motorola обновила через 9 дней. Самой медленной была Sony с моделями Xperia Z3 и Xperia Z3 Compact.

Google быстрее всех обновила и предыдущий флагман, Nexus 5 производства LG Electronics, — через 14 дней. В мае для Moto X 2013 обновлений не было, Android 5.1 Lollipop для него Motorola представила 20 июня. Лидеру рынка, Samsung, понадобился 91 день для обновления Galaxy S5 и 122 дня для Galaxy Note 4. Ранее Samsung загодя обнародовал планы по обновлениям, но перестал это делать: несколько невыполненных обещаний выплеснули изрядный негатив пользователей. LXF

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

Сравнение

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

Обмен сообщениями

Сбросьте оковы проприетарных сетей обмена сообщениями! **Марко Фиоретти** представляет 5 свободных программ для чата в реальном времени.

Про наш тест...

Хороший клиент обмена быстрыми сообщениями ненавязчив и не приковывает вас и ваши контакты к одной определенной программе, устройству или сервис-провайдеру. Со всеми этими требованиями мы решили ограничиться простым текстом. Чтобы вы ни к чему не были привязаны, мы также решили использовать учетные записи и программы XMPP.

Протокол eXtensible Messaging and Presence Protocol (http://xmpp.org, появившийся в Jabber в 1999 г.) для обеспечения обмена сообщениями почти в реальном времени и для управления списком контактов использует XML. Сообщения XMPP должны пройти через сервер XMPP и там интегрироваться, позволяя взаимное общение. (Можно запустить и свой сервер). Мы также выбрали клиенты XMPP, пригодные к установке в качестве родных пакетов в большинстве дистрибутивов. И сознательно сосредоточились на текстовых сообщениях, а не поддержке телефонных звонков и видео.

Зачем в наши дни заморочиваться со Сравнением технологией из 1990-х, когда есть приложения и сервисы вроде *WhatsApp*, *Facebook Messenger*, *Google Talk* и *Skype*? И правда, многие сейчас попросту откажутся использовать нечто иное. Но мы все равно считаем, что когда нужно текстовое общение в реальном времени, представляемые здесь традиционные клиенты обмена быстрыми сообщениями достойны знакомства и применения. Во-первых — как и электронная почта — они никого не заставляют употреблять один и тот же интерфейс или провайдера. Хотя эти клиенты пригодны и для общения, скажем, на Facebook, все они по праву являются настоящими программами, более эффективными и гибкими, чем любой онлайн-сервис. Вдобавок обмен быстрыми сообщениями гораздо меньше подвержен корпоративному контролю и примени́м в частных сетях. Вдобавок вы получите на своем компьютере доступ и к списку своих контактов, и к полной записи своих сообщений — для архивирования, поиска или повторного использования в других программах. И, наконец, все программы в нашем Сравнении отлично подходят для онлайн-обучения, удаленной техподдержки и любой другой деятельности, где нужен обмен скриншотами или документами с другими пользователями.

> «Обмен сообщениями гораздо меньше подвержен корпоративному контролю.»

Наша подборка

- Empathy
- Jitsi
- Kopete
- Pidgin
- Psi

Управление учетными записями

Сколькими своими личностями онлайн можно управлять с удобством?

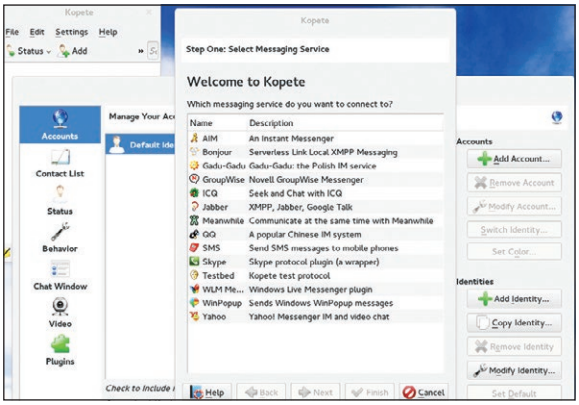
Все программы в нашем Сравнении умеют работать с несколькими стандартными учетными записями XMPP, уже имеющимися на любом сервере XMPP. Сюда входят Facebook и *Google Talk*, если вы будете следовать официальным инструкциям каждой компании для обычных клиентов обмена быстрыми сообщениями. В каждом клиенте также поддерживается создание новых учетных записей XMPP, но на практике это работает только если сервер разрешает. Если вам больше не нужна учетная запись, ее удаление из каждой программы — дело пары щелчков. *Psi* по вашему запросу попытается еще и удалить учетную запись с соответствующего сервера.

Все клиенты, кроме *Psi*, способны обрабатывать учетные записи на многих не-XMPP сервисах и сетях обмена быстрыми сообщениями, без многочисленных нажатий на кнопку или ввода множества данных. Список всех сервисов, поддерживаемых по крайней мере одним клиентом (или по умолчанию, или через плагины), до смешного длинен. Если вы слышали — не говоря уж о применении — обо всех этих сервисах, значит, вы слишком много времени проводите онлайн, и, видимо, вам давно пора отключиться на долгий срок. Просто чтобы вспомнить несколько старых имен — можете использовать *AIM*, *Gadu Gadu*, *ICQ*, *MySpace*, *Mxit*, *Zephir* и даже старый добрый Internet Relay Chat (IRC).

Набор сервисов, поддерживаемых *Jitsi*, всех меньше и всех разнообразнее в нашей подборке. Будучи более других заинтересованным в IP (видео) телефонии, этот клиент знает, скажем, о SIP и *iptel.org*.

Все операции по управлению учетными записями (типа создания, настройки и удаления), которые мы испытали с пятью клиентами, были успешны, с одним исключением: мы не смогли удалить одну учетную запись из *Psi* из-за неведомых «событий, ожидающих очереди на обработку».

По числу поддерживаемых сетей *Pidgin* со своей армией плагинов легко выходит в лидеры. Но *Kopete*, по нашему мнению, дает оптимальный компромисс между поддерживаемыми сетями и эффективным управлением соответствующих учетных записей. С клиентом *Kopete* вы можете (и должны) сперва создать идентификаторы для разных своих ролей: работа, семья, друзья и т. д. Затем каждый идентификатор будет связан (делиться данными) с любым числом учетных записей в любой сети. По умолчанию *Kopete* поддерживает как персональные данные только электронную почту, личный и мобильный телефон. А вот *Jitsi*, наоборот, предлагает уйму полей для заполнения, чтобы публиковать информацию об учетной записи.



› Управление учетными записями в *Kopete* выполняется посредством определения ваших идентификаторов, которые затем привязываются к вашим учетным записям IM.

Вердикт

Kopete ★★★★★
Jitsi ★★★★☆
Pidgin ★★★★☆
Empathy ★★★☆☆
Psi ★★★☆☆

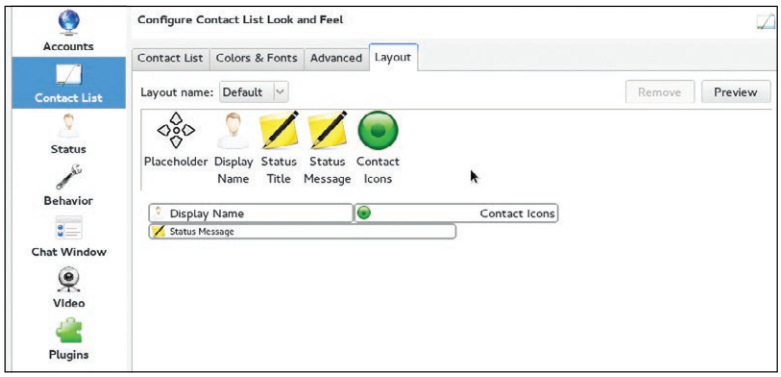
» Работа Kopete с идентификаторами и учетными записями делает его победителем.

Управление контактами

Хорошо организованный список приятелей — абсолютно необходимая вещь.

Как вы контролируете все свои контакты IM? Их добавление — не проблема; тем не менее, как оказалось, клиенты сильно различаются между собой по предлагаемым возможностям отображать и классифицировать контакты. Но все они позволяют по вашему усмотрению присваивать псевдонимы и группировать контакты. *Kopete* делится своими контактами с *Kmail*, *Kontact* и прочими приложениями, используя один и тот же формат, и становится победителем, когда дело доходит до их отображения. Интерфейс списка контактов имеет четыре вкладки, в каждой из которых множество опций, включая: сортировку групп и личностей; включение анимации для онлайн- и оффлайн-статусов; и изменение структуры списка посредством перетаскивания полей. Есть даже плагин Contact Notes для добавления персональных заметок по каждому контакту. *Kopete* и *Pidgin* умеют группировать разные контакты, относящиеся к одному человеку в разных сетях, в один общий «мета-контакт».

Обратите внимание, что *Pidgin* и *Jitsi* могут рассматривать в качестве «контактов для обмена быстрыми сообщениями» заодно и ленты RSS. Иными словами, вы можете найти последние голливудские сплетни, спортивные новости или политические заявления и включить их в свой поток сообщений вместе с приветами от своих друзей и вопросами от своих коллег. Одному вам известно, осчастливит это вас или взбесит, но это очень удобная дополнительная функция.



› Чтобы изменить расположение каждого поля в списке контактов *Kopete*, вам просто надо перетащить их на требуемое место.

Вердикт

Jitsi ★★★★☆
Kopete ★★★★☆
Pidgin ★★★★☆
Empathy ★★★☆☆
Psi ★★★☆☆

» Kopete побеждает, даже если вам придется долго возится с менеджером!

Интерфейс пользователя

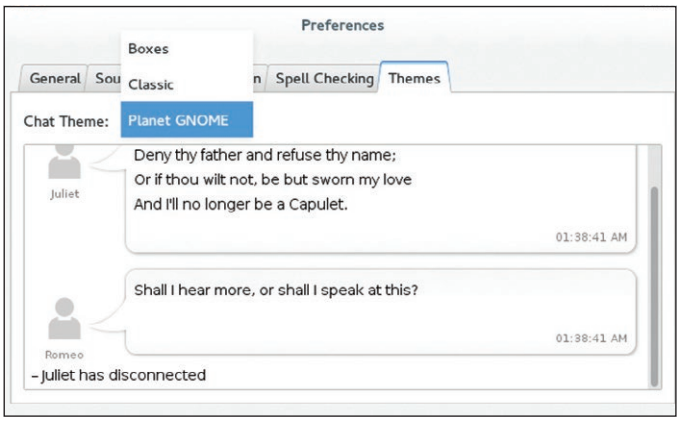
Часто ли, и, главное, *зачем* вам придется использовать обмен сообщениями?

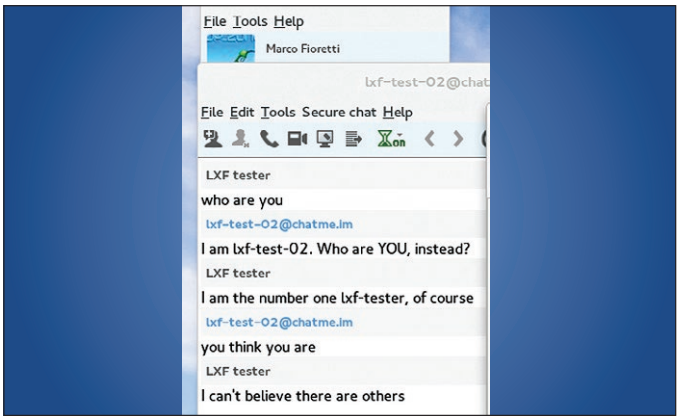
По части клиентов IM есть такие люди, которые вообще не берут в голову пользовательский интерфейс, и такие люди, которые ставят интерфейс во главу угла. Те, что из первой категории, в основном просто ежедневно обмениваются несколькими — правда, важными — напоминаниями и заметками. Даже если их клиент работает нон-стоп, они видят его всего пару минут, пока наскоро строчат ответ. Для такого пользователя важно, насколько быстро он сможет обмениваться файлами и ссылками. Но когда IM используется в качестве основного рабочего или социального инструмента — например, чтобы общаться всю ночь с любимым человеком или с семьей — приоритеты меняются. Такой пользователь хочет видеть нечто привлекательное внешне, чтобы эффективно делиться своими эмоциями, или получить эффективную систему для ведения ряда независимых разговоров одновременно. К счастью, в свободном ПО для каждого найдется что-то на его вкус.

Empathy ★★★☆☆

Empathy — клиент чата/IM по умолчанию для Gnome, и это заметно. Может, поначалу даже слишком. Фактически, он настолько интегрирован, что при первом запуске *Empathy* на рабочем столе Gnome кажется нерабочим, потому что вы обнаруживаете, что смотрите в пустое окно посреди своего экрана. Щелкать негде, и вы гадаете, в чем ваша ошибка; однако, внимательно рассмотрев небольшой виджет, который появляется на панели задач, вы поймете, что именно здесь и размещается меню *Empathy*.

После этого открытия все пойдет легко. У *Empathy* понятный интерфейс со всеми основными функциями под рукой и с приятным внешним видом, который можно изменить с помощью тем Adium (http://bit.ly/EmpathyThemes). Если вы не слишком засиживаетесь в IM-клиенте и вам нужен клиент, который вы можете использовать пять секунд безо всяких усилий и не вникая в протокол OTR, выбирайте *Empathy*.





Jitsi ★★★★☆

Это приложение Java, также работающее на Windows и Mac OS, обладает самым «корпоративным» внешним видом и образом действий из всех участников нашего Сравнения, и больше всех сосредоточено на аудио- и видеочате. Это также превращает *Jitsi* в наименее дружелюбный к пользователю клиент обмена быстрыми сообщениями в нашем Сравнении; но тут не совсем его вина. Правда в том, что все его дополнительные функции изначально сложнее и менее стандартизованы, чем предназначенные чисто для обмена текстовыми сообщениями. Поэтому в *Jitsi* больше опций, и многие из них требуют, чтобы пользователь, как минимум, знал некую терминологию в области телефонии. Однако есть и такие отличные функции, как хороший интерфейс для поиска по Истории Звонков [Call History] и панель для настройки горячих клавиш. Если вы уделите время на ознакомление с *Jitsi*, он станет отличным инструментом для повышения производительности.

Интеграция с рабочим столом

Всегда ли ваш клиент под рукой, не путаясь при этом под ногами?

Все настольные программы должны хорошо интегрироваться со средой рабочего стола. С этой точки зрения, *Empathy* и *Pidgin* созданы для Gnome, а *Kopete*, как ясно по начальной букве его названия, отлично впишется в любую тему KDE. *Psi* настолько компактен, что вряд ли его внешний вид имеет значение. *Jitsi* — программа, которая выглядит наиболее «чужеродно» на любом рабочем столе Linux, но не в плохом смысле.

Если оставить в стороне внешний вид, самым важным в интеграции IM являются уведомления: маленькие диалоговые окна, которые всплывают для отображения новых сообщений. Хорошо ли эти удобные виджеты выглядят и работают в Linux, сильно зависит от того, какой менеджер окон или рабочий стол вы используете.

При этом все программы, кроме *Jitsi*, имеющего собственную систему уведомлений, работали на наших Gnome или KDE более или менее одинаково; однако *Pidgin* и *Kopete* по ряду причин выделяются. Всплывающие окна libnotify в *Pidgin* позволяют настроить сочетания новых сообщений или бесед, уведомлений о подписках или отписках от ваших друзей, которым позволено прерывать вашу работу.

Kopete также умеет прекращать все уведомления, кроме относящихся к выделенным контактам. Кроме того, с помощью плагина Highlight можно дать команду вас беспокоить, только если сообщение в групповом чате соответствует указанному вами регулярному выражению. *Kopete* еще может регистрировать определенные сообщения или запускать индивидуальный скрипт, когда приходят сообщения.

Вердикт

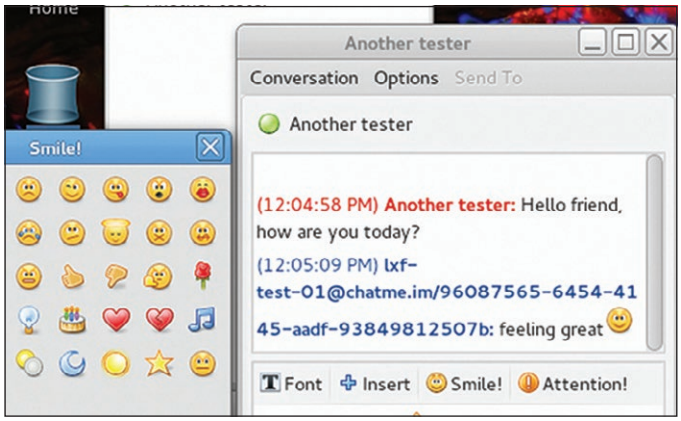
Empathy ★★★★☆
Kopete ★★★★☆
Pidgin ★★★★☆
Jitsi ★★★☆☆
Psi ★★★☆☆

» Ничья между Kopete и Pidgin: оба предлагают массу индивидуальных настроек.

Kopete ★★★★☆

Вы любите аскетичные, традиционные приложения рабочего стола? Обратитесь к *Kopete* — он так и кричит «Старый добрый KDE!» всеми своими пикселями. Может быть, *Kopete* и не очень стильный, но его цель — быть конкретным, эффективным и очень гибким одновременно.

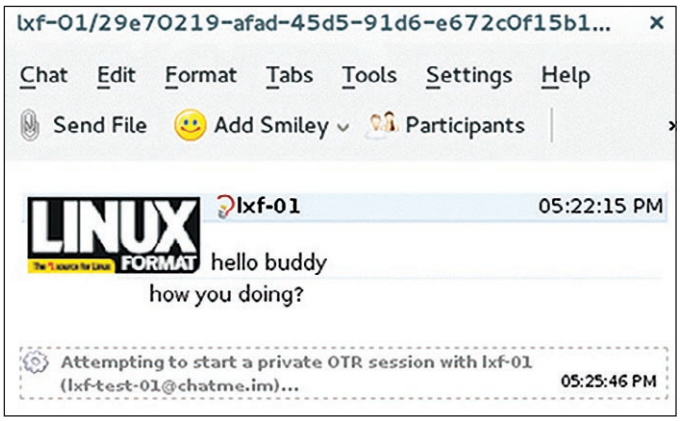
Вы можете настроить практически все, если не прочь потратить на это некоторое время и, возможно, повозиться с файлами исходника. Например, стиль окна чата можно изменить через таблицы стилей XSL и CSS, и подобным же образом можно обзавестись собственными эмотиконами. Даже панель настройки уведомлений ошеломляет. Вот почему, как нам кажется, в *Kopete* меньше плагинов, чем в *Pidgin* — в них просто нет нужды. Однако имеющиеся плагины обеспечивают полезные функции: от автоматического перевода и мгновенной проверки орфографии до статистики ваших бесед и интеграции с Адресной книгой KDE, и все совершенно бесплатны.



Psi ★★★☆☆

Посмотреть на *Psi* было интересно, но иногда он сильно разочаровывает. С точки зрения его достоинств, он доступен почти на двадцати языках и на нескольких операционных системах, и обладает гибкостью, сравнимой с *Kopete*, хоть и не в столь огромном объеме. Раскладка по умолчанию основного окна Contacts является неплохим компромиссом между компактностью и функциональностью, а интерфейс поддается настройке множеством способов, и вы можете подбирать его свойства — от цвета до смайликов и клавиш быстрого запуска.

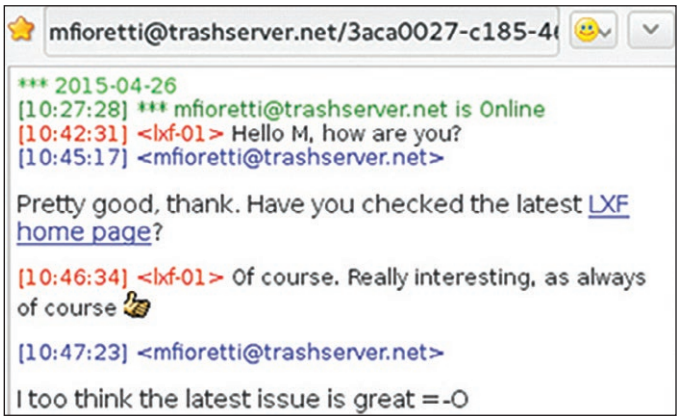
К сожалению, мы также обнаружили некоторые странности, которые затрудняют его использование. Одной из них оказалось периодическое «замерзание», а другой — то, что список друзей из Facebook невозможно свернуть, и его приходится прокручивать целиком, чтобы добраться до других списков друзей.



Pidgin ★★★★★

Пользовательский интерфейс *Pidgin* по умолчанию не загроможден излишествами и прост в навигации. Если какой-либо функции вам не хватает, на официальном сайте *Pidgin*, скорее всего, найдется подходящий плагин. Но будьте внимательны: некоторые пункты, имеющиеся на сайте, уже успели совершенно исчезнуть из Интернета или доступны исключительно для Windows.

Зато версия Linux позволяет использовать альтернативу *Pidgin* для командной строки, именуемую *finch*. Говоря о *Pidgin*, мы также должны упомянуть pounces, очень удобную и дружелюбную функцию. Разработчики описывают pounces как нечто «очень похожее на макрос или записанные действия». Когда вы определите pounce, *Pidgin* его выполнит, если статус ваших контактов изменится. Pounce могут быть чем угодно, от отправки сообщений до запуска скрипта.



Голосовые и видеозвонки

Можете ли вы общаться лицом к лицу, если текста недостаточно?

Хотя мы сосредоточились на обмене текстовыми сообщениями между людьми в реальном времени, нельзя игнорировать тот факт, что многие при упоминании о цифровых коммуникациях в первую очередь думают о голосовых и видеозвонках. И *Pidgin*, и *Empathy* — на бумаге — имеют хорошую поддержку этих функций. Но вам потребуются специальные настройки, и простота в работе с ними и надежность чересчур зависят от всяческих комбинаций версий программ и настроек всех учетных записей IM, чтобы обсуждать их здесь. Если вы хотите применять эти клиенты именно так, готовьтесь потратить некоторое время на онлайн-поиск информации по каждому из них и по вашей учетной записи IM.

Если вы упорно хотите совершить телефонные или видеозвонки с помощью свободных программ, которые также поддерживают стандартный обмен сообщениями, советуем попробовать *Jitsi*, предложив всем вашим друзьям и родным сделать то же самое — возможно, через учетные записи у одного и того же провайдера.

Это самое безопасное решение, как с чисто практической точки зрения, так и по соображениям безопасности. Звонки между пользователями *Jitsi* будут иметь голосовое и видеошифрование по т.н. технологии ZRTP (см. https://jitsi.org/Documentation/ZrtpFAQ), разработанной тем же автором, что и PGP. Вы также сможете использовать *Jitsi* для разговоров с пользователями в других сетях, от SIP до *Google Chat*, но не попробовав, вы об этом наверняка не узнаете. Желаем вам удачи в этом эксперименте.

Вердикт

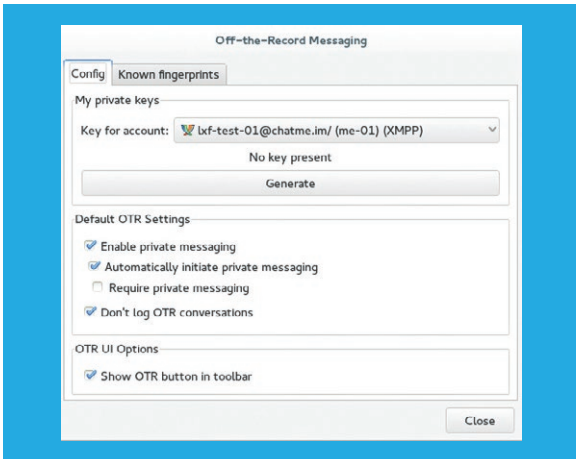
Jitsi ★★★★☆
Kopete ★★★☆☆
Pidgin ★★★☆☆
Empathy ★★☆☆☆
Psi ★★☆☆☆

» Для тех, кому нужны голосовые и видеозвонки, лучшим выбором будет Jitsi.

Безопасность

Осторожно! Мало ли кто тебя слушает!

Как уже упоминалось во введении, одной из важных причин использовать учетные записи и клиенты IM в реальном времени является обеспечение большей конфиденциальности, чем внутри «запертых садов» типа *Skype* или *WhatsApp*. Чтобы взаправду достичь этой конфиденциальности, надо удовлетворить трем требованиям. Первое, и самое сложное, заключается в том, чтобы убедить тех, с кем вы хотите беседовать частным образом, установить нужную программу, настроить и использовать ее как полагается. Второе требование состоит в том, чтобы вы и сами это сделали, а третье требование — самое простое: не оставлять следов своих бесед на компьютере.



› Интерфейс настройки OTR в *Pidgin* настолько прост, что грех им не воспользоваться.

Empathy — самый худший выбор в аспекте безопасности. По своему дизайну он поддерживает только шифрование SSL. В наши дни самым обычным выбором для шифрования IM является протокол OTR (Off The Record messaging), который работает в основном так же, как PGP и GPG, т.е. вы и ваши контакты создаете ключи шифрования, обмениваетесь ими в безопасном режиме (желательно оффлайн) и даете команду своим IM-клиентам их использовать. Все эти процедуры хорошо поддерживаются и относительно просты в использовании во всех остальных клиентах нашего Сравнения, как и во многих других.

Помимо ключей, есть еще пароли учетных записей. На этом фронте выделяется *Jitsi*, поскольку данный клиент не только хранит ваш зашифрованный пароль, как и другие, но и предлагает мастер-пароль для доступа ко всем остальным паролям. Копирование всех файлов настройки OTR и прочего на другие компьютеры позволяет использовать те же самые ключи, с теми же самыми учетными записями, на всех этих компьютерах, однако должны предупредить, что вы делаете это на собственный страх и риск!

Вердикт

Jitsi ★★★★☆
Kopete ★★★★☆
Pidgin ★★★★☆
Psi ★★★☆☆
Empathy ★☆☆☆☆

» OTR, шифростандарт де-факто для свободных/открытых IM, в Pidgin проще всех.

Управление информацией

Обеспечит ли он требуемое вам общение, помогая работать эффективнее?

Одним из важных аспектов использования IM является степень контроля, получаемого над тем, объявлять или нет о своем присутствии в Сети. По умолчанию *Jitsi*, *Empathy*, *Psi* и *Kopete* предлагают базовые опции. Статусы *Pidgin* для каждой учетной записи отдельно можно установить в Available [Доступен], Away [Вышел], Extended Away [Вышел надолго], Do Not Disturb [Не беспокоить], Chatty [Настроен на разговор] и многое другое. Все программы позволяют создавать новые индивидуальные статусы и настраивать к ним сообщения. *Pidgin* также предлагает хороший выбор «настроений пользователя», от Anxious [Встревожен] до Intoxicated [Пьян].

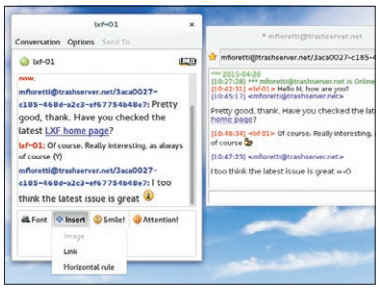
В контексте данного Сравнения управление информацией — это просто удобный обобщающий термин для всех типов стримов, контентных и обычных данных, которые вы отправляете, получаете или обрабатываете во время беседы в IM.

Насчет этого мы можем дать вам только общее представление, поскольку ответ будет зависеть от сочетания клиента, рабочего стола и учетной записи XMPP, применяемых для каждой беседы.

Например, смайлики — штука несложная, но нет никаких гарантий, что они будут выглядеть одинаково во всех клиентах. А вот передача файлов и обмен уведомлениями, похоже, отлично работают во всех клиентах.

То же относится к созданию и использованию групповых чатов или чат-румов, если используемые контакты находятся или на обычных учетных записях XMPP, или все используют одну и ту же нестандартную сеть. Все клиенты могут вести несколько разговоров одновременно, каждый в собственной вкладке, и позволяют настроить окно чата разными способами. Например, *Psi* имеет панель для настройки цветов контактов и определенных слов в тексте, чтобы быстро находить нужные части разговора, а темы *Empathy* могут изменить его вид до неузнаваемости.

По нашему опыту, наилучшие клиенты для управления информацией — *Pidgin* и *Kopete*. В них, например, можно быстро проверить, какие из ваших контактов готовы к шифрованному разговору; настроить их отображение в групповом чате; и автоматически сократить или добавить в закладки URL. Область, где *Kopete* и *Pidgin* по-настоящему блистают — их взаимодействие с другими программами в вашей системе. Вы можете обеспечить расширенный доступ к своему рабочему столу, пересылать сообщения через любую программу и вывести на новый уровень такие вещи, как, например, домашнее задание по математике, выполненное с помощью Интернета, встроив в ваши сообщения математические формулы в формате TeX.



› Иногда IM-клиенты плохо справляются с простыми вещами — им, например, трудно сочетать смайлики в *Pidgin* и *Psi*.

Вердикт

Kopete ★★★★★
Pidgin ★★★★★
Jitsi ★★★★☆
Empathy ★★★☆☆
Psi ★★★☆☆

» Не пожалейте времени на его настройку, и Pidgin повысит вам эффективность общения.

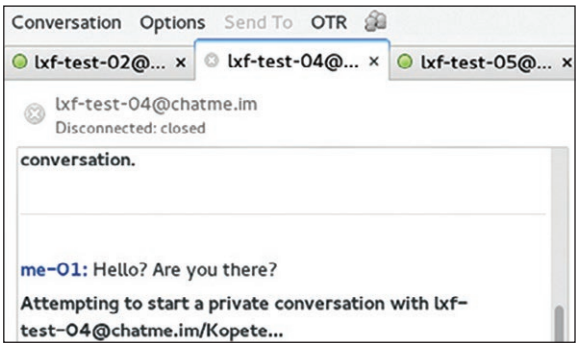
Клиенты обмена сообщениями

Вердикт

Учетные записи XMPP свободны, и на многих интегрированных провайдерах для них даже не нужна учетная запись электронной почты. Клиенты рабочего стола XMPP также доступны для всех ОС и дружественных мощных пейджеров. Есть даже основанные на XMPP приложения обмена быстрыми сообщениями для мобильных устройств. Почему же тогда ими не пользуются все? Возможно, предпочтение отдается *Facebook Chat* или привязанным к телефону сервисам вроде *WhatsApp* — только потому, что они освобождают от скучной необходимости знакомиться с инструкциями и вручную вводить контактные данные.

Более серьезная проблема — взаимодействие. Увы, обмен быстрыми сообщениями не настолько универсальный цифровой метод, как скажем, электронная почта. Представьте, какой была бы жизнь, если бы форматирование, вложения или подписи сообщений электронной почты работали так, как вы рассчитываете, только среди пользователей одного и того же клиента, да еще и одного провайдера почты, и вы нас поймете. Как бы хорошо какой-то клиент ни служил вам, вы наверняка окажетесь в такой ситуации, в отличие от электронной почты, когда для вас он не сработает. И причиной почти наверняка будет то, что у вас и ваших контактов разные сочетания клиента, плагина и ОС.

С этой жесткой оговоркой мы объявляем *Pidgin* победителем. *Kopete* отстал только потому, что способен устрашить новичков изобилием — поклонники *Kopete* могут считать это побочным комплиментом — параметров для первичной настройки. *Pidgin*, возможно, даже более настраиваемый — через плагины; вдобавок и его настроек по умолчанию вполне достаточно, чтобы удовлетворить большинство пользователей. И *Psi*, и *Empathy* — хорошие программы IM, но мы нашли *Psi* нестабильным, а отсутствие поддержки OTR портит дело; поэтому третье место занял *Jitsi*. Если не меньше, чем чисто текстовый IM, нужна альтернатива *Skype* для голосовых и видеозвонков, берите *Jitsi*. Но он не похож на канонические IM, и это осложняет.



› Вкладки, шифрование OTR, смайлики и тонны плагинов для общения по любому типу: *Pidgin* нам нравится.

> «Pidgin настраиваемый, но и по умолчанию удовлетворит большинство.»

I Pidgin ★★★★☆

Сайт: https://pidgin.im **Лицензия:** GPL 2+, GPL 2, MIT **Версия:** 2.10.10

» Простой в использовании, но при необходимости и очень мощный.

II Kopete ★★★★☆

Сайт: http://kopete.kde.org **Лицензия:** GPL 2+ и GFDL **Версия:** 14.12.3

» Умеренный, но солидный, и с бóльшим числом опций, чем хочется настраивать.

III Jitsi ★★★★☆

Сайт: www.jitsi.org **Лицензия:** LGPL **Версия:** 2.8

» Впечатляющие функции, хотя большинство их — для голосовых и видеозвонков.

IV Psi ★★★☆☆

Сайт: http://psi-im.org **Лицензия:** GPL 2+ **Версия:** 0.15

» Не столь полный, как Pidgin или Kopete, и чуть менее стабилен.

V Empathy ★★★☆☆

Сайт: http://live.gnome.org/empathy **Лицензия:** GPL 2+ **Версия:** 3.10.3

» Хорошая программа, хоть и на пятом месте. Увы, не понимает OTR.

Обратная связь

Часто ли и зачем вы используете в наши дни старые добрые IM, и для каких задач? Присылайте нам свои мнения на lxf.letters@futurenet.com.

Рассмотрите также...

Есть много других бесплатных и открытых клиентов для обмена быстрыми сообщениями в Linux. И это хорошо — причины мы уже объяснили в вердикте: возможно, иногда вам придется использовать более одного IM клиента. Полуофициальный список таких программ имеется на https://xmpp.org/xmpp-software/clients. Настоятельно рекомендуем ознакомиться с этим списком, уделив особое внимание приложениям для Android и iOS. Пусть даже лично вам они не нужны — если вы убедите своих друзей и, что куда важнее, свою семью, их использовать, то, возможно, больше не будете вынуждены общаться с ними через проприетарные закрытые сервисы. Конечно, до перехода к реальному общению следует для обеспечения конфиденциальности и переносимости использовать шифрование OTR с временными ключами, и хранить эти ключи в безопасном месте.

Еще один важный пункт для рассмотрения — серверы XMPP. Если вам нужна очень защищенная сеть для обмена быстрыми сообщениями, установите один из серверов из списка на https://xmpp.org/xmppsoftware/servers. LXF

СОБЕРЕМ LINUX-ПК

Собрать ПК не так уж сложно, чего не скажешь о процессе выбора компонентов. **Нейл Ботвик** рассматривает разные возможности.

Уже лет пятнадцать по Сети блуждает документ под названием «Если бы ОС были авиакомпаниями». Раздел, посвященный Авиалиниям Linux, гласит: «При входе в самолет вам дают кресло, четыре шурупа, гаечный ключ и руководство по сборке кресла seat-HOWTO.html». Шутка старая, однако, будучи пользователями Linux, мы больше привыкли иногда делать что-то самостоятельно, чем пользователи других ОС.

И это не обязательно плохо, ведь мы лучше понимаем наш компьютер. Тем не менее, когда вам нужен новый ПК, можно просто выйти в магазин (возможно, онлайн) и купить его. Так почему же некоторые предпочитают создавать собственный компьютер? Далее мы попытаемся ответить на этот вопрос, как и на вопросы посложнее, которые возникнут, когда вы попытаетесь это сделать, например: насколько это вообще сложно? Какие имеются риски? Как насчет гарантий? Позволит ли это сэкономить деньги? Можно ли собрать компьютер без Windows? И еще немало других вопросов. Есть ряд причин, по которым вам, возможно, захочется самим собрать компьютер, и не последняя среди них — удовлетворение от понимания своего компьютера; но эта информация будет полезна не только в том случае, если вы хотите с нуля создать новую систему. Многое из того, о чем мы здесь расскажем, пригодится вам, если вы решите обновить уже имеющийся компьютер. Мы займемся настольными системами, работать с которыми очень просто — лично мне LEGO показался намного сложнее. Ноутбуки — дело другое, однако многое, касающееся выбора подходящих совместимых с Linux компонентов, применимо и к ним, и в конце мы обсудим выбор ноутбука или иного подобного устройства, например, одной из систем неттопов.

> «С настольными системами работать просто — LEGO показался мне сложнее.»

Зачем собирать собственный компьютер? Может, вам и удастся что-то сэкономить, подобрав компоненты самостоятельно, но не стоит на это сильно рассчитывать. Что вам действительно надо сделать — это точно выбрать

› Вы можете приобрести один из этих и получить то, что сборщик счел нужным вложить, а можете собрать свой и получить именно то, что вам нужно.

Помните закон Мура

Закон Мура гласит, что мощность компьютеров удваивается каждые два года. Если вы не планируете собирать новый компьютер ежегодно, то надо учесть рост своих потребностей в течение его срока жизни. Всегда оставляйте возможность для увеличения мощности: если вам нужно 16 ГБ памяти, покупайте материнскую плату на 32 ГБ и заполните ее наполовину. Использование накопителя тоже будет увеличиваться. Объем хранимой информации тоже будет увеличиваться. Видео 4K уже на подходе, так что собирайте систему, в которой будет больше отсеков для накопителей и коннекторов SATA, чем вам нужно на данный момент. Увеличить память и объем накопителей легко, если такая возможность была заранее предусмотрена. Будем надеяться, что ваш процессор и материнская плата прослужат вам несколько лет.

требуемую вам спецификацию, чтобы не выбрасывать запчасти, обновляясь до своей реальной потребности. Кроме того, определитесь с качеством применяемых компонентов. В готовом ПК может быть заявлен жесткий диск 2 ТВ, однако это вряд ли вам о многом говорит. Большинство производителей жестких дисков производят несколько типов дисков подобного объема с разными скоростью, энергопотреблением и назначением. Вы узнаете намного больше о компонентах, выбирая их сами; сборка ПК — не такое уж трудное дело, но позволит вам чувствовать себя куда увереннее, когда дойдет до модернизации вашего компьютера. Вы можете облагораживать систему постепенно — имея компьютер, который вы хорошо изучили, надставить объем жесткого диска или памяти, а потом, через год, сменить процессор и материнскую плату на более быстрые.

> «До сих пор одни компоненты поддерживаются лучше других.»

Что вам понадобится?

Для сборки компьютера нужно несколько стандартных блоков. Как минимум, вам понадобится процессор, материнская плата, память, жесткий диск, видеокарта, корпус и блок питания. Кроме того, вероятно, потребуются монитор, клавиатура и мышь — это если вы собираете не мультимедиа-центр, который использует пульт дистанционного управления и подключается к телевизору, или сервер без клавиатуры и монитора, такой, как бокс NAS.

На следующих страницах мы более подробно рассмотрим каждое из этих устройств, объясняя, что лучше выбрать. Смысл не в том, чтобы назвать вам компоненты, которые надо брать, а в том, чтобы предоставить информацию, позволяющую принять решение, основываясь на том, что именно вам нужно. Когда вы соберете свой компьютер, вам надо будет установить на него операционную систему и программы. Естественно, мы предполагаем, что вы установите Linux, но, возможно, вам также нужно будет установить Windows для игр или для других программ, так что мы рассмотрим, как именно установить две ОС, чтобы они сосуществовали в гармонии друг с другом (ну, хотя бы относительной).

Выбор компонентов

При выборе компонентов вы должны учитывать, как вы будете использовать компьютер сейчас и впоследствии. Цель использования может сильно повлиять на выбор: например, игровая машина должна обладать быстрым диском, однако его объем не так важен для ОС и для игр, в которые вы играете в данный момент, так что идеальным выбором будет SSD. Для NAS, наоборот, важно пространство, но не скорость, так что более медленный жесткий диск будет лучше.

Пользователям Linux придется принимать во внимание еще один аспект: совместимость. Хотя сейчас ситуация намного лучше, чем когда-либо в прошлом, до сих пор одни компоненты поддерживаются лучше других. Давайте рассмотрим основные компоненты, которые вам потребуются, и имеющиеся опции. »

› Найдите оптимальное соотношение цены и качества у предпочитаемого вами производителя процессоров.

Совместимость Linux

Сейчас оборудование в Linux поддерживается очень хорошо. Большинство устройств имеют поддержку прямо в ядре. Незачем лазать по сайтам производителей в поисках драйверов. При покупке компонентов для своего компьютера надо знать, поддерживается ли каждый конкретный компонент. Первый шаг — определить, какой именно компонент используется, что на самом деле труднее, чем кажется. На сайте производителя материнских плат может быть указано, что на плате есть порт Gigabit Ethernet, но при этом не указано, какой чипсет, так что вам придется провести самостоятельное исследование.

Как правило, если ввести полный код продукта и слово Linux в вашу любимую поисковую машину, она выдаст вам достаточно информации. Стоит ограничить ваш поиск недавними постами: нет никакого смысла читать жалобы годовой давности на отсутствие поддержки выбранного вами оборудования, поскольку на данный момент оно уже вполне может поддерживаться. Ищите посты с подробной информацией, например, с указанием кода чипсета и наименованием модулей, и избегайте пустых многословных постов. Если вы зададите вопрос о поддержке определенного устройства на таких форумах, как наш (www.linuxformat.com/forums) или www.linuxquestions.org, вы получите сведения из первых рук; однако сначала поищите информацию на форуме. Возможно, этот вопрос уже задавали, и никому не нравятся вопросы от людей, которым лень искать самим.

Обычно проблемы с совместимостью возникают, когда производитель не сотрудничает с разработчиками драйвера ядра, и поэтому во всем нужно использовать обратное проектирование — обычно это касается беспроводных адаптеров, но и некоторых звуковых карт тоже. Другой причиной бывает новизна искомого оборудования, из-за чего его не успели добавить в ядро вашего любимого дистрибутива. И вам придется обратиться к более новому, более передовому релизу за поддержкой. Обычно эта ситуация в первую очередь возникает с сетевыми адаптерами, как проводными, так и беспроводными.

Процессор

Самое первое решение, которое нужно принять — какой взять процессор, поскольку это повлияет на выбор материнской платы и практически всего остального. Выбор может показаться очевидным — взять самый быстрый из тех, которые вы можете себе позволить; но на самом деле всё не так просто. Самые быстрые процессоры также и самые дорогие, причем разница огромна. Всегда можно найти оптимальную цену, и она обычно несколько ниже цены на самый навороченный процессор. Скорость для процессора — это не все: память тоже сильно влияет на производительность вашего компьютера. Как правило, пойдя на уступки в скорости процессора и потратив сэкономленные деньги на больший объем памяти или SSD, вы в итоге получите более быстрый компьютер. Попробуйте оставить работать системный монитор, например, *top*, пока вы используете свой компьютер, и посмотрите, часто ли ваш процессор работает на полную мощь. Если только вы не геймер и не проводите свободное время, компилируя ядра, вас сильно удивит, как редко потребление процессора становится ограничением.

Следующий выбор, который вам предстоит сделать — выбор между Intel или AMD. Можете поискать в Сети, что лучше, но будьте готовы к тому, что придется потратить некоторое время на поиск реальных фактов среди религиозного пыла. Правда в том, что процессоры сегодня настолько быстрые, что для домашней системы вам подойдет любой из них. От выбора процессора зависит выбор материнской платы, так что вполне может выйти, что, в зависимости от ваших потребностей и возможностей, вы выберете процессор, подходящий к нужной вам материнской плате.

› Выглядит скромно, однако оказывает куда большее воздействие на производительность вашего компьютера, чем многие другие компоненты.

Материнская плата

Если процессор — это сердце вашего компьютера, то материнская плата — его центральная нервная система. К материнской плате подключается всё. В прошлом в этом и заключалась работа материнской платы: всё соединять. Теперь туда встроены ранее разрозненные компоненты, например, сетевые интерфейсы и звуковые карты. Поскольку Intel и AMD используют разные разъемы для CPU, выбор процессора сразу сокращает количество материнских плат, из которых вы можете выбирать. Вот некоторые вопросы, которые вам следует задать: нужно ли мне более одного сетевого интерфейса? Сколько памяти мне надо будет установить — сейчас и в будущем? Сколько мне понадобится SATA-коннекторов — не забудьте принять во внимание по одному для оптического диска и, возможно, коннектор для внешнего eSATA. Кроме того, насколько быстрые интерфейсы SATA? Это особенно важно, если вы используете SSD. Сколько USB-портов мне понадобится и какого типа? USB 2.0 прекрасно подходит для клавиатуры и мыши, но вам все равно понадобятся коннекторы USB 3.0 для устройств хранения, и, вероятно, коннектор USB-C.

Примите во внимание встроенные устройства интересующей вас материнской платы, такие, как сетевые интерфейсы и звуковые «карты». Если сами материнские платы будут работать с Linux, будут ли работать периферийные устройства? В наши дни звуковые карты не представляют особой проблемы, так как большинство из них поддерживает стандарт Intel HD Audio; но проверить всё же стоит, поскольку могут различаться используемые кодеки. Проводные сетевые устройства тоже в основном поддерживаются неплохо, но у вас могут быть проблемы с очень новой материнской платой, которая использует устройство, пока не поддерживаемое ядром вашего дистрибутива. Как обычно, быстрый поиск в вашем любимом поисковике по имени материнской платы и дистрибутива Linux должен выявить как потенциальные проблемы, так и решения. Поддержка встроенных устройств может и не создать проблемы, если у вас достаточно слотов PCI, чтобы установить альтернативные устройства и потом отключить встроенные.

Размер тоже бывает важен. Если вы собираете компьютер в большом корпусе, вам будет проще работать с полноразмерной материнской платой ATX, но для мультимедиа-центра вам может понадобиться не столь крупный форм-фактор, который поместится в симпатичный корпус, способный приютиться под телевизором.

› Говоря о сборке ПК, материнская плата — это та штука, к которой всё присоединяется.

Память

Память — один из самых простых способов разогнать производительность современного компьютера. Она позволяет обрабатывать больше задач и данных одновременно, и любая незадействованная память используется ядром Linux для кэширования данных диска, вследствие чего ваш компьютер и показывает, что у него практически нет свободной памяти спустя некоторое время после начала работы. При покупке памяти надо обратить внимание на три вещи: физическую раскладку, размер и скорость. Современный стандарт

Терминология

Подыскивая оборудование, вы будете встречаться с большим количеством сокращений и жаргонными словами, не всегда понятными новичку.

Вот некоторые из наиболее часто встречающихся терминов.

» **DDR** 2/3/4 Double Data Rate [Двойная скорость данных], технология, используемая в чипах памяти.

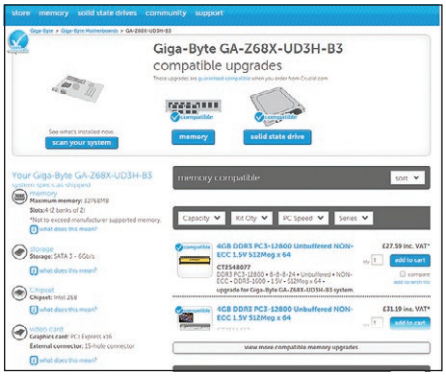
» **DIMM** Dual Inline Memory Module [Двойной встроенный модуль], это подключаемый модуль памяти.

» **WAF** Wife Acceptance Factor [Фактор приемлемости для жены], встречается на чисто мужских форумах и считается важным для домашнего оборудования.

» **Northbridge/Southbridge** [Северный мост/Южный мост] Чипы на материнской плате, которые работают с основной логикой и коммуникацией между CPU и другими компонентами.

» **SATA** Serial ATA, текущий стандарт для соединения устройств хранения с компьютерами.

» **ATX** Спецификация конфигурации материнской платы, означающая, что большинство плат взаимозаменяемы в физическом формате и со стандартными коннекторами.



› Некоторые производители памяти предоставляют базы данных, очень удобные при выборе оптимальной памяти для вашей материнской платы.

памяти — DDR3 и DDR4. Не пытайтесь вставить в эти слоты более старые планки памяти DDR или DDR2: они несовместимы. С объемом всё очевидно: приобретайте самый большой, какой позволяет ваш бюджет. Загляните в руководство вашей материнской платы, чтобы узнать максимальный размер для каждого слота, потому что, хотя спецификации разрешают до 16 ГБ на DIMM, большинство систем Intel имеют ограничения до 8 ГБ на единицу. Обычно материнские платы имеют двухканальную архитектуру для памяти, так что вставляйте планки парами и посмотрите в руководстве, какие слоты идут вместе. В этом нет нужды, если вы заполняете все четыре слота, но, если только вы не используете планку максимального размера, лучше поставить две более крупные планки. Если вы хотите получить 16 ГБ памяти, то две планки по 8 ГБ позволят вам увеличить объем памяти впоследствии, добавив еще две, а вот если использовать четыре планки по 4 ГБ, такой возможности у вас не будет.

Накопители

Когда-то единственный вопрос, задаваемый о жестком диске, был «какого объема?» Ныне мы еще добавляем: сколько? И какого типа? При выборе типа нам обычно приходится выбирать между традиционным жестким диском с вращающимися дисками и твердотельными жесткими дисками (SSD). (Хотя есть и гибридные диски, которые используют вращающиеся диски и SSD в качестве кэша.) SSD намного быстрее жестких дисков, однако их стоимость за гигабайт намного выше: 256-ГБ SSD стоит примерно столько же, сколько 3-ТБ диск. Итак, SSD предлагают намного более быструю загрузку компьютера и программ, однако на жесткий диск вмещается больше. Кроме того, SSD использует меньше энергии и более надежен, что делает его очевидным выбором для ноутбука. А с настольным ПК вы можете позволить себе роскошь использовать более одного диска, так что оптимальным будет SSD для операционной системы и программ плюс жесткий диск для данных. Вы можете поместить операционную систему на SSD и **/home** на жесткий диск, или оставить **/home** на SSD, и это означает, что у вас будет дополнительная скорость для таких вещей, как браузер и кэш электронной почты. Затем вы можете подмонтировать свои данные, например, свою коллекцию мультимедиа, на жесткий диск. В любом случае, вам не понадобится исключительно большой SSD, 128 ГБ будет более чем достаточно, а значит, вы уж точно выберете хороший. SSD сильно различаются по производительности, так что убедитесь, что вы выбрали тот, у которого достойная скорость чтения и записи; в районе 500 Мб/с будет весьма неплохо. Настройка SSD практически та же, как и у жесткого диска.

› **Для ноутбуков SSD — выбор очевидный, но SSD идеальны и для настольных ПК в качестве носителя ОС и используемых вами программ.**

Еще одна достойная рассмотрения возможность — использование нескольких жестких дисков в массиве RAID. Это позволит вам создать резерв: если один диск в массиве откажет, ваши данные будут находиться на другом. Регулярного резервного копирования это не заменит, но спасет вас в случае отказа диска. В RAID 1 самая простая конфигурация: два диска зеркально отражаются. Все данные записываются на оба диска, но считываются с одного (что дает выигрыш в скорости чтения, поскольку данные идут с того диска, который первым их найдет). Большинство программ установки дистрибутивов справятся с установкой на массив RAID, но в RAID 1 вы также можете сделать установку на один диск и добавить второй для создания массива. RAID обрабатывается ядром Linux; не включайте никаких настроек RAID на материнской плате. Это так называемые fakeRAID [ложные RAID], и они требуют работы драйверов Windows. Просто дайте вашей программе установки увидеть, что у вас два диска, и ядро позаботится о них.

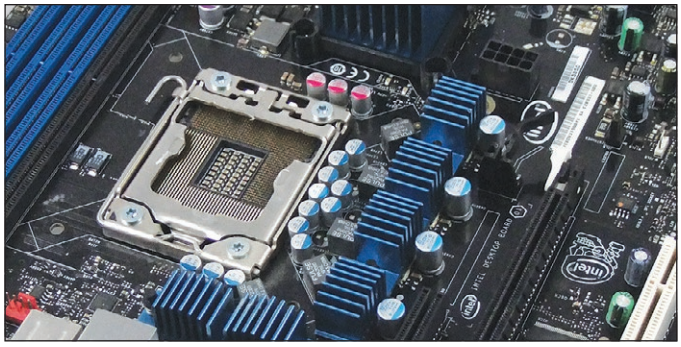
› **Графические карты по сути сами являются компьютерами, но для многих систем вы, вероятно, обойдетесь встроенной видеокартой.**

Видеокарта

В наши дни большинство материнских плат предлагают весьма разумную встроенную графику, так что отдельная карта вам, возможно, и не понадобится. Если она вам нужна, то выбирать придется между Nvidia и AMD, и это еще одна тема, способная привести к бурным конфликтам.

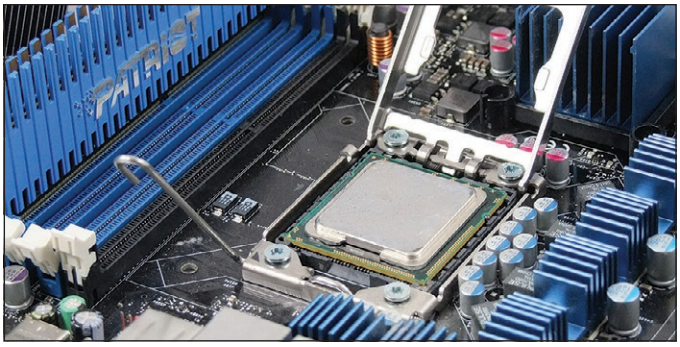
Видеокарты Nvidia позволят загрузить компьютер в графический режим, но выбор встроенных в ядро драйверов nv ограничен. У вас есть парочка опций. Новые драйверы с открытым кодом nouveau сейчас работают очень неплохо — мы сами их используем — и обеспечивают достойную »

Сводим все сказанное воедино



1 Раскладка материнской платы

Найдите в руководстве материнской платы схему ее раскладки и коннекторов, да не закрывайте эту страницу. Вы должны заглянуть в нее, прежде чем делать любое соединение на плате. Многие коннекторы — просто контактные штырьки, из-за чего можно употребить их неправильно. Законы Мэрфи здесь явно работают. RTFM применимо к оборудованию не в меньшей степени, чем к программам.



2 Установка без усилий

Отведите прижимной рычажок на сокете CPU и вставьте CPU, выровняв угловую точку на чипе с отметкой на сокете. Он должен войти без усилий, и затем закрепите его рычажком. Удалите пленку с проводящей подложки радиатора и установите его над CPU, следуя инструкциям для CPU и материнской платы. Соедините кабель вентилятора с материнской платой.

производительность. Если вам нужна оптимальная производительность вашей видеокарты, лучше установить собственные драйверы Nvidia. Поскольку они имеют закрытый код, они часто не включены по умолчанию, так что вам придется включить опцию для ограниченных или сторонних драйверов в вашем дистрибутиве. Можно также скачать и установить драйверы непосредственно с www.nvidia.com, но этого делать не рекомендуется, поскольку тогда ваш менеджер пакетов не сможет отслеживать и обновлять их. Ситуация с видеокартами AMD такая же: потенциальная морока с запуском двоичного драйвера или драйверов с открытым кодом и низкой производительностью. Выбор между AMD и Nvidia — почти такой же религиозный вопрос, как выбор между CPU Intel и AMD. Если вам не нужна высококачественная 3D-производительность, то подойдет любая, а если нужна, то вам в любом случае придется использовать двоичный драйвер, какую бы из них вы ни выбрали. На данный момент проприетарные драйверы Nvidia считаются лучше и надежнее, но это может измениться.

Корпус

Корпус компьютера часто оставляют напоследок, но большой и качественный корпус может значительно облегчить жизнь. Дешевый корпус усложнит прокладку проводов, и в нем часто имеются острые края, а это означает порезанные пальцы и капли крови на вашей новенькой материнской плате (вероятно, вы почуяли, что за этим утверждением кроется личный опыт).

Кроме того, большой корпус также обеспечивает лучшую циркуляцию воздуха для лучшего охлаждения. Если вы собираете мультимедиа-центр для гостиной, то немаловажным фактором является высокий WAF (см. врезку *Терминология* на стр. 36), но обращайте внимание на уровень шума. То, что изначально кажется не громче шепота, при просмотре телевизора может сильно раздражать. Если вы решили выбрать корпус с окошком сбоку, помните, что тогда придется быть особенно внимательными при прокладке проводов. Спутывать провода в рыболовную сеть и вообще не лучшая идея, а особенно когда они выставлены на всеобщее обозрение.

Блок питания

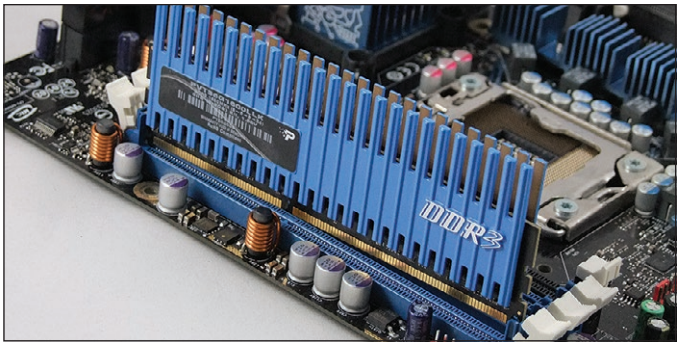
Один из самых недооцененных компонентов — это блок питания (БП — PSU). Избегайте дешевых БП, поставляемых вместе с корпусом (на самом деле, избегайте настолько дешевых корпусов, что они идут вместе с БП). БП должен быть качественным и надежным. Это может показаться тривиальным утверждением, применимым к любому оборудованию, однако сломавшийся БП способен причинить ущерб другим компонентам. Заплатить £100 за материнскую плату или жесткий диск, или за то и другое, а потом позволить им сгореть из-за БП стоимостью в £15 — не самый лучший вариант экономии.

Озвучим ряд других аспектов, которые стоит учесть при выборе БП: он должен обеспечивать достаточную мощность для ваших нужд и сейчас, и в будущем; он должен быть эффективным (любой стоящий БП имеет рейтинг 80 Plus, но лучше искать модели Gold, Platinum и Titanium); он должен быть тихим (особенно это важно для компьютера в гостиной). Есть несколько сайтов, где, перечислив используемые вами компоненты, вы получите рекомендации по нужному вам питанию, например, http://support.asus.com/PowerSupply.aspx?SLanguage=en. Убедитесь, что выбранный вами БП имеет все нужные коннекторы; большинство материнских плат принимают коннектор 24+4-pin ATX и отдельный шнур питания для CPU. Проверьте также наличие достаточного количества нужного типа проводов питания для дисков: некоторые БП до сих пор идут с со старыми коннекторами Molex, а не с новыми SATA. Если вы используете большой корпус с БП, установленным внизу, проверьте, достаточна ли длина проводов, чтобы дотянуться до дисков вверху, хотя можно достаточно недорого приобрести удлинители SATA, чтобы решить две последних проблемы. Вам будет приятно узнать, что насчет совместимости с Linux никаких проблем нет!

Спецсистемы

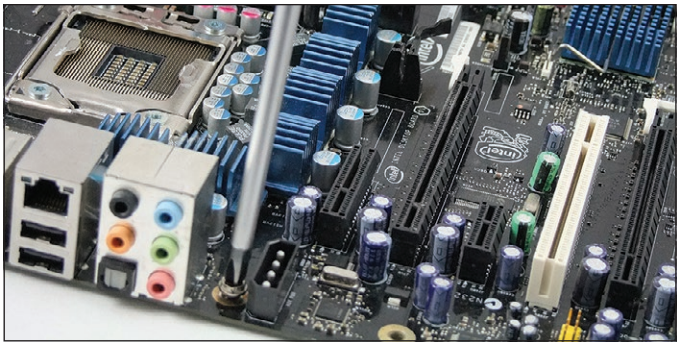
Пока мы рассматривали настольные системы общего назначения, однако нельзя забывать и о специальных областях использования, таких, как домашний сервер или высокопроизводительная игровая система. Требования к домашнему серверу будут в корне отличаться от требований к обычному настольному ПК или игровой системе.

Поскольку данные передаются через сеть, быстрый SSD совершенно не нужен. Вы можете использовать его для ОС, но серверы редко перезагружаются, так что вы ничего не выиграете от предлагаемой ими высокой скорости загрузки; да и число работающих на них программ ограничено, так что время запуска опять-таки неважно. Обычно серверы работают без клавиатуры, мыши и монитора, так что крутая видеокарта тоже



3 Слоты памяти

Установите планки памяти DIMM в разъемы на материнской плате. Убедитесь, что они вошли правильно; слоты для DIMM должны соответствовать выступам в разъеме, и они входят только в одном направлении. Если вы не используете все слоты, загляните в руководство своей материнской платы, чтобы задействовать оптимальные слоты и обеспечить все возможности двойного канала.



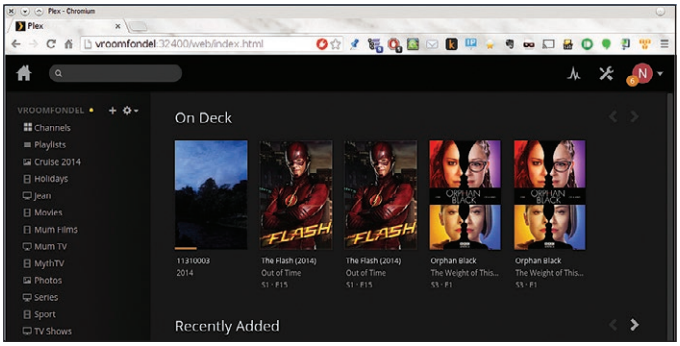
4 Кронштейн материнской платы

Совместите плату ввода-вывода материнской платы с отверстием в задней стенке корпуса и затем поместите материнскую плату в корпус. Она должна стоять на небольших опорах, которые, возможно, перед установкой платы вам сначала придется прикрутить к корпусу. После этого зафиксируйте плату на опорах с помощью прилагающихся шурупов — но не закручивайте их слишком сильно.

незачем; монитор понадобится только в процессе установки, а для этого вполне хватит встроенной графики на большинстве материнских плат. А вот действительно важны достойное количество памяти и достаточный объем хранилища. Если вы собираете файловый или мультимедиа-сервер, выберите материнскую плату с достаточным количеством портов SATA и корпус с таким же количеством отсеков для дисков. Но независимо от количества ваших накопителей в определенный момент вам понадобится еще больше. Если вы собираете сетевой или почтовый сервер, то объем хранилища не будет основной проблемой, разве что вы решите обслуживать медиа-файлы через интернет-соединение, однако большая память явно не повредит, особенно с *Apache*.

Еще один тип домашнего сервера — мультимедиа-сервер. Это может быть просто репозиторий для ваших видеофайлов, а может, вы будете делать записи на сервере с помощью серверного ПО *MythTV* или чего-нибудь вроде Tvheadend. В любом случае, вам потребуется много места. Если ваш сервер умеет также перекодировать видео, например, как *Plex*, который переформатирует видео для соответствия устройству, которое его воспроизводит, и скорости его соединения с сервером. Вам понадобится достойный объем ОЗУ и довольно быстрый CPU — не игровой монстр, конечно, но не нечто малобюджетное.

Поскольку всё работает через сеть, надо ли говорить, что у вас должен быть быстрый сетевой интерфейс, особенно если его будут использовать более одного клиента за раз, так что убедитесь, что у вас гигабитный интерфейс или разоритесь на достойную сетевую карту PCI-e.

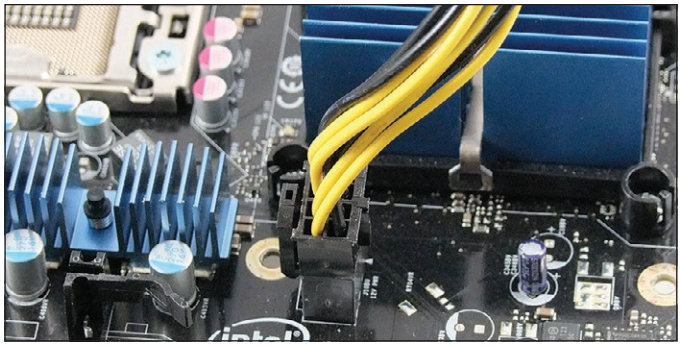


› Если вы собираете сервер, подобный этому мультимедиа-серверу, ваши запросы будут иными. Вам нужно достаточно места на диске и быстрое сетевое соединение (или два). Возможно, вам даже понадобятся лошадиные силы CPU для перекодирования мультимедиа.

Высокопроизводительные игры

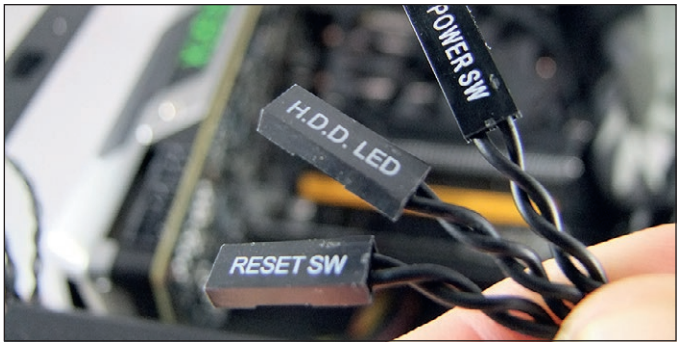
Игры — причем мы имеем в виду игры с достойной графикой, а не азартные игры — предъявляют особые требования к компьютеру. Производительность является очевидным фактором, но это не просто производительность CPU. В некоторых играх GPU на видеокарте играет куда более важную роль, чем CPU. Естественно, вам нужен быстрый CPU и видеокарта. Многие из высокопроизводительных видеокарт занимают место двух слотов PCI-e, поэтому вы должны проверить, достаточно ли место у вас в корпусе и на материнской плате, и что ваш БП вам подходит, т. е. сможет обеспечить достаточно энергии и имеет достаточно коннекторов. Память тоже важна: у вас ее должно быть достаточно, и она должна быть как можно быстрее. Место на диске не так важно, как скорость, поэтому в качестве хранилища предпочтительнее будет SSD. Игроманы постоянно жаждут производительности, и один из способов ее обеспечения — разгон системы, это когда вы разгоняете свой процессор и память до скорости, которая превышает штатную. При обычном применении мнения насчет разгона неоднозначны; он обеспечивает более высокую производительность, но ценой вероятного уменьшения срока жизни компонента и определенной степени ненадежности, если вы зайдете слишком далеко, плюс увеличение перегрева, с которым надо как-то бороться. Некоторые CPU на определенной скорости блокируются, и следует убедиться, что выбранный вами процессор допускает разгон. Кроме того, понадобится соответствующая материнская плата. Все платы предлагают определенный контроль за частотой и напряжением, »

> «Многие из высокопроизводительных видеокарт занимают два слота PCI-e.»



5 Коннекторы

Установите блок питания и присоедините кабели питания и CPU к материнской плате. Пора присоединить и все кабели вентиляторов. Добавьте жесткие диски или SSD и оптические диски. Они либо вкручиваются в корпус, либо, в некоторых случаях, имеют соединения без шурупов. Соедините кабели SATA от дисков с материнской платой. Соедините их шнуры питания с БП.

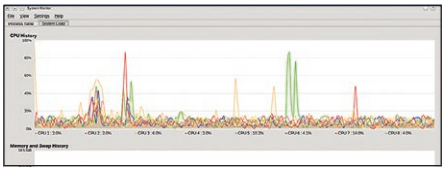


6 Светодиодные индикаторы корпуса

Присоедините провода от переключателей корпуса и светодиодных индикаторов к материнской плате. Здесь довольно хлопотные коннекторы, но надо присоединить их правильно; руководство от вашей платы подскажет вам, что с чем соединять. Некоторые провода идут с блочным коннектором, облегчая вам работу. Пора также соединить USB-порты на корпусе с соответствующими входами платы.

но некоторые предусматривают разгон и предлагают намного бóльшую степень контроля — это стоит хорошенько уточнить.

Чтобы справиться с перегревом, необходим корпус большого размера, обеспечивающий достаточный приток воздуха, в сочетании с достойным кулером CPU (обычный кулер, поставляемый с процессором, для разгона не предназначен), и достаточно вентиляторов. На материнской плате, которая предусматривает разгон, может быть больше коннекторов для вентиляторов и возможности управления их скоростью. При выборе вентиляторов для охлаждения берите самый большой, какой только влезет. Более крупные вентиляторы обеспечивают больший приток воздуха при меньшей скорости вращения, а быстро вращающиеся вентиляторы создают больше шума.



› Даже в этой системе, которой уже четыре года, CPU в основном простаивает, но ведь всегда приятно иметь запас.

Если шум для вас проблема или вы хотите добиться разгона по максимуму, рассмотрите возможность водяного охлаждения. Когда-то разгон был весьма специфичной областью, которая требовала покупать и собирать разные компоненты, потом тратить кучу времени на прокачку системы и выполнение тестов на герметичность (вы удивитесь, но вода не очень хорошо ладит с электроникой). Сейчас можно найти готовые системы, где вы прикручиваете к отведенному месту в корпусе готовый радиатор, а вместо вентилятора CPU помещаете теплоотвод. Некоторые комплекты включают также кулеры видеокарт, потому что игры их тоже сильно нагружают. Водяное охлаждение хорошо не только для геймеров и фанатов скорости: его можно использовать на обычном настольном ПК для снижения шума. При подобной сборке самый громкий шум издает двигатель жесткого диска.

Замкнутые системы

Сборка настольного ПК или сервера — довольно простая задача, а вот создание собственного ноутбука — отнюдь нет. Купить ноутбук по частям можно, но обычно так не делается. Однако большая часть того, о чем мы говорили в нашей статье, применима и здесь. Просто нужно выбрать подходящие компоненты для ноутбука, и это даже важнее, потому что их нельзя просто вытащить, если вам что-то не понравится. Так что следует выбрать предварительно собранный ноутбук себе по бюджету, отвечающий вашим потребностям, а потом

› Сборка собственного ноутбука — не самое разумное предложение, однако не помешает исследовать его компоненты и убедиться, что они отвечают вашим потребностям.

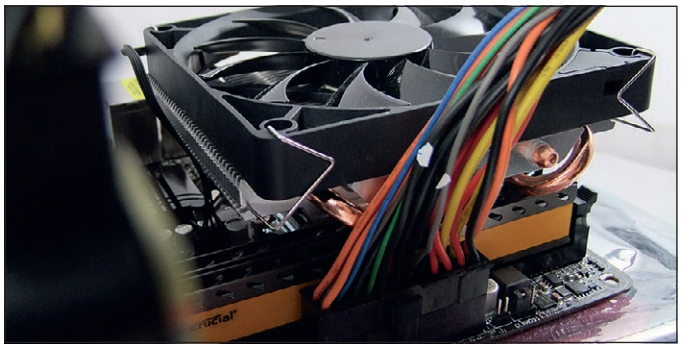
Установка операционной системы

Когда вы соберете свой компьютер, вам надо будет установить на него операционную систему. Здесь мы не будем рассматривать этот процесс — программы установки в дистрибутивах достаточно хорошо документированы; но на несколько моментов стоит обратить внимание. Новые материнские платы будут использовать UEFI вместо BIOS. Возможно, на них также включена Secure Boot. Вам почти наверняка придется ее отключить в прошивке. Нажмите на соответствующую клавишу при загрузке, чтобы настроить прошивку — в руководстве материнской платы должно быть написано, какую именно (кстати, одна из прелестей сборки собственного компьютера заключается в наличии руководства материнской платы).

У вас должно получиться использовать UEFI для загрузки компьютера, большинство дистрибутивов уже работают с ним, и с ним приятнее работать, чем со старой системой MBR. Если вам нужна загрузку MBR, для этого должна быть опция в меню прошивки. В некоторых имеется опция загрузки UEFI, которую можно отключить, в некоторых есть опция с загадочным названием CSM (Compatibility Support Module), которую надо включить для загрузки MBR. Если вы создаете систему только с Linux, можно просто загрузиться с любимого DVD и произвести установку. Если вам нужна двойная загрузка с Windows, сначала установите Windows, а потом Linux. Программа установки Linux найдет Windows и настроит меню двойной загрузки, а если вы решите сделать всё наоборот, программа установки Windows просто проигнорирует загрузчик Linux.

7 Видеокарта

Установите свою видеокарту в слот PCI-16. Загляните в ее документацию, чтобы выяснить, достаточно ли ей питания от слота PCI или нужно внешнее питание. Одновременно установите остальные карты PCI, если вы их используете. Многие новые блоки питания (БП) предлагают модульную систему кабелей; кроме того, имеются адаптеры на тот случай, если у вашего БП отсутствуют соответствующие коннекторы, однако он должен быть достаточно мощным.



8 Аккуратные провода = ясный ум

Размещайте провода аккуратно, даже если в вашем корпусе нет окна. Кое-как размещенные провода затруднят циркуляцию воздуха и сильно осложнят вам жизнь, когда вы решите добавить или заменить какой-то компонент. Если в вашем корпусе нет зажимов, чтобы удерживать провода, можно подвязать их проводами потоньше. Кроме того, можно завести провода за лоток материнской платы или внутрь крепежа жесткого диска.

проверить, всё ли вам подходит. То же можно сказать и о неттопе, который вы хотите использовать в качестве медиа-плейера.

Те же правила относятся к выбору подходящего процессора, видеокарты, звуковой карты и т.д. (хотя слово «карта» здесь не вполне правильно: почти все это размещается на материнской плате). Есть ряд важных отличий: жесткие диски ноутбука обычно медленнее, чем жесткие диски настольного ПК, и выигрыш в скорости SSD еще заметнее. Опции обновления памяти ограничены, поэтому выбирайте устройство с достаточным объемом памяти и убедитесь, что в нем есть как минимум один свободный слот для расширения. В некоторых ноутбуках всего один слот, поэтому обновление памяти означает, что вам придется выбросить то, что у вас было, а не добавить что-то новое.

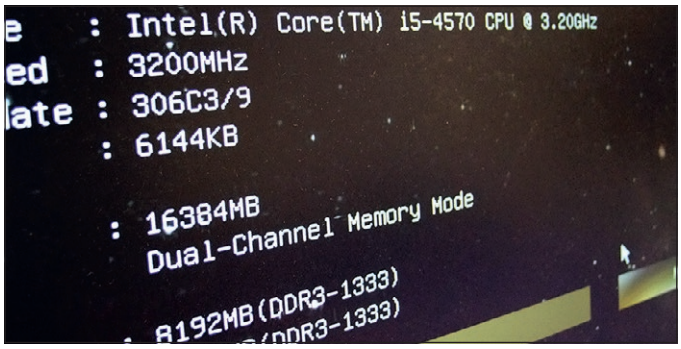
Постарайтесь загрузить ноутбук с новейшего live CD, например, с **LXFDVD**. Попробуйте System Rescue CD с опцией Alt kernel; вы получите самое свежее ядро с самой современной поддержкой оборудования. Запустите **$ ifconfig -a**, чтобы увидеть проводные и беспроводные интерфейсы, и **$ aplay** для проверки работы звуковой системы, от файлов до колонок. Беспроводные интерфейсы у ноутбуков иногда проблемны, но в большинстве есть слот mini-PCI, и тогда можно вставить другую карту.

На фоне войн между Intel и AMD в области CPU и между Nvidia и AMD в области видеокарт, Intel предлагает отличную поддержку своих компонентов. Если в вашем ноутбуке есть видеокарта и беспроводная карта Intel, бед с ними не будет: всё должно просто работать, без всякой установки дополнительных драйверов. А теперь ставьте любимый дистрибутив и наслаждайтесь скоростью и свободой Linux на новом оборудовании, выбранном лично вами: право же, это чудесно! LXF

А теперь — осторожно!

Собрать ПК несложно, но при работе с электронными компонентами требуется определенная осторожность. Подготовьте достаточно просторное рабочее место, свободное от всего ненужного и с непроводящей поверхностью. Статическое электричество способно испортить электронные компоненты, и оно может накапливаться на вашем теле, о чем вы можете даже не подозревать, пока не возьметесь за компонент и не убьете его. Этого можно избежать, если вы заземлитесь, чтобы избавиться от статического заряда. Простейший способ это сделать — коснуться заземленного объекта, например, батареи центрального отопления или кухонной раковины перед тем, как прикасаться к компоненту. Вы также можете купить антистатический браслет, соединенный гибким проводом с точкой заземления, чтобы на вас не скапливалось статическое электричество во время сборки.

У компьютера, который вы собираете, нет гарантии, но она есть у его компонентов. И если они не были повреждены в процессе сборки, любой нормальный производитель их заменит в случае необходимости.



8 Тили-тили тесты

Присоедините монитор и клавиатуру и включите компьютер. Удерживайте соответствующую клавишу, чтобы загрузить меню BIOS/прошивки, и перейдите на страницу состояния системы, чтобы убедиться, что ваш процессор не перегревается. Только после этого можете проверять, распознаются ли ваша память и диски. У некоторых материнских плат придется изменить настройки, чтобы увидеть всю память, и тем не менее ваш новый ПК собран!

Сохраняя мировую историю

Могут ли один человек, маленькая команда и открытый робот сохранить визуальную историю мира? Мэтью Хэнсон узнает у Тома Смита.

Том Смит [Tom Smith] называет себя изобретателем и предпринимателем, создателем Project Gado — открытого робота-сканера, способного оцифровать большие коллекции исторических артефактов. Последние зачастую являются не только бесценными, но и очень хрупкими. Наш разговор — о том, как появился проект Gado и как идеология Open Source помогает проекту в оцифровке визуальной истории мира.

Linux Format: **Вы называете себя изобретателем и предпринимателем; как вы к этому пришли? С чего все началось?**

Том Смит: Пожалуй, я всегда к этому стремился. Я с детства норовил разбирать всякие вещи. Покупал на дворовых распродажах разные устройства и разбирал. Смотрел, как работает. Лет в пять-шесть я уже собирал компьютеры и прочее, и электроинструмент у мня появился гораздо раньше, чем следует! Так что я всю жизнь этим увлекаюсь. А появление проекта Gado обусловлено конкретными причинами. Когда сталкиваешься с какой-то проблемой, действительно насущной для данной сферы, и сначала просто задумываешься: «Можно ли как-то решить ее посредством технологии, посредством инноваций». А потом это развивается в нечто большее.

LXF: **Расскажите нашим читателям о своем проекте.**

TC: Начнем с того, что в фотоархивах, музеях, газетах порой хранятся сотни, тысячи, даже миллионы исторических снимков более чем вековой давности, и все это — в аналоговом формате. И у многих подобных организаций нет ресурсов, чтобы оцифровать эти коллекции.

Так что миллионы фотографий просто лежат там, и по сути никто не имеет к ним доступа. Эти данные должны быть в электронном формате, чтобы люди могли их найти и изучить, и чтобы для самих организаций это стало регулярным источником дохода. Но у них нет средств, чтобы нанять кого-то сделать это вручную. А многие существующие технологии машинной оцифровки, такие как сканер с автоматической подачей бумаги, не подходят для работы с фотографией 150-летней давности. Положить ее туда — все равно что положить миллион, и одно зажатие бумаги может уничтожить бесценный фрагмент истории! Так что при отсутствии достойного технического решения остается только нанять людей, которые бы сидели там и целый день их сканировали, а если денег на это нет, история, которую вы храните, будет просто утрачена. Ведь для тех, кто не может посетить ваш архив и просмотреть аналоговые каталоги, она недоступна.

И я столкнулся с этим на практике, в газете *Afro-American Newspaper* в Балтиморе. Это одно из старейших в своем роде ныне действующих изданий в мире, в США-то уж точно. Они существуют с 1892 года, и их фотогалерея с того времени насчитывает полмиллиона исторических снимков. Ими практически забито все помещение их главного офиса в Балтиморе. А оцифровать за всю историю газеты им удалось только 5000.

На самом деле, у меня тогда был проект по устной истории. Я по образованию антрополог. Работал на базе Университета Джона Хопкинса в Балтиморе. И я хотел найти фотографии, чтобы

проиллюстрировать рассказы местных жителей, которых я опрашивал. Цифровых нигде не нашлось. Я думал, вся визуальная история этого района утрачена. И тут, увязавшись за другим исследователем в Afro American Newspaper, я натыкаюсь на этот полумиллионный архив. Там были тысячи снимков местных жителей, и я понял, что это уникальные свидетельства важнейших исторических событий, но беда в том, что в аналоговой форме мир о них никогда не узнает, а средств на оцифровку нет, и я подумал, что если с помощью технологии создать нечто способное автономно перевести эти хрупкие исторические фото в электронный формат, тогда о подобных архивах узнают все, и для других исследователей местной культуры ее визуальная история будет полностью доступна. Так в 2010 г. и появился первый робот Gado.

LXF: Как раз насчет робота: при работе с такими невосполнимыми артефактами он должен быть очень осторожным. В чем преимущества использования Gado при оцифровке архивов?

TC: Думаю, самый большой плюс в том, что он освобождает сотрудников архива, позволяя им заниматься куда более интересными вещами, требующими людского вмешательства. Эта технология изначально создавалась для того, чтобы упростить людям жизнь, делать эту работу за них. Чем больше мы применяли эту технологию в работе с архивами, погружаясь в эту среду, тем больше мы понимали, что самое большое ее преимущество — то, что ее можно просто настроить, запустить, и она будет механически перекладывать фотографии и сканировать их.

Она не заменяет архивариуса: просто она работает сама по себе, а вы можете изучать происхождение снимка, заботиться о его сохранности, заниматься исследованием с посетителями архива. Так что я считаю, главный плюс в том, что она берет на себя эту трудоемкую и не особо интересную работу, а архивариусы занимаются чем-то более увлекательным. Ведь за этим они и пришли в профессию.

LXF: И какая доля архива *Afro American Newspaper* была оцифрована за время существования проекта Gado?

TC: Мы обработали 120 000 снимков. Так что еще много осталось, мы все еще туда приезжаем, находим папки, где вроде ничего не должно быть, а там большая часть коллекции; мы выбираем то, что представляет наибольшую историческую ценность. Совсем недавно мы обнаружили еще 30 фотографий доктора Кинга, Мартина Лютера Кинга-младшего [Martin Luther King Jr.]. Нашли целую папку снимков Элеоноры Рузвельт [Eleanor Roosevelt] 1940-х годов. Так что да, около 120 000, и это не конец.

LXF: Создавая Gado 2, вы использовали Arduino и Python. Почему именно они?

TC: Мы всегда хотели, чтобы у этого устройства была, так сказать, своя история, цель создания. Своя миссия. Это устройство создавалось, чтобы техническим путем решить проблему, и первым моим роботом был Gado 1.

Он занимал целый стол, был сделан из ДВП и потому весил фунтов 200. В архив его заносили втроем. И на его создание ушло около 3 месяцев. Второй такой громадины было не создать. Это действительно был пробный вариант, мы отсканировали им 1000 снимков просто чтобы доказать, что это возможно. Поскольку вначале многие говорили, что это «невозможно без больших затрат». Так что приняв решение развивать проект, при создании Gado 2, мы доказали, что за $500 можно сделать робот, который будет сканировать бесценные исторические фотографии, не повреждая их, и мы всегда хотели, чтобы это был открытый код. Поэтому мы проектировали устройство так, чтобы любой мог взять эти чертежи и начать работать. Так что Arduino и Python были выбраны не только потому, что это отличная платформа и отличный язык, но и ради прекрасного сообщества разработчиков Open Source.

По той же причине, все изначально писалось для Linux. Легко найти желающих включиться в проект Linux, легко сформировать команду; к тому же, имея дело с Python, Arduino или Linux, получаешь массу готовых наработок, и не надо думать: «Так, как же контролировать сканер через драйверы TWAIN?». Не надо беспокоиться «как же управлять этой „зеркалкой“?», поскольку робот делает снимки оборотной стороны каждой фотографии, и компьютер должен это контролировать. В Windows это кошмар: надо получить пакет разработки программ от Canon, от Nikon, потом объединять их в своем коде. А в Linux есть *gphoto2*, и это одна строка. Начать проще простого. Так что мы хотим как можно скорее построить вокруг проекта сообщество.

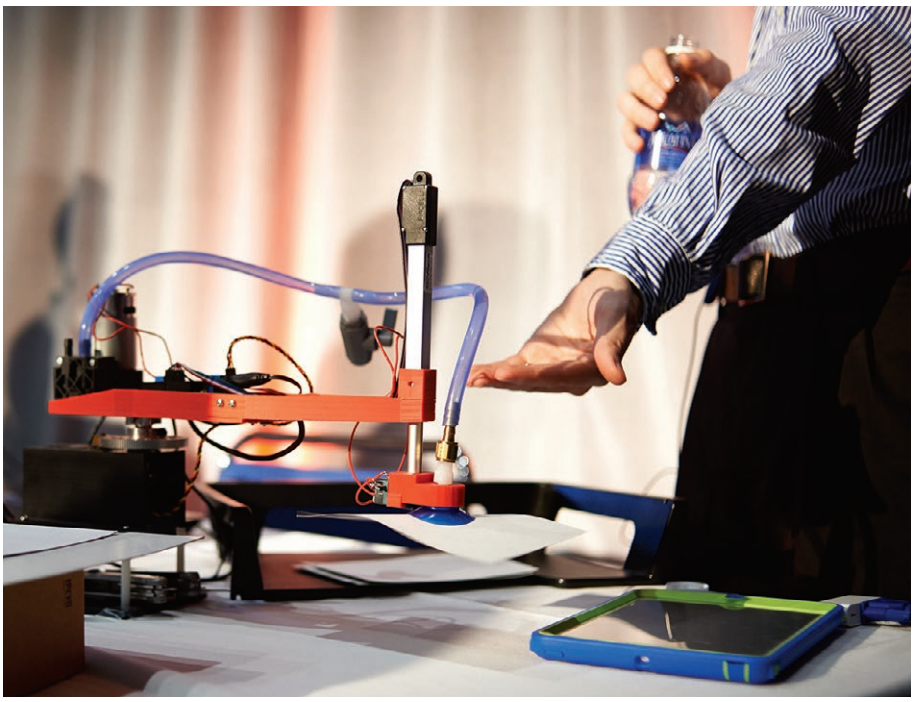
РОБОТ GADO

«За всю историю газеты им удалось оцифровать только 5000 снимков.»

LXF: И вы выкладываете весь код на GitHub. Хотите, чтобы люди создавали своих роботов на базе Gado?

TC: Конечно. Все, что касается робота, все программы для управления являются открытыми. Не только код, но и прошивка Arduino, вплоть до файлов для 3D-печати, чтобы создать части самой конструкции. Это тоже Open Source. Как и список материалов. Мы даже собрали несколько комплектов и продали по себестоимости первым пользователям, и продолжаем это делать, когда к нам поступают такие просьбы. Желающим создать свою группу пользователей мы можем предоставить все необходимое. Так что мы действительно хотим сделать все открытым, продвигать эти идеи, обращаться к группам пользователей и говорить: «Все это совершенно бесплатно, скачивайте», а также «вот требуемые вам составные части, собирайте, обращайтесь к нам. Мы приедем, поможем, объясним». Так что да, я думаю, проект изначально мыслился как открытый, и именно это помогло так быстро улучшить технологию и из неуклюжего бегемота, занимавшего целый стол, сделать робот вестом в 6 фунтов, способный за пару лет и $500 отсканировать 120 000 снимков.

LXF: То есть открытость подстегнула развитие Gado? »



TC: Несомненно. Об этом, о сообществе в том числе, я говорю на своих презентациях — на построение сообщества ушло немало времени — и я подробно об этом рассказываю. Как это происходило. Но как только сообщество сформировалось, самое крутое в Open Source то, что оно начинает поддерживаться и укрепляться само по себе.

Например, я общался с группой наших пользователей в Финляндии, это удивительная история. Мы объявили, что будем делать эти наборы. Решили как-то осветить это в прессе, и об этом начали говорить. Потом нам приходит e-mail: «Мы из университета Аалто в Хельсинки и хотели бы получить ваш набор». Мы продумали, как доставить робот в Финляндию, провезти через таможню, и отправили им. Через три месяца получаем сообщение: «Мы собрали Ваш робот, у нас есть вопросы». Выяснилось, что с ним работает целый коллектив исследователей. И вот они и один их практикант связывались с нами по *Skype*, из-за 11-часовой разницы во времени мне приходилось вставать в 5 утра и помогать им решать проблемы, но нам удалось наладить им работу.

Месяца четыре от них ничего не было слышно, а затем получаем сообщение: «Мы полностью переписали ваш код Linux, не могли бы вы дать нам доступ на GitHub, чтобы его выложить?» Разумеется, мы это сделали, и выяснилось, что они взяли этот кое-как сляпанный код и переписали его в модульной структуре, в соответствии со всеми стандартами, и выложили совершенно бесплатно. А потом, где-то черед полгода, мы получили от них еще одно сообщение: «Мы хотим организовать конкурс по промышленному дизайну среди финских студентов-дизайнеров на усовершенствование физической стороны робота, и сделать его портативным, чтобы работать на выезде». Мы сказали: «Здорово, давайте!»

И они провели этот конкурс, объявили призовой фонд около €1750, собрали студентов, привлекли несколько промышленных дизайнеров для разработки чертежей в САПР и всего, что нужно для изготовления переносного робота, которого можно взять с собой в музей. Приз вручили, нашелся человек, которому это действительно удалось; это полнофункциональная версия нашего робота, она раскладывается и легко поместится в переносной кейс. И полностью открытая, всеми разработками они поделились, а мы были просто поражены, и с удовольствием написали об этом в Сети, и рекламировали их.

Затем они стали ездить с ним по финским музеям, и мы начали получать фотографии, как финские дети смотрят за тем, как робот сканирует их личные фото и вещи. В конечном итоге, у них было три выездных дня в музеях, 300 человек пришли, чтобы увидеть этого робота. Около 30 человек прошли мастер-класс по тому, как его использовать и производить архивное сканирование. И все это стало возможным, потому что мы делились материалами, а они уже от этого отталкивались. Думаю, в этом и есть сила Open Source: вы выкладываете материал, делаете его доступным, даете ему толчок, но когда подключается сообщество, начинают происходить просто невероятные вещи. Все идет от этого.

ПОМОЩЬ УНИВЕРСИТЕТА ААЛТО

«Мы переписали ваш код Linux, можете дать нам доступ на GitHub.»

LXF: Вы можете прикинуть, как велико это сообщество?

TC: Если считать пользовательские группы, с которыми мы сотрудничали напрямую, то одна в Калифорнии, точнее, две, одна в районе Лос-Анджелеса, в библиотеке Santa Maria Public Library, мы туда ездили, проводили мастер-классы со студентами. Еще одна в Балтиморе, в Финляндии, в Бостоне, плюс те, куда мы отправляли наборы, и знаем, что с ними работают, но пока не так активно. Так что — по всему миру. На самом деле, все очень быстро разнеслось.

LXF: Как устроен Gado?

TC: В основном он состоит их пластиковых частей, изготовленных на 3D-принтере, оптимизированных так, чтобы их можно напечатать на каком-нибудь MakerBot, дома или в мастерской. Сам робот — это рука на гигантском сервомеханизме. На конце руки размещен привод манипулятора, а на нем — пневматическое подъемное устройство. Рука движется по кругу над стопкой фотографий, привод опускается, включается подъемник и берет одну из фотографий и помещает в центр круга, где расположен планшетный сканер, любой из существующих, мы поддерживаем все, что поддерживается в Linux, и есть еще куча всего, чтобы сделать то же самое в Windows. Затем он опускает фото на планшет сканера, настраивает ПО и сканирует фотографию в полном архивном разрешении. Обычно это 600 DPI. Снимок именуется в соответствии со способом, принятым в данном архиве — по названию коллекции или по расположению, что-то вроде этого. Затем разворачивается на 90 градусов, выключает подъемник и кладет фотографию в выходной лоток, а потом просто разворачивается и делает все по новой.

То есть делает все то же, что делал бы человек, перенося снимки на сканер и закрывая крышку. Единственное, что можно добавить к вышесказанному, что во входном лотке в начале цикла расположена зеркальная камера, или web-камера, делающая снимки оборотной стороны каждого изображения. Сейчас нам интересно другое: при оцифровке фотографий, вы получаете этот черновой скан. С очень высоким разрешением, в полный размер планшета, каждый такой файл весит 100 МБ. Они будут правильно поименованы, но когда у вас будет 120000 таких снимков, вы зададитесь вопросом: «И что мне делать дальше? Как мне получить из этого сырого материала нечто, чем можно поделиться с сообществом, или даже лицензировать, чтобы получать доход на поддержку архива?» Так что теперь мы занимаемся отсканированным материалом, пытаемся сделать так, чтобы посредством ПО распознавать, что на этих снимках, работать с метаданными и обработкой готовых снимков, и как можно более автономно.

Именно для этого нужны зеркалка или web-камера, снимающие оборотную сторону артефактов, на многих там есть пометки. Фотограф мог сделать снимок и сзади написать, что на нем, или поставить дату. Если он принадлежал кому-то другому, там может быть авторская печать, о чем нам следует знать... благодаря снимку с зеркалки, эти записи у вас есть. Сейчас мы готовим к выпуску ПО для оптического распознавания знаков для считывания текста, печатей, автоматического определения дат и добавления всего этого в базу данных, чтобы осуществлять поиск по материалам и добавлять метаданные можно было полностью автоматически, так же просто, как и сканировать.

LXF: И когда оно ожидается?

TC: Мы сейчас начинаем его бета-тестирование с разными организациями. Один из главных наших партнеров — компания Getty Images, мировой лидер в сфере лицензирования изображений. Когда вы видите какой-то исторический снимок на CNN, скорее всего, его предоставили именно они. Совместно мы работаем над тем, как показать наши фотографии миру. А также над тем, чтобы архивы приносили доход, и над их пополнением; такая система должна подпитываться изнутри.

Мы также сотрудничаем с некоторыми другими организациями, которые я, к сожалению, пока назвать не могу, в режиме бета-тестирования работая с другими коллекциями, отличными от тех, что мы сканировали сами.

LXF: Помимо фотографий, можно ли будет архивировать другие вещи?

TC: Возможно. Мы начинали этот проект, чтобы решить конкретную проблему. Изначально мы видели свою задачу в том, чтобы создать открытого робота для автономной оцифровки архивов менее чем за $500. Она была технической, конкретной, а цель — измеримой. Так было первые два — два с половиной года. Когда у нас образовалось целое сообщество и число оцифрованных снимков дошло до сотен тысяч, мы сказали: «OK. Цель достигнута! Такое устройство у нас есть. Что дальше? Что будет более востребованным?» И наша миссия расширилась до оцифровки и представления миру его цифровой истории. То есть к чисто технической добавилась идея сделать историю доступной. Точнее, визуальную историю: фотографии, визуальные артефакты. Как угодно, посредством любой технологии, где бы она ни хранилась, в личных коллекциях или музейных архивах. Поэтому расширив наши цели до визуального в целом, мы стали внедрять множество новых технологий.

Началось с проблем обработки материалов: как не повредить фотографии, перемещая их, как автоматически их сканировать, как понять, что на них? Как обрабатывать слайды? Как обрабатывать негативы, которым может быть сто лет? Как обрабатывать ветхий материал? То, что находится в переплетенном, перепечатанном виде? На все это мы ищем ответы. Думаю, наиболее эффективно мы сможем работать именно с визуальным материалом. В частности, фото, рисунки. Вероятно, текст и документы, больше заниматься оборотной стороной, дополнительными материалами, пытаясь понять те визуальные артефакты, что у нас есть. Конечно, хочется думать, что мы сможем оцифровать всю коллекцию...

Теперь, когда наша миссия уже не чисто техническая, мы смотрим вокруг и говорим: «Мы создали эту технологию, сделали ее доступной, теперь над ней работают по-настоящему увлеченные технари, но это тоже проблема». Мы отдали ее архивам, но сделать так, чтобы они действительно решили попробовать этого открытого робота в работе с их бесценными коллекциями, вписаться в свой бюджет, найти людей, которые сделают что угодно, так же сложно. Мы выложили ее в доступ, но просто выложить недостаточно. Нужно ее продвигать, работать с этими организациями, предельно упрощать ее, как мы упростили разработчикам создание платформы, выпустив набор.

В последнее время мы работаем напрямую с архивами, как с *Afro American Newspaper*, переходя от просто робота, которого можно создать, к готовому решению для оцифровки архива. Робот и сотрудничество с Getty Images позволяют нам попасть в архивы и бесплатно, без начальных вложений, оцифровывать огромные части коллекций. И благодаря тому, что мы можем получать доход от лицензирования материалов, мы позволяем архивам платить за оцифровку за счет уже составленных коллекций.

Можете представить себе, как это важно. У вас есть коллекция, но нет средств, чтобы привлечь людей, недостаточно технических знаний, чтобы использовать робота — даже зная о его существовании. Теперь архивы могут обратиться к нам, и мы поможем им с этим. Без каких-либо начальных вложений. Это не только оцифрованные материалы, открытые всему миру; мы заботимся о сохранении истории, а они могут получать доход, чтобы платить за дальнейшую оцифровку или что угодно еще.

Для нас это был новый шаг, и прекрасный пример тому — наша новая коллекция, которую мы цифровали таким образом: она принадлежит фоторепортеру, который 17 лет жил и работал в Сан-Франциско в газете *Sun Reporter*. Он умер в 1992 г., завещав всю свою коллекцию сестре. Исследователи ею занимались, но оцифровать ее возможности не было. Поэтому подключились мы и помогли перевести эти материалы в электронный формат и сделать доступными.

LXF: Сколько человек у вас в команде?

TC: Смотря как считать. На основной работе нас двое, а если учитывать сообщество, создавшее робота, наверное, около 50 человек. И это тоже плюс Open Source: мы можем ездить и заниматься этим, потому что расходов почти никаких. У нас есть преданное, очень опытное и знающее сообщество, оно занимается аппаратной поддержкой и помогает нам при возникновении проблем в архивах. С проприетарной системой нам бы понадобилась целая команда исследователей. А мы можем предоставить это целому сообществу. »

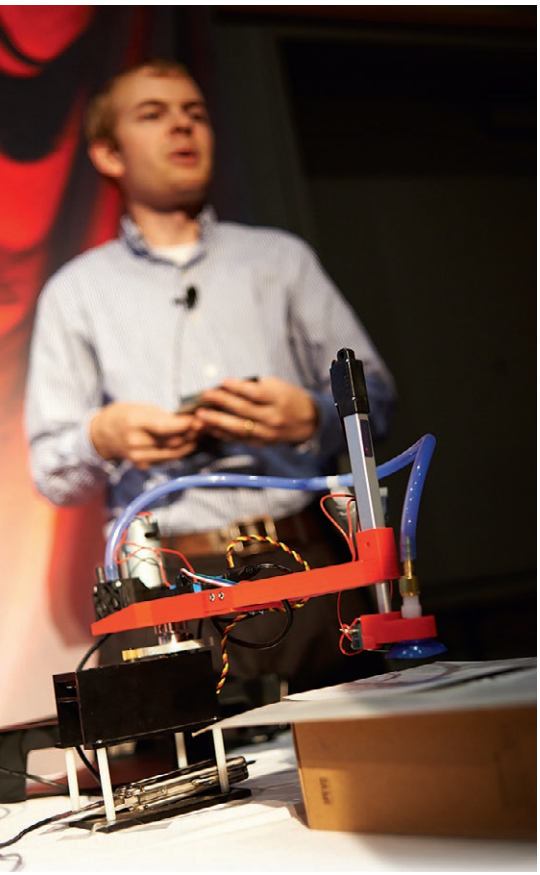
LXF: А вы не опасаетесь, что кто-нибудь, получив доступ к архиву, случайно проявит неосторожность и нанесет ущерб его уникальному содержимому? Не страшно? Надеюсь, такого еще не случалось.

TC: Нет, за все время работы ничего не пострадало. Ни мы, ни другие наши группы не повредили ни один артефакт — насколько нам известно! Мне кажется, как только ты доверяешь что-то сообществу, все начинают вести себя на удивление ответственно и профессионально. Люди не будут делать робота, чтобы цифровать что попало. По нашему опыту, они создают группу пользователей и начинают использовать эту технологию по тем же причинам, что и мы. Им нравится делать историю доступной. Многие занимаются этим в составе более крупной организации, так что если кто-то получает доступ к архиву, он, вероятно, работает с теми, у кого этот материал есть, и мы видели, что люди сотрудничают. Случай в Финляндии — прекрасный пример того, как университет спонсировал исследования, проводившиеся в музеях и нужные им. Но, знаете, я мыслю шире, это риск, на который нужно пойти. Наша миссия — оцифровать всю визуальную историю человечества, и она очень велика.

LXF: Целенаправленное сотрудничество — это очень важно. Заинтересованы ли в проекте Gado международные партнеры?

TC: О да, очень! Отличный пример — компания под названием Capture, в Великобритании. Мы работали с ними над добавлением подписей к готовым изображениям. Как мы уже говорили, большая проблема в том, как после оцифровки добавить метаданные. Как помочь людям найти то, что вы оцифровали? С Capture мы сотрудничали относительно наших самых важных для истории материалов. У них большой опыт в изучении истории, стоящей за фотографией, определения того, кто на ней, когда она была сделана, и прочего. Совместно мы добавили эти метаданные к некоторым коллекциям. Многое будет сделано с помощью новой системы, многое будет делаться автоматически, но есть материалы, где не обойтись без совета экспертов. Такие партнеры могут этот совет предоставить.

Поразительно и то, что вот мы выложили эти снимки, с помощью нашего партнера Getty Images, а о них знают не только в США, но и по всему миру. И сначала эти снимки привлекли серьезное внимание в Америке. Их использовали CNN, Scholastic взяли несколько наших фото в свой учебник.



О ТРУДНОСТЯХ

«Мы понятия не имели, возможно ли такое, но тем не менее мы это сделали.»

Но в последние пару месяцев они стали появляться по всему миру. Повсеместно, в Японии, в Индии, фото из одного здания в Балтиморе разнеслись на тысячи километров, напечатаны в китайских учебниках. Так что именно интернациональный компонент — не только с технической стороны, как в Финляндии, но и с программной — придает этим снимкам значение, уже то, где они окажутся, охватив весь мир.

Именно поэтому, переводя в электронный формат ту или иную сферу истории, часть истории одного народа, мы хотели, чтобы это движение стало всемирным; даже когда мы начинали в Балтиморе и Сан-Франциско, мы видели в этом что-то международное. И на наших глазах оно становится таковым.

LXF: Что вы можете посоветовать — если можете — тем, кто стремится стать изобретателем и пытается дать старт своим проектам?

TC: Думаю, главное — понимать, что это не быстрый процесс: вы его начинаете, не зная, к чему все придет. Вы даже не знаете, будет ли он работать, зато значительное число людей будет вас убеждать вас, что не будет, или будет не так, как вы задумали.

Будет масса препятствий и причин им не заниматься. Но смысл в том, что если вы верите в свою идею, вы ее воплощаете.

Когда мы начинали, было столько вопросов. Это была безумная идея — как создать за $500 робота, чтобы он работал, при этом не повредив бесценный артефакт? Мы понятия не имели, возможно ли такое, но тем не менее мы это сделали.

Все получилось так, как мы и вообразить не могли. Так что лучший совет — дерзайте! Пробуйте, смотрите, что получится. Будьте готовы, что это займет много времени, будьте готовы идти наперекор, будьте готовы, что временами будет казаться, что все пропало.

Когда мы отправили свой первый набор, возникла проблема с одной из частей, у многих она просто не работала. И тогда мы обратились к сообществу: «Как вы думаете, чем можно ее заменить?». И получили массу прекрасных ответов, разработали новую, распространили ее по всем, и все заработало! Как же без трудностей... но это и есть путь, который может привести вас туда, куда вы и не мечтали. LXF

Новое поколение средств защиты

Межсетевые экраны ССПТ, не имеющие IP-адреса

ССПТ-2 — это сертифицированное ФСТЭК, ФСБ и ГАЗПРОМСЕРТ средство защиты информации нового поколения, реализующее функции межсетевого экрана, но при этом остающееся «невидимым» для любых протоколов и тестовых воздействий, что достигается за счет отсутствия физических и логических адресов на его фильтрующих интерфейсах. **ССПТ-2 невозможно обнаружить никакими известными средствами удаленного мониторинга сети.**

Скрытность функционирования межсетевого экрана повышает надежность системы защиты в целом и существенно упрощает процедуру установки ССПТ-2 в компьютерные сети и функционирующие на их основе информационные и телематические системы.

Назначение устройства

Основное средство защиты для реализации различных политик информационной безопасности с помощью:

— фильтрации пакетов на канальном, сетевом, транспортном и прикладном уровнях;

— управления транспортными соединениями между отдельными узлами ЛВС или виртуальной ЛВС (VLAN);

— контроля контента данных на прикладном уровне с учетом направления, времени и типа протоколов передачи трафика.

Дополнительное устройство защиты для:

— обеспечения безопасности функционирования ранее установленных в компьютерной сети средств защиты и устройств маршрутизации;

— мониторинга трафика с возможностью анализа данных регистрации пакетов по различным критериям и интеграции с IDS;

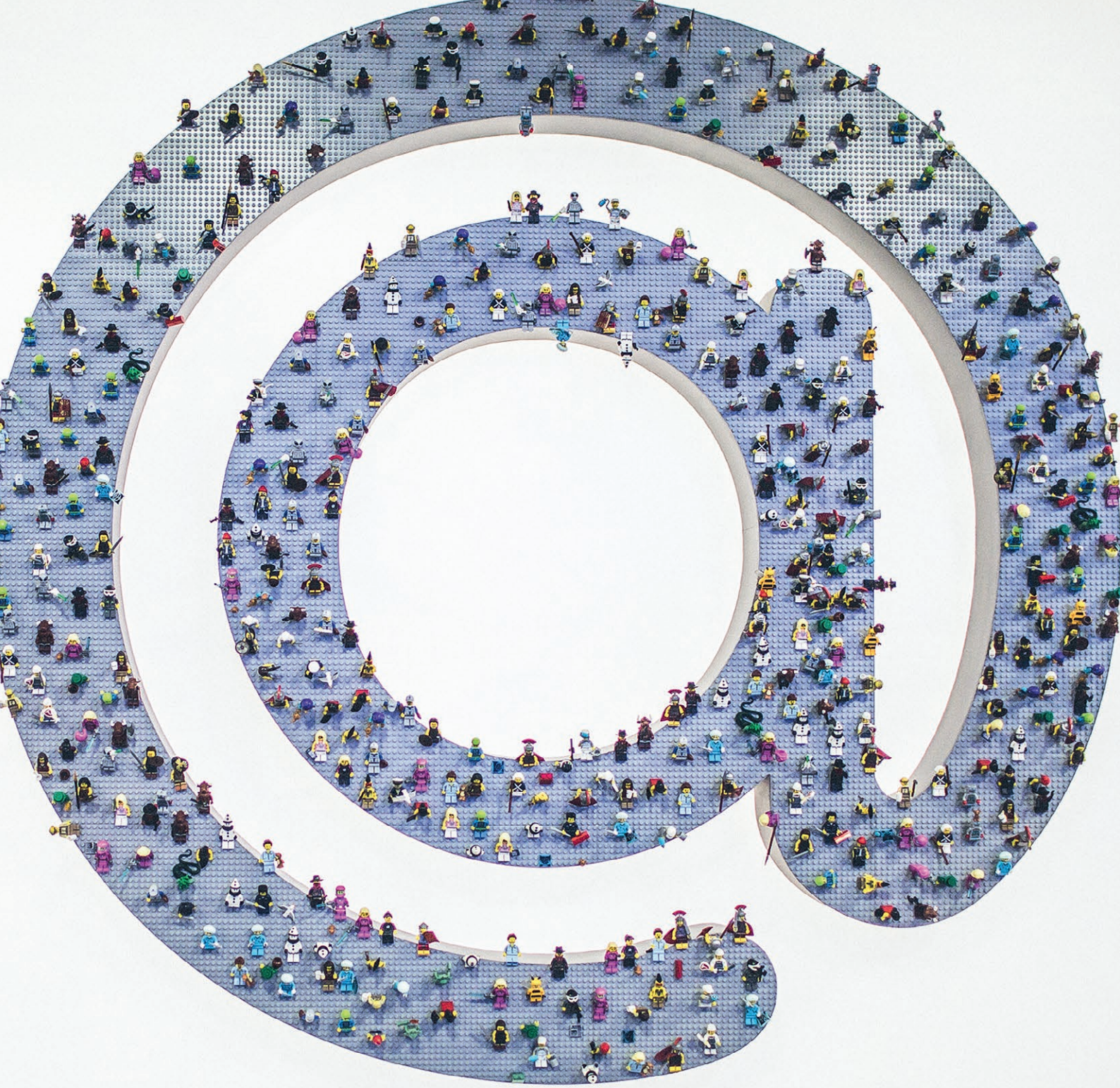
— обеспечения функционирования сетевых распределенных телематических приложений и GRID-ресурсов.

Москва +7 (499) 271-49-54

Санкт-Петербург +7 (812) 309-06-86

Linux-эксперт для вашего бизнеса. www.linuxcenter.ru

Linux center



Mail.Ru Group и свободное ПО

Игорь Штомпель выяснял взаимоотношения этих понятий с Александром Горным и Яном Киселем из Mail.Ru Group.

Даже тем, кто довольно далек от компьютерной техники, сервис Mail.Ru прекрасно известен, хотя они могут и не подозревать о его связи со свободным ПО и Linux. Нам удалось побеседовать с людьми, которые стоят за этим и многими другими сервисами компании Mail.Ru Group, обеспечивая ее деятельность: Александром Горным, директором по информационным технологиям Mail.Ru Group, и Яном Киселем, руководителем группы разработчиков в Поиске Mail.Ru.

Александр Горный с отличием окончил Факультет вычислительной математики и кибернетики Московского государственного университета им. М. В. Ломоносова, после чего начал работу в компании Mail.Ru, прошел путь от программиста до руководителя отдела Почты Mail.Ru. В марте 2006 г. перешел в интернет-холдинг «Рамблер» на должность директора по разработке web-проектов. С осени 2006 г. руководил сервисом «Почта.ру» компании «Медиа Мир» (группа компаний РБК). Затем совмещал эту должность с позицией технического директора компании и руководителя проекта Readme.ru. В 2009 г. — заместитель генерального директора компании «Медиа Мир». В 2010 г. возглавил проект ICQ в Mail.Ru Group. В 2011 г. назначен директором по информационным технологиям Mail.Ru Group.

Ян Кисель — выпускник Факультета автоматики и вычислительной техники Вятского государственного университета, в компанию пришел в 2008 г. как программист на C++. Основная его специализация — back-end- и middle-end-разработка, а также проработка архитектуры высоконагруженных систем. Несмотря на статус руководителя, в свободное время любит попрограммировать.

LXF: Чем вы занимаетесь в Mail.Ru Group?
Александр Горный: Я директор по информационным технологиям, занимаюсь различными активностями компании, которые выходят за рамки отдельных проектов: от общей портальной авторизации Mail.Ru до нашей собственной СУБД *Tarantool*.

LXF: Как Linux используется в ваших проектах? Что это за проекты?
АГ: Linux — наша основная ОС. Она используется на тысячах серверов практически во всех проектах Mail.Ru Group.

LXF: Как и когда впервые познакомились с Linux?
АГ: У нас был специальный курс в университете, там давали и теорию UNIX, и возможность попрактиковаться.

LXF: Расскажите о ваших первых проектах на Linux. Какие первые проекты на базе этой операционной системы были у вас в Mail.Ru Group?
АГ: Не считая курсовых работ, первыми моими Linux-проектами были проекты портала Mail.Ru: Авто Mail.Ru, Афиша Mail.Ru и другие.

»

LXF: Есть ли предпочтения в отношении дистрибутивов Linux? С какими удалось поработать?
АГ: Дистрибутив должен быть знаком системному администратору, а в большой компании — унифицирован. Других предпочтений у меня нет, хотя видел я их штук шесть, наверное.

LXF: Как ваша компания взаимодействует с Open Source-сообществом? Расскажите о вкладе вашей компании в Open Source.
АГ: Мы активные пользователи многих Open Source-продуктов с самыми разными лицензиями: от того же Linux до *MySQL*. В нашей компании работают контрибьюторы многих известных продуктов (например, *Postgres*).

Сами мы исторически развиваем несколько небольших OS-проектов и один крупный — *Tarantool*. *Tarantool* — это application–сервер, поддерживающий язык Lua со встроенной NoSQL-СУБД. Используя его, вы можете писать Lua-программы, делая запросы к базе данных в том же адресном пространстве, и не терять процессорное время на переключение контекста и работу с сетью. Если же такая скорость вам не нужна или Lua — неприемлемый для вас язык разработки, то *Tarantool* можно запустить как отдельный процесс. В этом случае вы получите классическую отдельно стоящую СУБД с очень богатой для NoSQL функциональностью: поддержкой транзакций, репликацией, хранимыми процедурами и вторичными индексами. Работает он при этом в десятки раз быстрее любой SQL-базы.

Сервис MAPS.ME, который мы приобрели в конце прошлого года, использует свободные картографические данные Open Street Map. Мы пообещали сделать приложение MAPS.ME открытым — это еще один крупный вклад в Open Source-движение, который нам пока предстоит сделать.

LXF: Расскажите о главных новшествах последнего релиза *Tarantool*. Например, появилась ли в нем поддержка *systemd*?
АГ: Главное нововведение именно последней версии — это первый стабильный релиз дискового движка для хранения данных, который позволяет управлять терабайтами информации на одной машине. Говоря о том, что мы сделали за последние несколько лет, главное — конечно, переход от key–value хранилища к текущей модели application–server со встроенной СУБД. Это позволяет писать на *Tarantool* полноценные сервисы с достаточно сложной бизнес-логикой без использования других баз и внешних языков программирования. Все необходимое для этого, от вторичных индексов до хранимых процедур, уже есть. Например, наша демонстрационная песочница, try.tarantool.org, полностью написана на *Tarantool*, без использования Python или других языков программирования.

Поддержка же *systemd* — это очень небольшая в подобных масштабах задача. У нас она реализована. В частности, есть рабочий service-файл, включенный в дистрибутивы для всех систем, в которых работает *systemd*.

LXF: Каково ваше отношение к этой системе инициализации? В чем ее преимущества, а в чем недостатки?
АГ: На мой взгляд, *systemd* до сих пор слишком сырая и скорее вредит работе. Совсем недавно разворачивали кластер на основе CentOS 7, проводили нагрузочное тестирование. В итоге столкнулись с зависанием системы, невозможностью перезапустить сервисы, зомби-процессами и всеми прелестями ранних Linux, про которые, казалось, уже можно было навсегда забыть.

LXF: С какими технологиями в Linux удалось поработать?
АГ: Хорошее забывается быстро, плохое остается в памяти навсегда. Все-таки любая, даже самая мелкая авария у нас способна задеть тысячи, если не миллионы пользователей. Так что до сих пор лучше всего помню проблемные технологии десятилетней давности: недостатки Reiser FS и ранних реализаций тредов.

LXF: А что представляет собой архитектура программной составляющей Mail.Ru Group?
АГ: У нас очень много различных проектов, которые развивались своими путями, на основе разных архитектур. Обрисовать даже в самом общем виде все это многообразие в рамках одного интервью совершенно невозможно. Но мы стараемся быть открытыми миру, много рассказываем о своих технологиях на конференциях, ведем блог на Хабре: http://habrahabr.ru/company/mailru/.

LXF: В компаниях, которые поддерживают высоконагруженные сервисы, важное место занимает организация хранения и обработки данных. Как этот процесс реализуется в Mail.Ru Group?
АГ: Последний наш проект, имеющий дело с многопетабайтными хранилищами — Облако Mail.Ru. Очень подробно о его внутреннем устройстве мы писали тут: http://habrahabr.ru/company/mailru/blog/211340/.

LXF: Какое программное обеспечение используете для работы с большими данными?

АГ: Помимо Облака, в Mail.Ru Group много других продуктов, которые оперируют большими объемами данных. В некоторых случаях это классические Big Data — «большие данные», разнообразные и плохо структурированные. В других случаях это просто большое количество понятных данных — например, если речь идет о хранилище фотографий.

Для каждого конкретного случая мы используем наиболее подходящие технологии. Часть из них — классические Open Source-решения. Другая часть — наши собственные системы, которые мы написали для конкретного проекта.

К последним относится тот же *Tarantool*, который используется для хранения и быстрого доступа к «маленьким большим данным» (то есть когда речь идет о сотнях гигабайт, а не о сотнях терабайт). При этом данные относительно структурированы и к ним нужен постоянный и очень быстрый доступ.

Мы рады, что *Tarantool* применяется не только у нас, но и за пределами Mail.Ru Group. Это уникальное решение, сочетающее в себе application–server и in-memory-хранилище данных — возможно, лучшее среди аналогов на сегодняшний день.

LXF: Какие знания и навыки необходимы для работы с Big Data?

АГ: Если говорить о личном опыте решения сходных задач, то для работы с Big Data необходим, в первую очередь, профессиональный кругозор. Он позволяет увидеть реальную задачу, которая стоит за ее технической постановкой. Очень часто техническое задание неправильно или не совсем точно сформулировано, и его можно переформулировать, чтобы решить задачу эффективнее.

Безусловно, необходимо знать конкретные инструменты и уметь работать с ними, но выучиться этому несложно. Знания в области статистики или простой математики гораздо полезнее, куда тяжелее приобретаются и могут быть использованы значительно эффективнее.

LXF: Какое место в этом ряду можно отвести операционной системе Linux?

АГ: Как я говорил ранее, инструменты не очень важны: освоить их легко. В нашей компании, как, вероятно, и в большинстве других, все Big Data-системы работают поверх Linux. Каждый, кто знает все остальное, Linux освоит быстро.

LXF: Что можно отнести к основным задачам Big Data?

АГ: Наверное, главная философская задача Big Data — это возможность ответить на неявный вопрос.

Вся информация, которую мы копим, посвящена каким-либо конкретным действиям: пользователь куда-то зашел, что-то купил, с кем-то подружился, и так далее. Имея эти явные знания, легко отвечать на вопросы: «Сколько пользователей купило эту книжку? Сколько книжек купил этот пользователь? Какой пользователь купил больше всего книжек?» и т.д. Big Data же позволяет отвечать на неявные вопросы, ответы на которые не содержатся в данных напрямую. «Какой жанр любит этот пользователь? Что он сделает или купит в будущем? Что ему наиболее интересно? Что ему стоит предложить?» Big Data позволяет все это узнать или предугадать.

LXF: Какое место, на ваш взгляд, занимает технология LAMP в реализации высоконагруженных систем?

АГ: LAMP — это прекрасный, уникальный и очень полезный стек технологий, который является оптимальным для большинства сервисов. В том числе он очень важен для высоконагруженных систем. Понятно, что, например, ядро Почты Mail.Ru на Python или PHP писать невыгодно, потому что реализация потребует непропорционально много железа. Но даже очень сложные проекты состоят не только из ядра. Если учитывать объем кода, число функций, количество разработчиков, занятых на проекте, то получается, что «обвязки», дополнительные сервисы и прочее составляют едва ли не больше половины высоконагруженного проекта. И для всего этого LAMP подходит идеально.

Разрабатывать на LAMP можно очень быстро. Маленький цикл внесения изменений, маленькая цена и высокая скорость разработки — одни плюсы и почти никаких минусов. Если подвести итог, с использованием данного стека можно и нужно делать что угодно — от небольших до высоконагруженных проектов.

Простые задачи должны быть просто реализуемы — для начала работы с Hadoop пользователю не обязательно досконально знать, как он устроен.

LXF: Расскажите об экосистеме Hadoop. Какова в ней доля Open Source-решений?

ЯК: Основными и самыми крупными компонентами Hadoop изначально были HDFS — распределенная файловая система, MapReduce — система распределенных вычислений в соответствующей парадигме и *HBase* — NoSQL распределенная база данных. В дальнейшем Hadoop развился в YARN — систему, предоставляющую доступ к вычислительным мощностям кластера для множества других различных парадигм. Такие библиотеки, как *Apache Spark*, *Apache Storm*, могут работать поверх YARN и на сегодняшний день успешно применяются для решения тех задач, с которыми не справляется MapReduce.

Есть еще некоторое количество вспомогательных подсистем: Apache Pig, Apache Oozie, Apache Flume и другие, которые могут помочь и ускорить решение некоторых конкретных задач.

Сейчас все компоненты Hadoop могут использоваться независимо друг от друга и успешно функционировать. Однако в том же HBase есть некоторые оптимизации, которые делают его совместное использование с HDFS более удобным, нежели с другими РФС.

Hadoop — полностью Open-Source-система, изначально разрабатываемая как открытая реализация Google BigTable. Немалая часть изменений и исправлений делается сообществом (в современном Hadoop имеются в том числе и наши исправления и добавления).

О НАДЕЖНОСТИ

«Иногда наш кластер продолжал работу, лишившись четверти серверов.»

LXF: Что представляют собой системные принципы Hadoop?

[Hadoop — свободное ПО: фреймворк, библиотеки и утилиты для обеспечения работы программ на кластерах, способных включать тысячи узлов. Широко применяется для статистики и обработки больших данных. Проект поддерживается Apache Software Foundation (http://hadoop.apache.org/), — *прим. ред.*]

Ян Кисель: Прогнозируемое масштабирование — большее количество серверов выполняет большую работу за то же время. Hadoop спроектирован для работы на множестве серверов относительно средней конфигурации.

Отказоустойчивость — выход из строя одного сервера никак не влияет на работоспособность кластера в целом. Иногда были моменты, когда наш кластер продолжал работать, лишившись четверти серверов.

Локальная работа с данными — Hadoop старается по максимуму исполнять код именно на тех серверах, где расположены обрабатываемые данные. Это снижает сетевую нагрузку и ускоряет выполнение задач.

LXF: Что такое дистрибутивы Hadoop? Из каких компонентов они состоят?

ЯК: Так как Hadoop — комплексная система, несколько компаний предлагают свои решения по упрощенной установке и настройке компонентов системы для реального кластера. Из самых известных стоит упомянуть Hortonworks, Cloudera и MapR. Все эти компании выпускают свои релизы Hadoop и других компонентов, которые проходят через их внутреннее тестирование и гарантированно совместимы внутри одной версии.

Также компании публикуют подробные инструкции по установке, настройке и апгрейду Hadoop.

Все популярные компоненты, из которых состоит Hadoop, обычно в этих дистрибутивах имеются, а если какого-то и нет, то почти всегда его можно без каких-либо сложностей добавить самостоятельно.

Так как разработка и тестирование этих дистрибутивов начинается только после официальных релизов, они обычно отстают на некоторое число версий от транка Apache. Поэтому, если хочется применять самые последние новинки, придется »

собирать и конфигурировать самостоятельно, используя данные напрямую с официального сайта.

LXF: Если говорить о Hadoop API — в чем его преимущества? Что хотелось бы в него добавить?

ЯК: Основным преимуществом Hadoop API можно назвать низкий порог вхождения. В этом, правда, кроется и один из недостатков — многие начинают использовать Hadoop как площадку для запуска своей однопоточной задачи во много потоков. При этом они не задумываются о том, какие процессы проходят внутри и как на это реагирует кластер. В итоге кластер используется неэффективно, а производительность других пользователей страдает.

Обычно, когда мы понимаем, что определенная часть Hadoop нас не устраивает, мы патчим ее самостоятельно. Нередко это узкоспециализированные решения, поэтому до trunk'а они не всегда доводятся.

LXF: Какое место занимает Hadoop в Поиске Mail.Ru? Какие трудности возникли в процессе миграции на его использование?

ЯК: Мы используем Hadoop только для оффлайн-обработки данных. Запросы, для выполнения которых важно время ответа, не затрагивают Hadoop. Обработка поискового запроса пользователя — это набор своих демонов, написанных преимущественно на C++. А вот оффлайн-обработка, для которой более важна пропускная способность, выполняется преимущественно на Hadoop-кластере.

До использования Hadoop для обработки мы использовали самописные решения, не обладавшие ни достаточно удобным масштабированием, ни общим API. Иными словами, для каждой оффлайн-подсистемы существовала своя независимая схема работы.

С переходом на Hadoop в поиске пришлось многие вещи написать, что называется, «с нуля», поскольку парадигма MapReduce подразумевает подход, отличный от того, который мы использовали прежде. Вдобавок полноценное использование Hadoop возможно лишь на Java, тогда как наш код был преимущественно на C++. Конечно, мы сохранили множество наработок, продолжив использовать прежние модули на C++ через JNI.

Дальнейшие же трудности были скорее эксплуатационными. Об этих особенностях эксплуатации мы неоднократно рассказывали на технических конференциях.

LXF: Почему стали использовать *HBase*? Что в ней храните?

ЯК: В *HBase* мы храним информацию, паттерн доступа к которой похож на «записали, многократно считали, переписали кусок, ...».

Почему именно *HBase*? К тому времени у нас уже имелся положительный опыт работы с HDFS и MapReduce, так что

использовать систему, интегрированную в этот фреймворк, мы посчитали правильным.

Таблицы *HBase* занимают у нас порядка 4 ПБ. В них хранится web-контент, множество извлеченных и рассчитанных свойств документов, ссылочный граф, а также запросы пользователей.

LXF: Расскажите об архитектуре HDFS. В чем ее сильные и слабые стороны?

ЯК: Индексом файлов в HDFS является NameNode — она хранит в памяти записи о каталогах и описания файлов. Файлы в HDFS представлены относительно большими блоками (как правило, 128 МБ), которые, в свою очередь, реплицируются. Адреса реплик каждого блока и хранятся в NameNode. При запросе NameNode перенаправляет клиентский код к DataNode — движкам, обслуживающим операции с блоками. Как следствие, в HDFS практически отсутствует узкое место в плане пропускной способности.

Сильные и слабые стороны HDFS скорее являются особенностями ее ориентации на MapReduce-задачи. К сильным, безусловно, относится отказоустойчивость — потеря реплики блока автоматически инициирует восстановление ее из оставшихся. Большой плюс — масштабируемость и скорость работы.

Из минусов: HDFS является append-only файловой системой, то есть перезаписать часть файла нельзя — можно только удалить и записать его заново. Простота организации NameNode также порой может сыграть отрицательную роль: большое количество мелких файлов приведет к огромному расходу памяти NameNode, и взаимодействия с ней могут стать узким местом.

ОБ АРХИТЕКТУРЕ

«Простота организации также порой может сыграть отрицательную роль.»

LXF: Какие преимущества на практике дает оптимизация файловой системы HDFS под последовательное чтение?

ЯК: Такая оптимизация HDFS очень тесно связана с парадигмой MapReduce. Эта парадигма параллельной обработки подразумевает обработку относительно больших блоков данных. Поэтому «заточенность» HDFS под последовательный доступ играет здесь положительную роль. К тому же это упрощение положительно сказывается на работе прочих компонентов Hadoop. Например, DataNode и NameNode.

LXF: Поверх каких файловых систем может работать HDFS?

ЯК: HDFS является сетевой надстройкой над файловой системой. Каких-то необычных требований к системной ФС, насколько нам известно, она не имеет. В нашей практике эксплуатации это ext3/ext4.

LXF: Почему HDFS не подходит для работы с многопоточной записью?

ЯК: Думаю, причина здесь в простоте организации — HDFS не несет каких-либо файловых синхронизаций. Но обычно при использовании HDFS это и лишено смысла. Вы просто пишете несколько независимых файлов из разных задач, а их конкатенация будет являться результатом работы.

Это также соответствует парадигме MapReduce. Так, например, при выходе машин из строя их подзадачи будут запущены повторно, и часть выходных файлов будет просто записана заново. С многопоточной записью подобная задача была бы проблемой.

LXF: Возвращаясь к файловым системам — в чем преимущества использования Ext4 на высоконагруженных серверах? Чего ей не хватает? Есть ли альтернативы?

ЯК: В поиске мы довольно слабо зависим от конкретной файловой системы на серверах: подавляющая часть онлайн-обработки происходит в памяти, а для работы оффлайн куда важнее производительность дисков и сетевой инфраструктуры.

LXF: Какие демоны используются для управления HDFS? В чем отличие Namenode от Secondary Namenode?

ЯК: Основной мастер-демон — Namenode, который хранит информацию о расположении и назначении блоков.

На каждой из машин кластера должен быть демон Datanode, который занимается отправкой и приемом собственно данных от пользователей, сохраняя их в виде фрагментов, равномерно распределенных по кластеру.

Namenode может иметь Hot-Standby демона (Backup Node), который перехватывает управление в случае падения основного.

Secondary Namenode (с некоторого времени называется Checkpoint Node) — вспомогательный демон, который в режиме реального времени объединяет дамп метаинформации с Namenode и Write-Ahead-Log на ней, чтобы при экстренном перезапуске большой объем WAL не стал причиной задержки в перезапуске на несколько часов. В случае использования Backup Node не нужен, так как он, по сути, выполняет ту же операцию и держит эти же самые данные в памяти.

LXF: В чем особенности работы с HDFS из консоли [shell] и через API с помощью языка программирования Java?

ЯК: Далеко не все команды из LSB есть в наличии по умолчанию. Иногда это неудобно — того же **find** часто не хватает. Но при необходимости нужный софт легко пишется самостоятельно (или готовый уже есть в Интернете). API, предоставляемый в оригинале, достаточно прост и функционален. С его помощью записать файл в HDFS не сложнее, чем записать файл из Java в локальную ФС.

LXF: Используете ли вы Pig? Если да, то для каких практических целей?

ЯК: Нет, Pig мы не используем. Подавляющее большинство наших задач очень сложно ложится на идеологию Pig, а поддерживать его ради тех немногих, для которых он бы хорошо подошел, мы не стали.

LXF: Если говорить о возможностях масштабирования программной составляющей инфраструктуры — как осуществляется, какие возможности заложены в нее при ее реализации?

ЯК: Общих принципов разработки, мне кажется, не так уж много. Во-первых, нужно минимизировать связи между различными элементами системы. Во-вторых, проблема с любой из этих связей не должна быть неожиданной. В-третьих, все используемые алгоритмы должны иметь линейную сложность, а если это невозможно, масштабирование должно быть продумано заранее, а не оставлено на потом.

LXF: Ну и напоследок парочка менее серьезных, но от этого не менее важных вопросов. Какое программное обеспечение используете? Операционные системы, почтовые клиенты, браузеры и т.д.?

ЯК: В качестве ОС я использую Windows 7, в качестве браузера — *Амиго*, в качестве почты — web-интерфейс Mail.Ru. Программ, которыми я пользуюсь регулярно, очень немного: кроме *Амиго*, это Агент Mail.Ru, *Putty*, *Microsoft Office* и *Far* — вот, по сути, и все. Что-то небольшое для разовых задач иногда программирую сам.

LXF: Есть ли аппаратные предпочтения, в том числе в отношении мобильных устройств?

ЯК: Для моего набора программ достаточно любого железа, при выборе ноутбука смотрю только на соотношение веса и времени работы от батареи. Смартфон — iPhone, планшет — 7" Samsung Galaxy.

LXF: Инструменты разработки? Среды?

ЯК: Давайте, давайте пофлеймим на тему *Vim* vs *Emacs*. Это будет очень весело и интересно! [*улыбается*]

LXF: Используете социальные сети? Web-сервисы?

ЯК: Не фанат, но захожу в соцсети довольно регулярно.

LXF: В каких ИТ-мероприятиях, конференциях принимали участие, в каких планируете?

ЯК: Mail.Ru Group участвует во всех значимых для отрасли событиях. Недавно, например, были четыре очень хороших доклада от наших сотрудников на фестивале РИТ. Последняя крупная конференция, в которой я участвовал лично — прошлогодний Highload, а на небольших встречах я бываю чуть ли не каждый месяц. LXF

Systemd

Позвольте **Джонни Бидвеллу** убедить вас в том, что *Systemd* не съест ваш компьютер, да и вообще отнюдь не плох.

```
ivities                                                    Thu 12:22
  • jbmachine
      State: running
       Jobs: 0 queued
     Failed: 0 units
      Since: Thu 2015-05-07 09:46:23 BST; 2h 34min ago
     CGroup: /
             ├─1 /sbin/init
             ├─system.slice
             │ ├─avahi-daemon.service
             │ │ ├─3606 avahi-daemon: running [jbmachine.local
             │ │ └─3607 avahi-daemon: chroot helpe
             │ ├─dbus.service
             │ │ └─256 /usr/bin/dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation
             │ ├─dhcpcd.service
             │ │ └─252 /usr/bin/dhcpcd -q -b
             │ ├─wpa_supplicant.service
             │ │ └─1280 /usr/bin/wpa_supplicant -u
             │ ├─accounts-daemon.service
             │ │ └─282 /usr/lib/accountsservice/accounts-daemon
             │ ├─colord.service
             │ │ └─1403 /usr/lib/colord/colord
             │ ├─systemd-journald.service
             │ │ └─150 /usr/lib/systemd/systemd-journald
             │ ├─udisks2.service
             │ │ └─1314 /usr/lib/udisks2/udisksd --no-debug
             │ ├─upower.service
             │ │ └─610 /usr/lib/upower/upowerd
             │ ├─systemd-logind.service
             │ │ └─254 /usr/lib/systemd/systemd-logind
             │ ├─sshd.service
             │ │ └─268 /usr/bin/sshd -D
             │ ├─systemd-udevd.service
             │ │ └─167 /usr/lib/systemd/systemd-udevd
             │ ├─polkit.service
             │ │ └─322 /usr/lib/polkit-1/polkitd --no-debug
             │ ├─gdm.service
             │ │ └─267 /usr/bin/gdm
             │ └─rtkit-daemon.service
             │   └─852 /usr/lib/rtkit/rtkit-daemon
             └─user.slice
               ├─user-1000.slice
               │ ├─user@1000.service
               │ │ ├─2417 /usr/lib/systemd/systemd --user
               │ │ └─2418 (sd-pam)
lines 1-44/118 25%
```

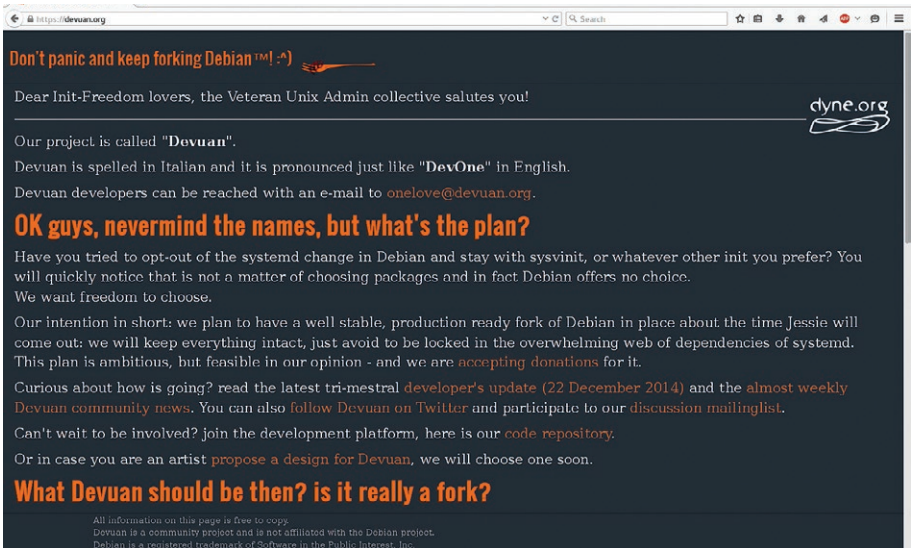
***Systemd*, ставший системой init по умолчанию в Fedora 15 в 2011 г., несмотря на неоднозначное отношение к себе, помаленьку стал применяться и другими дистрибутивами. Когда его приняли самые последние версии Debian и Ubuntu, только Gentoo и Slackware остались основным оплотом старой доброй *SysVinit*.**

Конечно, ряд более мелких нишевых дистрибутивов поступают так же, но то, что мы не наблюдаем массового исхода к любому из этих дистрибутивов, уже является подтверждением того, что пользователи, как минимум, вполне довольны производительностью *Systemd* и их совершенно не волнуют идеологические проблемы вокруг него. На самом деле, пользователи рабочего стола стали свидетелями значительного улучшения времени загрузки благодаря его параллелизации загрузочных сервисов, а то, как он унифицирует разношерстную свору скриптов и демонов, делает его более чем привлекательным для начинающих сисадминов, вот почему мы и рассказывали о *Systemd* довольно подробно в последних выпусках [Учебники, стр. 66 **LXF191/192**, стр. 62 **LXF188**]. Однако в *Systemd* постоянно возникают новые функции, и некоторые пользователи даже не подозревают о том, что там только что появилось. А мы заглянем во внутренности *Systemd* и посмотрим, что с ним происходит и как сделать так, чтобы он не делал того, что нам не нужно. Но сначала немного истории.

> «То, как он унифицирует скрипты и демоны, делает его еще более привлекательным.»

Systemd — это менеджер системы и сервисов. Его основной образ действия — инициализация системы, поэтому его основной бинарник имеет символическую ссылку на файл /sbin/init, который запускается как Process ID (PID) 1 после того, как загрузится ядро. Затем *Systemd* добросовестно запустит все сервисы (в буквальном смысле становясь матерью всех процессов) и продолжит управлять ими до момента выключения, после чего он выгружает себя, и компьютер останавливает работу и выключается.

Предыдущая система инициализации, известная, как *SysVinit*, происходила от System V — ранней версии Unix, и как таковая является практически мистической подборкой скриптов, собранных вместе седобородой магией. Все это довольно неплохо работало, но по мере развития дистрибутивов Linux начало давать сбои. *SysVinit* определила шесть уровней выполнения, которые дистрибутивы игнорировали либо употребляли неправильно, а зависимости и приоритеты сервисов были исключительно неудобны в работе. И вот в 2006 г. Canonical приступила к разработке замены, известной как *Upstart*. *Upstart* имел обратную совместимость с *SysVInit*, но предлагал куда лучшую работу с зависимостями и позволял выполнять задачи и реагировать асинхронно. Помимо Ubuntu, *Upstart* приняли все дистрибутивы Red Hat, а также Chrome OS. Но к 2013 г. основные дистрибутивы пошли путем *Systemd*. В 2014 г. Технический комитет Debian проголосовал за переход на *Systemd* в противовес *Upstart*, за чем последовали разборки с Ubuntu. В некотором смысле, это был последний гвоздь в крышку гроба *Upstart*, по крайней мере, в Linux (*Systemd* не поддерживает другие ядра, как BSDs или Hurd, что является предметом споров).



› Devuan — ответвление Debian без *Systemd*. Оно пока на стадии пре-альфа, и если вам подавай дистрибутив без *Systemd*, то уж лучше Slackware, PCLinux OS или Gentoo.

Места и сессии

Одна из причин столь широкого внедрения *Systemd* — его унифицкация сконцентрированных на рабочем столе функций. Его компонент *logind* (помимо принятия на себя старой функции обслуживания логинов) формализует концепцию мест, сессий и пользователей так, что — при наличии соответствующего оборудования — управление одновременно протекающими сессиями рабочего стола становится тривиальной задачей. Хотя пользоваться этим будет далеко не каждый, побочный эффект в том, что теперь старая логика ConsoleKit абсолютно устарела. Раньше любому, кто не использовал весь рабочий стол целиком, приходилось бороться с этим кошмаром, чтобы смонтировать устройства USB или выключить систему, не имея привилегий root, что вызывало множество недовольных постов на множестве форумов.

Systemd-logind также позволяет запускать X-сервер от имени пользователя, что повышает безопасность. И наоборот, среды рабочего стола, в частности, Gnome, начали использовать компоненты *Systemd* (не саму систему init — она здесь ни при чем), что вызвало определенное недовольство, поскольку установка этих компонентов (или их использование без системы инициализации *Systemd*) может быть весьма непростым. Команды **reboot**, **halt**, **shutdown** требуют доступа root, однако *systemd-logind* (вместе с пакетом *polkit*) позволяют выполнять эти функции любому локально вошедшему в систему пользователю с активной сессией X. Такой пользователь сможет выключить компьютер с помощью

```
$ systemctl poweroff
```

— конечно, при условии, что в системе нет других пользователей; а если есть, то пользователю предложат ввести пароль root. Можно также заменить **poweroff** на **suspend** или **hibernate**, при условии, что оборудование их поддерживает [более подробную информацию по проблемам с режимом ожидания вы найдете в Учебниках, стр. 80 **LXF196**]. *Systemd-logind* также обрабатывает события от кнопки питания и спящего режима, что всегда было функцией *acpid*. Это настраивается в файле **/etc/systemd/logind.conf**, который предлагает следующие вполне понятные настройки по умолчанию:

```
IdleAction=ignore
HandlePowerKey=poweroff
HandleSuspendKey=suspend
HandleHibernateKey=hibernate
HandleLidSwitch=suspend
HandleLidSwitchDocked=ignore
```

Внутренний Журнал

Исчез также и старый добрый сервис syslog (ну, по большей части *Systemd* может перенаправлять сообщения на демон syslog, если это необходимо). Демона журнала *Systemd* будет более чем достаточно для обработки требований лога Пользователя Васи. До *journald* сообщения собирались из ядра и изо всех работающих (или не работающих) сервисов демоном syslog, а тот отфильтровывал эти сообщения в текстовые »

Жизнь без Systemd

Некоторые дистрибутивы, хоть и используют *Systemd* по умолчанию, позволяют использовать альтернативную систему init, если вы этого хотите. Поддержка для них разная. Например, Ubuntu 15.04 делает этот процесс очень простым: и *Systemd*, и *Upstart* устанавливаются по умолчанию, и вы увидите пункт 'Ubuntu … (upstart)' в подменю *Grub* Advanced options for Ubuntu. Те, кому нужно переключение на постоянной основе, могут установить пакет *upstart-sysv* и запустить

```
$ sudo update-initramfs –u
```

На данный момент у большинства пользователей Ubuntu не будет трудностей с двумя системами (а многие, вероятно, даже не заметят разницы). Однако в будущем это изменится, особенно после выхода в следующем году LTS, когда уляжется пыль и *Systemd* встроится в экосистему Ubuntu.

Было бы упущением не упомянуть еще одну систему init: *OpenRC*. Хотя технически она и не заменяет *SysVinit*, но все же расширяет и модернизирует все, что происходит после PID1. *OpenRC* поддерживается — и используется по умолчанию — в Gentoo, где до 2007 г. применялась исключительно оболочка, очень корявое решение. После того, как *udev* вошел в *Systemd*, упрямцам приходится использовать *eudev*, еще одну махинацию Gentoo, которая ответвилась от *udev* до его ассимиляции. Но не беспокойтесь, вы можете использовать и *OpenRC*, и *eudev* и в других дистрибутивах тоже: в Arch Linux имеются пакеты в AUR. Некоторые привычные пакеты (например, *X.org*) используют библиотеки *Systemd*, так что вам не удастся полностью отделаться от этого монстра.

файлы в **/var/log**. Процессы области пользователя также будут размещать здесь свои логи напрямую. Чтобы эта директория не стала чудовищно огромной, стоит установить и настроить *logrotate*. С *Systemd* все логи централизованы, и доступ к ним можно получить через команду **journalctl**. Конечно, если вам все равно нужна реализация syslog, ее можно запустить в тандеме с *journald*, однако большинство обходится без этого. Исполнение **journalctl** покажет вам все логи, которые эта команда сможет вспомнить. Они автоматически пропускаются через **less** для облегчения просмотра. По умолчанию более старые логи не удаляются, пока место на диске не превысит указанного в файле **/etc/systemd/journald.conf**. Здесь есть три опции, которые вы, возможно, решите настроить:

» **SystemMaxUse** Указывает максимальное место на диске, отведенное журналу; по умолчанию это 10% от файловой системы, хранящей журнал.

» **SystemKeepFree** Минимальное место, которое *Systemd* попытается оставить свободным от логов файловой системы. Если установить его больше, чем имеющееся место, эта величина будет исправлена на размер свободного места на момент запуска *Systemd*.

» **SystemMaxFileSize** Максимальный размер каждого отдельного файла журнала. В конечном итоге он сообщает *Systemd*, на какое количество файлов разбивать логи, чтобы часть предыстории при ротации терялась.

История — это, конечно, хорошо, но если вам нужно просмотреть только сегодняшние логи, то с параметром **-b** выведутся только сообщения из текущей загрузки. Если что-то не работает, то инстинктивной реакцией линуксоида будет проверить вывод

```
$ dmesg | tail
```

на предмет сообщений об ошибках из ядра, или

```
$ tail /var/log/messages
```

на предмет сообщений изо всех других мест. Эквивалент для *Systemd* — запуск

```
$ journalctl -e
```

что позволит просмотреть информацию с конца журнала. Естественно, *dmesg* по-прежнему работает, но таким образом мы видим также сообщения из источников помимо ядра, и автоматически отображаются временны́е отметки по местному времени, а не в секундах после загрузки системы. Если в предыдущей загрузке что-то пошло не так, можно проверить эти логи, добавив к параметру **-b** число. Добавление **-1** относится к текущей загрузке (для **-b** по умолчанию), **-2** — к предыдущей загрузке, и т. д. Здесь также можно использовать абсолютную индексацию, когда **1** относится к самой ранней загрузке в логах *Systemd* (точно так же, как при вызове без опции **-b**), **2** — к следующей, и т. д.

```
$ systemctl list-unit-files
```

отобразит полный их список и покажет их статусы. Все файлы юнитов размещаются в поддиректории **system/** или **user/** основной директории *Systemd* (обычно **usr/lib/systemd/**). Файлы юнитов могут быть сервисами (например, sshd.service),

```
httpd.service                                  disabled
initrd-cleanup.service                         static
initrd-parse-etc.service                       static
initrd-switch-root.service                     static
initrd-udevadm-cleanup-db.service              static
ip6tables.service                              disabled
ipset.service                                  disabled
iptables.service                               disabled
kmod-static-nodes.service                      static
krb5-kadmind.service                           disabled
krb5-kdc.service                               disabled
krb5-kpropd.service                            disabled
krb5-kpropd@.service                           static
ldconfig.service                               static
libvirt-guests.service                         disabled
libvirtd.service                               disabled
lircd.service                                  disabled
lircmd.service                                 disabled
lm_sensors.service                             disabled
logrotate.service                              static
lvm2-lvmetad.service                           disabled
lvm2-monitor.service                           disabled
lvm2-pvscan@.service                           static
man-db.service                                 static
lines 38-81
jonni@jbmachine:~
```

› **Файлы юнитов — повсюду. Они — источник жизненной силы *Systemd* и, разумеется, вашего компьютера.**

Споры о двоичности

Логи Systemd хранятся в двоичном формате, для простоты индексации. Это позволяет мигом искать в большом количестве данных, но и является своего рода камнем преткновения. Двоичные логи более уязвимы, так что в теории отказ диска может повредить всего лишь сектор текстового файла размером 4к, а двоичному *journald* тут придет и конец.

Текстовые файлы допускают парсинг с помощью Perl, *grep*, *sed*, awk и т. д., и такими скриптами пользуются многие сисадмины, работая с логфайлами. То, что скрипты станут непригодны, похоже, вызвало серьезное недовольство у некоторых сисадминов, однако мы полагаем, что подобная критика неуместна: если вам нужны текстовые файлы, то более новые версии *syslog-ng* совершенно свободно вытащат их из *journald*.

Наиболее фундаментальные компоненты *Systemd* — это изобретательно названные файлы юнитов [unit-files]. Команда

Systemd — что здесь может не нравиться?

Самая громогласная жалоба на *Systemd* относится к его предполагаемому нарушению традиционной философии Unix, заключающейся в наличии одного инструмента, который хорошо делает нечто одно и при этом ладит с другими инструментами, которые, в свою очередь, тоже что-то делают хорошо.

Systemd обвиняют в том, что это — монолитный блок, который узурпирует (помимо всего прочего) *udev*, *cron*, PAM, *acpid* и *logind*. Наличие всех этих компонентов в одном бинарнике, работающем как PID1, кое-кого огорчает, однако по большей части весь этот плач вызван неосведомленностью. Тот факт, что *Systemd* столь широко используется, должен свидетельствовать в пользу его уместности и пригодности, однако вместо этого противники заявляют о заговоре, даже о навязывании разработчиками пользователям своих предпочтений [do-ocracy].

В защиту *Systemd*, он предлагает все виды современных функций: равномерное распределение ресурсов по cgroups ядра, удаленный доступ к логам, значительно улучшенную среду chroot (через *systemd-nspawn* и *machinectl*) и более быструю загрузку, и это лишь малая часть. Попытка понять процесс загрузки всегда страшит новичка, но в *Systemd*, по крайней мере, эта проблема проще благодаря тому, что компоненты четко разделены и используется современный синтаксис: полная противоположность кошмару в стиле Лавкрафта, с которым вам приходилось сталкиваться во время о́но.

Конечно, *Systemd* пока относительно юн, и некоторые будущие функции, о которых поговаривают, вызывают обеспокоенность: нам и вправду надо объединить PID1 с его собственным загрузчиком? Вы хотите работать с системой без состояний (без файлов статической настройки)?..

Ну что ж, поживем — увидим, что из этого дальше выйдет.

которые запускают программы, демоны и т. д., или более абстрактными вещами, такими, как точки монтирования, сокеты, устройства или цели. Цели являются более гибкой интерпретацией runlevels из *SysV*, они определяют набор сервисов для определенной задачи. Системы рабочего стола по умолчанию грузятся в цель **graphical**, что практически является runlevel 5, поскольку (надо надеяться) имеет результатом графический логин, такой, как Gnome GDM или легковесный SDDM. Серверы будут грузиться в **multi-user.target**, соответствующую runlevel 3, загружающему логин консоли. Если изучить файл **graphical.target**, то, помимо всего прочего, там обнаружатся такие строки:

```
Requires=multi-user.target
Wants=display-manager.service
```

Это сообщает нам, что наша цель **graphical** заключает все в **multi-user target**, но кроме того, ей нужно загрузить менеджер дисплея. Систему можно втиснуть в определенную цель (но только с привилегиями root), используя, например,

```
$ systemctl isolate multi-user.target
```

Файл **display-manager.service** на самом деле является символической ссылкой, которая настраивается при установке менеджера дисплея; она указывает на файл сервиса. Сервисы добавляются в цели *Systemd* с помощью команды **systemctl enable**, которая просто создает необходимые символьные ссылки. Например, чтобы запустить демон SSH при следующей загрузке, скомандуйте

```
$ systemctl enable sshd
```

и вам сообщат о действиях *Systemd*:

```
Created symlink from /etc/systemd/system/multi-user.target.wants/sshd.service to /usr/lib/systemd/system/sshd.service.
```

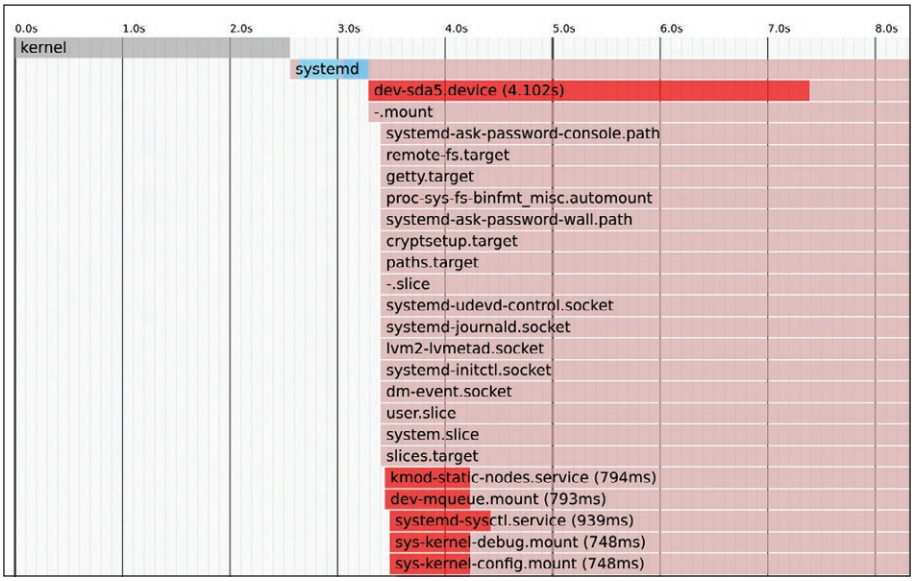
Если что-то пошло не так

Время от времени что-то неизбежно ломается [*Ред.*: — Это, кажется, второй закон термодинамики]. Иногда поломка приводит к тому, что система перестает загружаться, а с такими системами никто работать не любит. Обычно проблему создают графические драйверы, и когда системе не удается запустить менеджер графического входа в систему, она просто застревает на этом месте, беспомощная и немая. Чтобы это исправить, надо просто перезагрузиться (и, будем надеяться, машина позволит вам сделать это без проблем) и добавить следующую опцию в командную строку ядра (нажмите e, чтобы отредактировать ее в меню *Grub*):

```
systemd.unit=multi-user.target
```

Загрузка с этой опцией (при нажатии Ctrl+X) не позволит загрузиться менеджеру дисплея, если в нем есть ошибки, чтобы проблемы с драйверами могли быть исправлены (будем надеяться) из командной строки. При более серьезных проблемах с загрузкой вам, возможно, придется обратиться к целям **rescue** или **emergency**, или, в крайнем случае, использовать chroot из другой ОС.

Конечно, не все проблемы приводят к отказу системы загружаться. Симптомами могут служить странные сообщения об ошибках, которые



› Вряд ли кто-то любит хорошие графики больше нашего, а особенно с подробной информацией о процессе загрузки, полученной от *systemd-analyze*.

слишком быстро пропадают, чтобы их можно было прочитать, а иногда возникает 90-секундная пауза перед тем, как загрузка продолжится. Помимо журнала, вы можете получить общее представление о состоянии системы по команде

```
$ systemctl status
```

которая покажет все стоящие в очереди задачи и перечислит все запущенные файлы сервисов и процессы (опять же пропущенные через **less** для удобства просмотра). Если вторая строка такая:

```
# State: degraded
```

(где degraded [разрушен] окрашено в сеющий панику красный цвет), значит, что-то пошло не так. Обычно в силу неких причин отказывается загружаться файл юнита. Можно выяснить более подробно с помощью

```
$ systemctl --state=failed
```

Обнаружив негодный юнит, можно воспользоваться **journalctl**, чтобы посмотреть, оставил ли он хоть какую-то полезную информацию в журнале; например, если вышеуказанная команда сообщила о проблемах с sshd.service, можно запросить все, что он недавно писал в журнале, командой

```
$ journalctl -eu sshd
```

Она должна предоставить достаточно информации для диагностики и решения проблемы. Перезапустите сервис с помощью

```
$ systemctl restart sshd
```

и, будем надеяться, все будет хорошо, в каковом случае статус *Systemd* изменится с тревожного 'degraded' на обычный 'running [работает]'. Некоторые процессы области пользователя тоже делают записи в журнале, и мы тоже можем их отфильтровать по имени процесса (с помощью опции **_COMM=**), абсолютному пути или PID (**_PID=**). Начиная с Gnome 3.12, логи *X.org* больше не записываются в часто изучаемый (а ныне часто искомый) файл **/var/log/Xorg.0.log**. Вместо этого они теперь размещаются в журнале, который можно фильтровать с помощью

```
$ journalctl -e _COMM=Xorg
```

или

```
$ journalctl -e /usr/bin/Xorg
```

Если вы используете Gnome в Fedora или Arch Linux, в аргументе _COMM, который мы упоминали выше, придется использовать **Xorg.bin** или **gdm-x-session**.

> «Фундаментальные компоненты Systemd — изобретательно названные юниты.»

Ускорение загрузки

Одно из особенно приятных свойств *Systemd* — его способность анализировать время загрузки. Команда

```
$ systemd-analyze
```

выведет отчет о том, сколько драгоценного времени загрузки занято процессами ядра и области пользователя. Чтобы получить более подробную информацию, добавьте к команде **blame** — будет показано время, затраченное на каждый сервис. В итоге вы получите список, где сначала будут перечислены процессы, занимающее самое большое время, однако помните, что, поскольку всё, используя язык *Systemd*, «агрессивно запараллелено», время, указанное здесь, может быть намного больше, чем время на переход из *Grub* в окно/экран приглашения. И в качестве финального аккорда, вы даже можете сделать симпатичный график SVG, отображающий всю замечательную информацию по времени, с помощью

```
$ systemd-analyze plot > plot.svg
```

Прочитав наше руководство, вы поймете, что *Systemd* не очень и жуткая перспектива и, возможно, не такой уж злодей в склонном побрюзжать мире сисадминов. LXF

По советам м-ра Брауна

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

Эзотерическое системное администрирование из затейливых закоулков серверной.

Леопард меняет пятна

Недавно на конференции ChefConf в США Марк Руссинович [Mark Russinovich], уважаемый член команды инженеров Microsoft, спросил в собрании экспертов по открытому ПО, кто из присутствующих использует для разработки исключительно Windows — и руку поднял только один смельчак. На собрании также поинтересовались, появится ли когда-нибудь открытая Windows, и согласно Руссиновичу, «все обсуждения этого вопроса, которые только можно себе представить, уже проводились». Хотя я думаю, что мы вряд ли скоро увидим Windows на диске **LXF**, стоит сказать, что обновленная Microsoft, возглавляемая Сатьей Наделлой [Satya Nadella], кажется, принимает новый подход к открытому ПО в целом: в облаке Azure есть варианты для Linux; .NET можно загрузить с GitHub; также анонсирована кроссплатформенная поддержка Visual Code Studio (см. стр. 16).

Так было не всегда: каждый, кто помнит Microsoft в 1990-х и начале нулевых, вспомнит такие истории, как утечку документов «Хэллоуин» (Halloween — внутренние директивы Microsoft с планами по атаке на Linux) и Стива Балмера [Steve Ballmer], который обзывал Linux и лицензию GNU «коммунизмом» и «раковой опухолью». Тогда Microsoft была огромной 500-фунтовой гориллой — кое-кто сказал бы, «гориллой с нечестной конкуренцией», ибо она рвалась смести всё, что было на настольных ПК до нее (включая старый добрый *Netscape*), и была известна своей философией «принять, расширить, уничтожить».

Сейчас все иначе, и открытое ПО явно сыграло свою роль в изменении пейзажа, хотя бы по части Microsoft. Любой администратор Linux, которому нужно интегрироваться с Windows, с радостью примет эти изменения и будет надеяться, что компания продолжит работу в том же русле.

Был бы еще в Windows приличный SSH-сервер...

jolyon.brown@gmail.com.

Новые релизы OpenStack и Ubuntu Snappy

OpenStack поднимается до версии 11, а Ubuntu предлагает новую утилиту для управления контейнерами.

Проект *OpenStack* продолжает впечатляющее движение по графику выпуска релизов с новым релизом Kilo. Это одиннадцатая итерация платформы. (Данный релиз посвящен памяти Криса Йо [Chris Yeoh], известного популяризатора *OpenStack*, который, к сожалению, ушел из жизни в апреле этого года.)

В этом релизе появилось около 400 новых функций и было исправлено несколько тысяч ошибок во всей платформе (которая всё разрастается). Основные его цели — повышение стабильности и развитие компонента для развертывания на «голом железе» (внутри проекта он имеет соответствующее название “Ironic [железный *или* иронический]”), отныне готового к повсеместному применению. Он уже используется публичными провайдерами облачных сервисов. Ironic также предоставляет поддержку существующих конфигураций виртуальных машин и развивающихся технологий: контейнеров Linux, «платформы как сервис» и NFV (виртуализация сетевых функций — виртуализированные балансировщики нагрузки и т.п.). Ubuntu дополнила Kilo своим релизом 15.04 (Vivid Vervet). Он включает предварительную версию *LXD*, нового гипервизора контейнеров, который, как утверждает *OpenStack*, предоставляет «родную» поддержку, а также все утилиты управления и безопасности, необходимые для эффективного запуска подходящих конфигураций. В релиз включен драйвер *OpenStack*, и хотя для повсеместного использования продукт пока не готов, Ubuntu обещает включить его в следующий дистрибутив. В 15.04 также вошел первый релиз Ubuntu Core под кодовым названием “Snappy” для облачных хостов (хотя технология, вероятно, задумана для таких устройств, как смартфоны). Snappy, самая маленькая версия ОС от Canonical, использует транзакционные обновления — которые все шире используются в разных ОС и некоторое время находились в бета-версии. Этот способ предоставления обновлений и заплат к ОС сильно отличается от традиционного и работает в обход обычных утилит вроде *apt-get*, обещая откаты обновлений и установок одной командой. На систему повлияла модель контейнеров, где ОС и приложения предоставляются в виде образов и обновления представляют собой только изменения, применяемые поверх образов. Систему можно опробовать на самых разных платформах, от Raspberry Pi до облачных провайдеров.

› Уже получше, правда? Это не десятка. Видите ли, большинство сисадминов будут разворачивать десятку. Куда мы сможем двинуться дальше? Куда?

CoreOS: Инфраструктура нового поколения

Во второй части рассказа о чудесах CoreOS изучим обнаружение сервисов с *etcd* и кластеризацию с *fleet*.

Месяц назад я начал рассказывать о CoreOS, дистрибутиве Linux, который использует контейнеры для управления сервисами и обеспечивает автоматические обновления. Я описал настройку тестовой системы на основе *Vagrant*, исследовал работу с патчами и выполнил несколько базовых команд *fleet* (*fleet* — утилита управления кластерами CoreOS). Во второй части я продолжу тестирование CoreOS для настройки инфраструктуры клиента. Это поможет получить некоторое представление о том, как различные компоненты взаимодействуют друг с другом. Если вы не читали первую часть [см. «Советы», стр. 54, **LXF198**], стоит взглянуть на нее, если вы будете выполнять приведенные здесь примеры. Я привел краткое описание команд для настройки и запуска системы [см. врезку «Настройка CoreOS в *Vagrant*» внизу].

В конце моего последнего эксперимента у меня было три экземпляра CoreOS для предоставления сервисов кластеризации. Я хочу попробовать реализовать типовой сценарий, который может понадобиться клиенту, и к концу этой статьи получить отказоустойчивый сервис хостинга. Двинемся дальше: первое, что я хочу сделать — добавить экземпляр CoreOS для демонстрации того, как *etcd* обрабатывает выход узлов из строя. Для эффективной работы этой схемы нам нужны три узла, поэтому обязательно надо добавить четвертый.

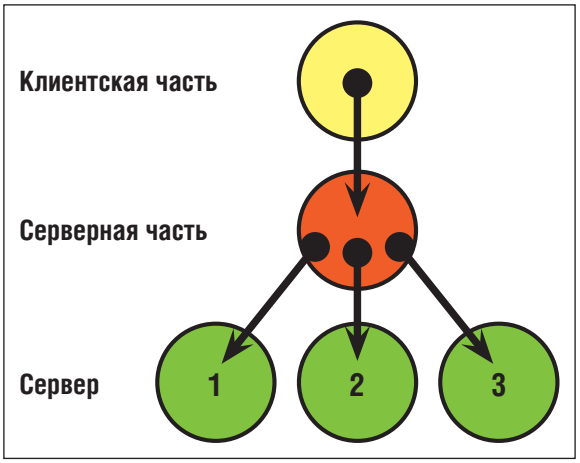
Предположим, что в начале эксперимента кластер выключен (вряд ли кто-то оставил его включенным на целый месяц с прошлого урока). Если это так, то первое, что нужно сделать — повторно зарегистрировать сервис публичного обнаружения (на https://discovery.etcd.io/new). Как и в прошлый раз, следует поместить значение, возвращаемое сервисом, в файл пользовательских данных, который находится в том же каталоге, что и Vagrantfile, обновив строку, начинающуюся с “discovery:” — поместить в нее полный URL, предоставляемый сервисом. Если вы следовали моему прошлому уроку, то я обнаружил, что существует образ *Vagrant* для CoreOS. Если это так, *Vagrant* спросит об этом. Команда

```
$ vagrant box update
```

обновит систему, загрузив последний образ (в моем случае до версии 647.0.0). Затем нужно изменить файл **config.rb**, изменив значение параметра “num_instances” с трех на четыре.

После этого кластер запускается простой командой

```
$ vagrant up
```



› ***Vulcand*** **направляет подключения на соответствующие серверы, сравнивая путь в URL запроса с информацией, предоставленной *etcd*.**

Все четыре узла загрузятся довольно быстро. Теперь я подключусь по SSH к первому узлу —

```
$ vagrant ssh core- 01
```

и смогу быстро подтвердить состояние кластера командой

```
$ fleet list-machines :
MACHINE IP METADATA
462cf0c6... 172.17.8.102 –
8841d0cb... 172.17.8.101 -
af2fde58... 172.17.8.104 -
cd8df6e3... 172.17.8.103 –
```

Vulcand: живи с балансом нагрузки

Я разверну с помощью *fleet* экземпляр *vulcand* (www.vulcanproxy.com) — прокси для микросервисов и API управления, хотя здесь для своих целей я использую ее более традиционным образом. Умная черта этой программы, от компании под названием Mailgun (и с открытым исходным кодом), в том, что для конфигурации сервера она использует *etcd*, а не более традиционную настройку через файлы, применяемую, например, *HAProxy*. *HAProxy* можно заставить сделать нечто похожее с помощью сторонних расширений, хотя для этого и потребуется чуть больше усилий. Я хочу, чтобы *vulcand* автоматически регистрировал экземпляры приложений, которые *fleet* будет также развертывать, и направлял на них трафик без перезапусков сервисов и перезаписи файлов »

Настройка CoreOS в Vagrant

Если вы пропустили прошлый номер [его можно добыть на стр. 108], вам будет трудно следовать этому уроку. Вот ураганный набор инструкций, которые вас ускорят. Сперва загрузите последнюю версию *Vagrant* с www.vagrantup.com (она немного свежее версии в моем Ubuntu 14.04). Я также установил *VirtualBox*: он довольно удобен с *Vagrant* и *Git* для работы с репозиториями. Сделав это, скомандуйте

```
$ git clone https://github.com/coreos/coreosvagrant.
git
$ cd coreos-vagrant
$ mv user-data.sample user-data
$ mv config.rb.sample config.rb
```

Получите токен, зайдя на https://discovery.etcd.io/new. Раскомментируйте строку “discovery” в **user-data** и замените строку https://discover.etcd.io строкой из окна браузера. Затем откройте файл **config.rb** и раскомментируйте и обновите параметры num_instances и update_channel так:

```
$num_instances=3
$update_channel='stable'
```

Ну и запустите свой новый кластер CoreOS. Загрузит образ CoreOS Vagrant и запустит его команда

```
$ vagrant up
```

настройки. Все они будут запускаться в контейнерах *Docker*, и *fleet* позаботится об отказоустойчивости, вновь разворачивая всё необходимое в кластере, если я перезапущу любой из родительских хостов. Возможно, и, наверное, желательно, управлять *fleet* с локального клиента (т. е. с любого компьютера, используемого для рабочего стола) и протуннелировать его через SSH к системам CoreOS. В рабочем окружении я бы сделал именно так (употребив какой-либо «промежуточный» хост в защищенном сегменте сети, для доступа к которому нужна двухфакторная аутентификация). Но так как это всего лишь тест и я ограничен в пространстве и времени, я буду управлять *fleet* с первого узла.

Первым делом надо поместить *vulcand* на свой кластер, поэтому я создам подкаталог под названием **units**. В этом каталоге я создам файл **vulcand.service** со следующим содержимым:

```
[Unit]
Description=Vulcan
After=docker.service
[Service]
Restart=always
TimeoutStartSec=0
ExecStartPre=-/usr/bin/docker kill vulcan1
ExecStartPre=-/usr/bin/docker rm vulcan1
ExecStartPre=/usr/bin/docker pull mailgun/vulcand:v0.8.0-beta.2
ExecStartPre=/usr/bin/sudo /usr/bin/ip addr add 172.17.8.100/24
dev eth1
ExecStartPre=/usr/bin/etcdctl set /vulcand/backends/lxf/backend
'{«Type»: "http»}'
ExecStartPre=/usr/bin/etcdctl set /vulcand/frontends/f1/frontend
'{«Type»: "http", "BackendId»: "lxf", "Route»: "Path(`/`)»}'
ExecStart=/usr/bin/docker run --rm --name vulcan1 -p 80:80
-p 8182:8182 mailgun/vulcand:v0.8.0-beta.2 /go/bin/vulcand
-apiInterface=0.0.0.0 -interface=0.0.0.0 -etcd=http://10.1.42.1:4001
-port=80 -apiPort=8182
ExecStopPost=-/usr/bin/sudo /usr/bin/ip addr del 172.17.8.100/24
dev eth1
ExecStop=-/usr/bin/docker stop vulcan1
```

Как и в первой статье, этот файл модуля *Systemd* определяет, как будет запускаться мой сервис *vulcand*. Рассмотрим этот файл повнимательнее. Первые несколько строк задают имя модуля и велят *Systemd* запускать модуль только тогда, когда сервис *Docker* настроен и запущен. Перезапуск (**restart**) говорит *Systemd*, что при возникновении неожиданного сбоя мне надо перезапустить сервис (учтите, что при корректном выключении этого не произойдет). Строки ExecStartPre немного прибираются после предыдущих экземпляров и связывают плавающий IP-адрес сервера с интерфейсом eth1 компьютера, на который *fleet* разворачивает сервис (в разделе stop он удаляется). Благодаря этому я всегда могу связаться с *vulcand* по одному и тому же IP-адресу. Дефис в начале строк docker kill и rm гарантируют, что если рассматриваемого экземпляра *Docker* не существует, то сообщений об ошибок не появится. Это вызвало бы у *Systemd* проблему с нашим модулем.

Строки edcdctl добавляют клиентское и серверное определения для *etcd* (примечание: необходимо указать абсолютный путь к двоичным файлам, и строки без него будут проигнорированы).

Строка ExecStart — та самая колбаса на бутерброде *Systemd*: она запускает *vulcand* в контейнере *Docker* и открывает порты. Я также сообщаю ей, на каком адресе и номере порта она может найти *etcd* — их покажет команда

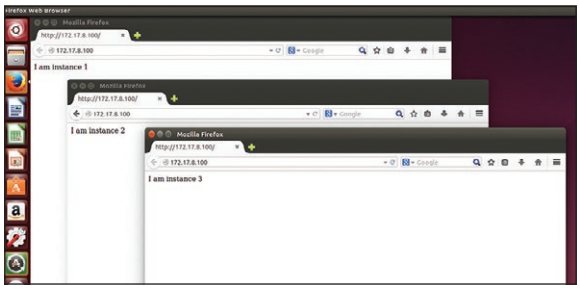
```
$ ip addr show
```

Для контейнера это будет адресом интерфейса docker0. Так как мы пользуемся *Vagrant*, этот адрес может отличаться от адреса, используемого в более традиционной CoreOS. В файле **Vagrantfile** должен быть указан верный адрес, используемый в модуле (выше). Измените его, если в вашей системе он отличается.

Без дальнейших предисловий я могу запустить *vulcand* на своем кластере и проверить ее статус. *Fleet* назначит узел, на котором она будет запущена — об этом просто не надо беспокоиться. Будь это рабочая система, у меня было бы несколько групп систем CoreOS, расположенных в различных сетевых сегментах по соображениям безопасности; но здесь я только демонстрирую функциональность.

```
$ fleetctl start vulcand.service
Unit vulcand.service launched on 8841d0cb.../172.17.8.101
[Сервис запущен...]
$ fleetctl list-units
UNIT MACHINE ACTIVE SUB
vulcand.service 8841d0cb.../172.17.8.101 active running
[сервис активен и работает]
```

Пока все выглядит неплохо. Я могут открыть http://172.17.8.100 (виртуальный IP-адрес, назначенный ранее) и получить сообщение об ошибке {«ошибка»: «не найдено»}. Оно появляется потому, что мне еще негде распределять нагрузку... пока.



› Я распределил трафик по нескольким серверам отказоустойчивым способом; осталось только разжиться приложением, которое что-то делает...

Шёлк снаружи, щёлк внутри

Vulcand похож на многие другие системы балансировки нагрузки в том, что для его работы требуется определить клиентские

Контейнерные войны и фланель

К счастью или к несчастью, во время написания этой статьи CoreOS проводила свой CoreOS Fest, на котором было сделано несколько объявлений, касающихся, в частности, собственной спецификации контейнеров, конкурирующей с *Docker*. Спецификация приложений контейнера appc (реализацией которой в CoreOS является rkt) теперь имеет поддержку в лице нескольких компаний, не аффилиированных с CoreOS — представители Google, Twitter и Red Hat теперь принимают участие в будущем спецификации. Google реализовала rkt в *Kubernetes* и *VMware*, входящих в проект Photon, облегченную реализацию Linux. Apcera, компания, нацеленная на пользователей «гибридных облаков» уровня предприятия, написала (и открыла) собственную реализацию известную под названием *Kurma*. Кажется, что битва начинается, хотя производители перестраховываются, поддерживая *Docker* наряду с новичком.

CoreOS также сообщила некоторые подробности о своих сетевых планах. На данный момент для загрузки доступна "flannel" — реализация модели *Kubernetes*, в которой каждому компьютеру кластера назначается полная подсеть. Это удобно для снижения сложности перенаправления портов, но из облачных провайдеров на данный момент такое может делать только Google. Flannel использует универсальное устройство TUN/TAP и создает оверлейную сеть с UDP для инкапсуляции IP-пакетов. Ее можно загрузить с GitHub, и она выглядит интересно; зайдите на https://github.com/coreos/flannel — там вы также найдете несколько полезных схем.

и серверные сервисы. Круто здесь то, что вместо перекапывания файлов настройки значения будут записаны в *etcd*. Более того, все это будет сделано автоматически. Это означает, что предварительно придется поработать чуть больше; но труды окупятся. В файле **vulcand** мы уже задали два ключа, которые задают сервер под названием lxf типа HTTP и клиент (под названием f1), с тем, чтобы весь трафик (определяемый контекстным путем **/**) отправлять на сервер lxf. Будь у меня несколько приложений, я мог бы задать различные серверы, определяемые различными контекстными путями (например, **/payments/** или **/login/**). Оставаясь в каталоге **units**, я создам еще три файла — **hosta.service**, **hostb.service** и **hostc.service** [их также можно найти на http://pastebin.com/jPiM5adb]:

Файл **hosta.service** выглядит так:

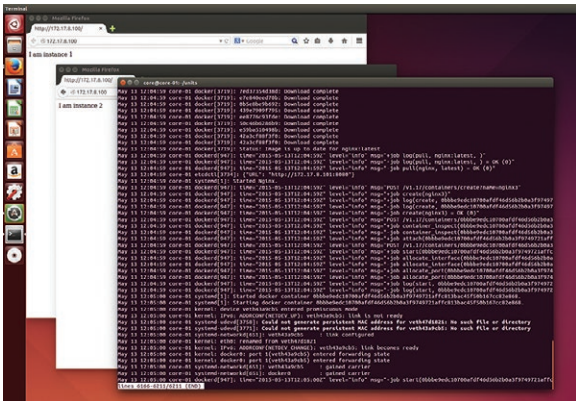
```
[Unit]
Description=Nginx
After=docker.service
[Service]
EnvironmentFile=/etc/environment
TimeoutStartSec=0
ExecStartPre=-/usr/bin/docker kill nginx1
ExecStartPre=-/usr/bin/docker rm nginx1
ExecStartPre=/usr/bin/docker pull nginx
ExecStartPre=/bin/sh -c “mkdir -p /home/core/www; echo ‘Я экземпляр 1’ > /home/core/www/index.html”
ExecStartPre=/usr/bin/etcdctl set /vulcand/backends/lxf/servers/srv1 ‘{«URL»: “http://${COREOS_PUBLIC_IPV4}:8080»}’
ExecStart=/usr/bin/docker run --name nginx1 -p 8080:80 -v /home/core/www:/usr/share/nginx/html nginx
ExecStopPost=/usr/bin/etcdctl rm /vulcand/backends/lxf/servers/srv1
ExecStop=-/usr/bin/docker stop nginx1
[X-Fleet]
Conflicts=host*.service
```

А вот файл **hostb.service**:

```
hostb.service
[Unit]
Description=Nginx
After=docker.service
[Service]
EnvironmentFile=/etc/environment
TimeoutStartSec=0
ExecStartPre=-/usr/bin/docker kill nginx2
ExecStartPre=-/usr/bin/docker rm nginx2
ExecStartPre=/usr/bin/docker pull nginx
ExecStartPre=/bin/sh -c “mkdir -p /home/core/www; echo ‘Я экземпляр 2’ > /home/core/www/index.html”
ExecStartPre=/usr/bin/etcdctl set /vulcand/backends/lxf/servers/srv2 ‘{«URL»: “http://${COREOS_PUBLIC_IPV4}:8080»}’
ExecStart=/usr/bin/docker run --name nginx2 -p 8080:80 -v /home/core/www:/usr/share/nginx/html nginx
ExecStopPost=/usr/bin/etcdctl rm /vulcand/backends/lxf/servers/srv2
ExecStop=-/usr/bin/docker stop nginx2
[X-Fleet]
Conflicts=host*.service
```

И, наконец, файл **hostc.service**:

```
hostc.service
[Unit]
Description=Nginx
After=docker.service
[Service]
EnvironmentFile=/etc/environment
TimeoutStartSec=0
```



› При решении проблем с *fleet* знание того, как опросить лог-файлы *Systemd* командой journalctl, будет бесценным.

```
ExecStartPre=-/usr/bin/docker kill nginx3
ExecStartPre=-/usr/bin/docker rm nginx3
ExecStartPre=/usr/bin/docker pull nginx
ExecStartPre=/bin/sh -c “mkdir -p /home/core/www; echo ‘Я экземпляр 3’ > /home/core/www/index.html”
ExecStartPre=/usr/bin/etcdctl set /vulcand/backends/lxf/servers/srv3 ‘{«URL»: “http://${COREOS_PUBLIC_IPV4}:8080»}’
ExecStart=/usr/bin/docker run --name nginx3 -p 8080:80 -v /home/core/www:/usr/share/nginx/html nginx
ExecStopPost=/usr/bin/etcdctl rm /vulcand/backends/lxf/servers/srv3
ExecStop=-/usr/bin/docker stop nginx3
[X-Fleet]
Conflicts=host*.service
```

У этих файлов много общего с файлом для *vulcand*, но есть пара дополнительных строк, о которых стоит поговорить. Я задал для *Systemd* параметр EnvironmentFile, который позволяет использовать переменную COREOS_PUBLIC_IPV4 позднее в модуле. Я использую его в выражении **edcdctl**, добавляя каждый сервер в серверный компонент lxf. Этот модуль запускает экземпляр *Docker* для *Nginx*, и я добавил пустой файл **index.html** в его **webroot**. Последняя строка связана с *fleet* — она сообщает *fleet*, что каждая версия этого модуля не может запускаться на узле, на котором запущена одна из других. Так модули распределяются по кластеру при планировании модулей:

```
$ fleetctl start host*.service
$ fleetctl list-units
UNIT MACHINE ACTIVE SUB
hosta.service af2fde58.../172.17.8.104 active running
hostb.service cd8df6e3.../172.17.8.103 active running
hostc.service 8841d0cb.../172.17.8.101 active running
vulcand.service 8841d0cb.../172.17.8.101 active running
```

Если теперь открыть адрес http://172.17.8.100, возникнет страница, где будет показано, на какой экземпляр *vulcand* я был направлен, и при принудительном обновлении страницы (в обход кэша браузера) каждый раз будет отображаться другой экземпляр. Также можно проверить содержимое кэша *etcd* с помощью **curl**:

```
$ curl -L http://127.0.0.1:4001/v2/keys/vulcand/backends/lxf/servers
```

Команда вернет набор данных JSON, который покажет, что серверы lxf зарегистрированы. Теперь выключим один из них, командой

```
$ fleetctl stop hostc.service
```

и снова выполним команду **curl**. Теперь ключ удален — *vulcand* не будет перенаправлять трафик через этот сервер. Команда

```
$ fleetctl list-units
```

покажет, что остановленный сервер неактивен/«мертв» — это нормально. Аналогично, если принудительно остановить сам *vulcand* (командой **kill -9** или полностью вырубив сервер, на котором он запущен), то *fleet* перезапустит его на другом узле. LXF

Red Hat Enterprise Linux

предоставляет вам **производительность**, **масштабируемость**, **безопасность** и **надежность**, ранее доступные только на очень дорогих платформах

Самая популярная в мире Linux платформа для бизнеса

Обеспечивает высокую производительность, надежность, масштабируемость и безопасность

Сертифицирована ведущими производителями оборудования и разработчиками ПО

Совместима с широким спектром оборудования от рабочих станций до серверов и мэйнфреймов

Обеспечивает одинаковые условия работы приложений при использовании в физической, виртуальной и облачной средах

Пользователи RHEL экономят на оборудовании, лицензиях на программное обеспечение и эксплуатационных расходах

redhat
PREMIER BUSINESS PARTNER

ГНУ/Линуксцентр — Linux-эксперт для вашего бизнеса

— Premier Business Partner компании Red Hat
— 12 специалистов по разработке и внедрению, сертифицированных компанией Red Hat
— Более 100 клиентов, использующих Red Hat
— 10 лет на рынке

Red Hat — ведущий серверный дистрибутив Linux

— Более 15 лет промышленного использования
— Свыше 80% рынка корпоративного Linux по данным CIO Insight
— 5 лет среди лучших вендоров
— Выгодная совокупная стоимость владения (TCO)
— Поддержка в течение 10 лет

Специальное предложение!

Закажите Red Hat Enterprise Linux в ГНУ/Линуксцентре и получите в подарок книгу «Полное руководство пользователя Red Hat Enterprise Linux»

Москва +7 (499) 271-49-54

Санкт-Петербург +7 (812) 309-06-86

Linux-эксперт для вашего бизнеса. www.linuxcenter.ru

Linux center

LINUX FORMAT

Учебники

Наши эксперты помогут вам с любым приложением Linux!

ЕВГЕНИЙ БАЛДИН
Подтвердивший свою квалификацию физик.

Свободные лекции (продолжение)

Кто порочит высшую достоверность математики, питается сумбуром.
Леонардо да Винчи

С год назад сотрудники кафедры теоретической физики физфака Новосибирского университета стали выкладывать видеолекции с магистерскими курсами кафедры в открытый доступ под свободной лицензией CC-BY-SA (атрибутивная, с требованием распространения производных под нею же). Тогда же я, один из участников этого процесса, написал колонку в *Linux Format*. Что же с тех пор изменилось?

Просто выложенные на институтском сервере (энтерпрайз не выдерживал наплыва любопытствующих), малоудобные для просмотра исходники сменились торрентами, включавшими и уже пожатые в *ffmpeg* файлы. Затем возник просто хороший человек, не имеющий отношения к теоретической физике, но взявший на себя общественно-полезную нагрузку и выложивший лекции на YouTube, благо лицензия позволяет. Спасибо ему. Чуть позже уже я освоил YouTube, и сейчас всё, что зафиксировала камера с осени 2013-го по весну 2015 г., доступно, так сказать, для казуального просмотра. Отснято и обработано 11 курсов лекций, 2 семинарских курса и 15 семинаров теоротдела ИЯФ им. Г.И. Будкера. Сделано это непрофессионалами, на энтузиазме, но результат вполне себе имеется и доступен всем желающим. Не знаю, как остальным, но нам это нравится.

PS Ссылки на это богатство можно найти тут: inp.nsk.su/~baldin/PhysLect.

E.M.Baldin@inp.nsk.su

В этом месяце вы научитесь...

Делать хромбуки 64
Нейлу Мору показалось обидным отставать от модного тренда, и в результате его ноутбук охромел. В смысле, на нем теперь стоит Chrome OS от Google.

Вещать аудио 66
Джонни Бидвелл взялся укротить демона *PulseAudio* — оказывается, не так он страшен, как его малюют — и настроил передачу аудиопотока через удаленный клиент.

Анализировать события 68
С присущей ему бдительностью, **Михалис Цукалос** выполняет статическую и динамическую трассировку процессов. Кто насел на мой процессор и сожрал его такты?

Потрошить пакеты 72
При содействии **Шона Конвэя** вскроем всю подноготную протокола TCP/IP и структуры сетевых пакетов. Но не применяйте эту технику в своей фирме без разрешения свыше.

АКАДЕМИЯ КОДИНГА

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

Системное время 76
Повелитель Вечности д-р **Крис Браун** углубился в историю, отыскивая начало всех начал и конец времен UNIX. На наше счастье, он установил, что пару сотен миллиардов лет мы еще протянем.

Chromebook: Сделай сам

Любимец мира ПК, **Нейл Мор**, поможет вам установить ОС от Google.

Наш эксперт

У **Нейла Мора** всё ломается с двух лет, после некого случая со спуском унитаза. С тех пор он пытается всё чинить, но это выходит неважно.

Ваш первый вопрос будет «а зачем мне вообще устанавливать Chrome OS, даже на моём старом ноутбуке, когда есть море абсолютно полноценных и крутых дистрибутивов Linux?» Хороший вопрос. Ответ в том, что не всем нужен полнофункциональный дистрибутив, и не всем он подходит. Успех Chromebook'ов — а 5,7 миллионов продаж за 2014 г. и 7,3 миллиона ожидаемых на 2015-й (источник: Gartner) — это успех, отчасти на счету их упрощённой, лёгкой ОС на базе Gentoo. Да и иметь полный доступ к сервисам Google весьма заманчиво.

Радует то, что Chromium OS, так называется открытая версия Chrome OS, доступной только на лицензированных Chromebook, устанавливается легко. Несмотря на очень непривычное, в синих тонах, оформление, Chromium OS работает с теми же учётными записями и сервисами Google и так же даёт возможность автоматически подключаться к своим плагинам и всему прочему, что хранится в облаке Google.

Система основана на Gentoo, то есть Linux, и потому имеет все преимущества ядра Linux но имейте в виду, что она появилась только в конце 2009 г., так что использовать её на более старом оборудовании, возможно, не получится. Однако мы попробовали установить её на стандартный ноутбук Lenovo X200, 2008 г. изготовления, и всё прошло гладко. Золотое правило: с любым стандартным железом Intel проблем быть не должно. Но проблемы непременно возникают с не-Intel'овскими беспроводными адаптерами; об этом мы ещё скажем подробнее.

Сборки Chromium OS присутствуют на ряде сайтов. Вряд ли кому захочется собирать её из исходников самостоятельно, так что есть готовые версии для *VirtualBox* и для запуска и установки с USB. Мы использовали сборку с http://arnoldthebat.co.uk, актуальную на май 2015 г. Ещё одна популярная версия есть на сайте Hexxeh, http://chromeos.hexxeh.net. Она не поддерживается с апреля 2013 г., но всё ещё работает.

Попробуйте Chromium OS

Для этого есть куча разных способов. Простейший — скачать образ, записать на USB-брелок или SD-карту и запустить на вашем ноутбуке или ПК. Есть и live-версии, но на записываемых устройствах можно сохранить свои настройки и носить ОС с собой.

Есть команда, позволяющая легко скопировать разделы OS на внутренний жёсткий диск (хотя система этого и не афиширует). Вся имеющаяся информация при этом стирается, но при дешевизне SSD вполне можно позволить себе купить отдельный диск. Мы попытались поставить Chromium параллельно с Linux Mint, но, похоже, Chromium воспринимал это как режим восстановления и отказывался запускаться. Дайте знать, если вам повезёт больше, потому что логически кажется довольно просто пересоздать два раздела ROOT-A и STATE, войти через **dd** с USB и обновить *Grub*.

Скорая помощь

ОС можно запускать с USB-брелка и работать с ней на любых ноутбуках, нетбуках и настольных системах. Linux с вами повсюду.

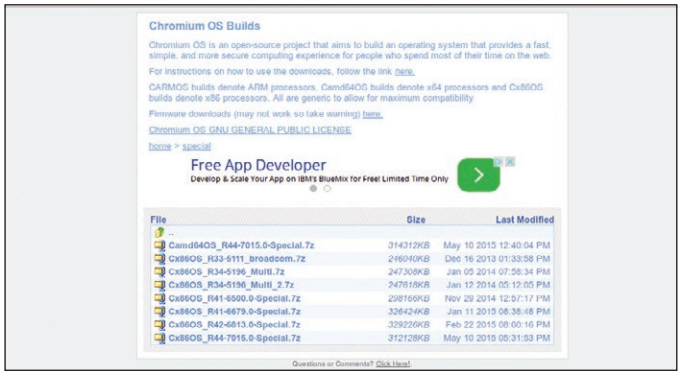
API-ключи для Google

Этот пункт пошаговой инструкции не обязателен и, честно говоря, довольно заковырист, поскольку ключи длинные, а если в вашем шрифте трудно различить 1, L и I, то и набирать такой ключ мучительно. Но если вы хотите подружить приложение Chromium Files с вашей внешней учетной записью Google Drive, способ на это есть.

Вам понадобятся ключи с https://console.developers.google.com. Под API, выберите Credentials [Учётные данные], затем Create New Client ID for Installed Applications [Создать новый ID клиента для установленных приложений], затем Other type [Другие], затем выберите Create New Key [Создать новый ключ]. Вы увидите три записи: свой API, ID и Secret Key. Далее важно добавить ваши переменные окружения в Chromium. Через *vi* добавьте в конец файла конфигурации ваши новые ключи и следующие три строки.

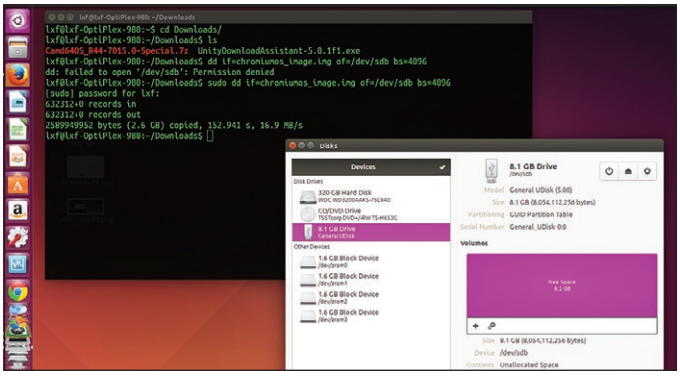
```
GOOGLE_API_KEY=
GOOGLE_DEFAULT_CLIENT_ID=
GOOGLE_DEFAULT_CLIENT_SECRET=
```

Установим и обновим Chromium OS



1 Скачиваемая сборка

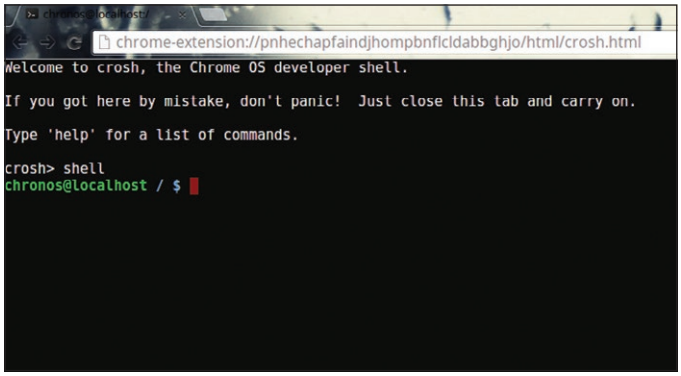
Выберите последнюю специальную сборку на http://bit.ly/LXFthebat — там есть все предыдущие драйверы, в том числе Broadcom Wireless, плюс непроверенные карты Realtek и Atheros. Обратите внимание, что Cx86OS и Camd64OS есть с 32-и 64-битной поддержкой, поэтому убедитесь, что вы берете правильную версию для вашего оборудования, чтобы оптимально использовать память.



2 Загрузочные носители

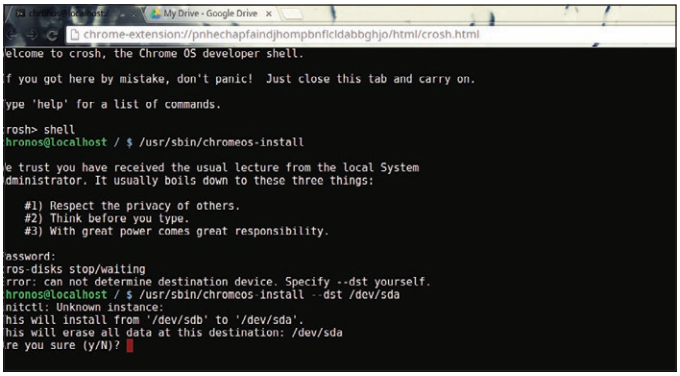
Файл 7ZIP весит 1 ГБ, несжатый файл IMG — 2,4 ГБ, а значит, объём USB-брелка или SD-карты для записи должен быть довольно большим. Вы можете запускать и использовать Chromium OS прямо с них, но, очевидно, для скорости лучше установить систему на внутренний SSD или HDD:

```
$ sudo dd if=<ИмяОбразаChromeOS.img> of=/dev/sdbX bs=4M
```



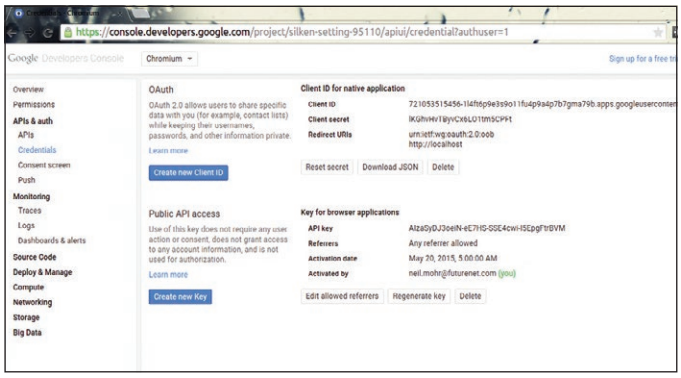
3 Запуск Chromium OS

Загрузите ноутбук с USB. Вероятно, придётся нажать волшебные F12/Del/Специальную клавишу, чтобы поменять приоритеты загрузки BIOS/UEFI. Вход с учётной записью не потребуется, поскольку можно использовать Гостевой просмотр. Нажмите Ctrl+Alt+T, чтобы попасть в *Crosh*, терминал Chrome Shell. Введите **shell**, чтобы открыть реальную оболочку Linux.



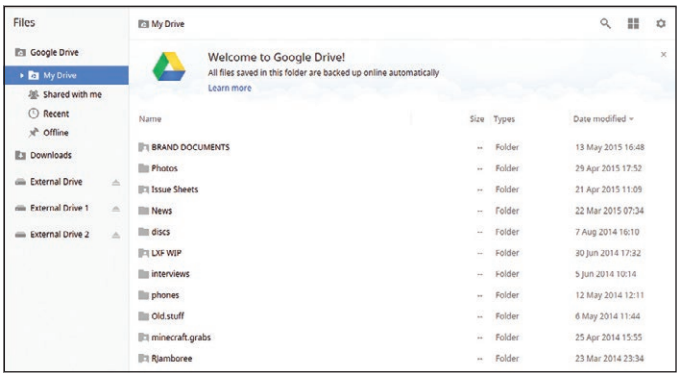
4 Установка Chromium OS

Внимание! Следующая команда сотрёт всё содержимое внутреннего жесткого диска, с dual-boot нам не повезло. Если запросят пароль, введите **password**, далее в командной строке введите **/usr/sbin/chrome-install --dev /dev/sda**, и начнётся установка. По завершении удалите USB-носитель, перезагрузитесь, и готово.



5 Получение API-ключей

Всё будет работать, но некоторые компоненты ОС не смогут получить доступ к услугам Google без ключей. Подпишитесь на эту группу Google Code http://bit.ly/LXFchromium, получите доступ на https://console.developers.google.com и выберите Create Project [Создать проект]. Активируйте Drive API/SDK, Google Maps Geolocation API, Google Now for Chrome API.



6 Добавление ключей

Ключи чувствительны к регистру, так что вводите их внимательно; при запросе на ввод пароля наберите **password**. Для этого откройте *Crosh* и наберите **shell**, **sudo su**, **mount -o remount,rw/**, **cd /etc**, **vi chome_dev.conf**, и в *vi* введите ключи, как описано во врезке на предыдущей странице. Нажмите Esc и скомандуйте **wq!**, чтобы выйти и снова войти. Отныне доступ к Chromium Files должен появиться. LXF

PulseAudio: Обслужим аудио

Джонни Бидвелл демонстрирует некоторые малоизвестные трюки *PulseAudio*, чтобы звук по вашей сети передавался на потоковое вещание.

Наш эксперт

Джонни Бидвелл нечаянно озвучил свой личный сервер *PulseAudio* на всю редакцию. Послышалась странная смесь из The Smiths, Элтона Джона и успокоительной песни китов.

В 2008 году, когда *PulseAudio* шагнул в наши системы, поднялся страшный шум: отовсюду посыпались жалобы на пропадающий либо хромающий звук, а то и падение системы. «Верните ALSA!», вопили невежды, естественно, не ведая, что ALSA никуда не делась и по-прежнему является предпочтительным для ядра вариантом взаимодействия системы с аудиоустройствами.

PulseAudio — это демон, обычно управляемый пользователем и способный управлять доступом (ну ладно, монополизировать доступ) к устройствам ALSA. Если у вас в каждый момент времени работает только одно приложение, использующее аудио, и есть только один аудиовыход, вы прекрасно обойдётесь без *PulseAudio*. Но ныне такое имеет место всё реже. Не столько потому, что некоторые сайты навязывают нам аудиоролики и звуковую рекламу, когда мы пытаемся послушать Spotify, сколько потому, что многим сегодня, наряду с традиционным аналоговым аудио через стереовывод, нравится слушать цифровое аудио через HDMI. Можно задействовать все выходы — через HDMI подключить телевизор или монитор, через аналоговый — свой проигрыватель в стиле 1990-х. И хотя ALSA прекрасно справляется с поддержкой нескольких звуковых устройств, *PulseAudio* умеет гораздо больше.

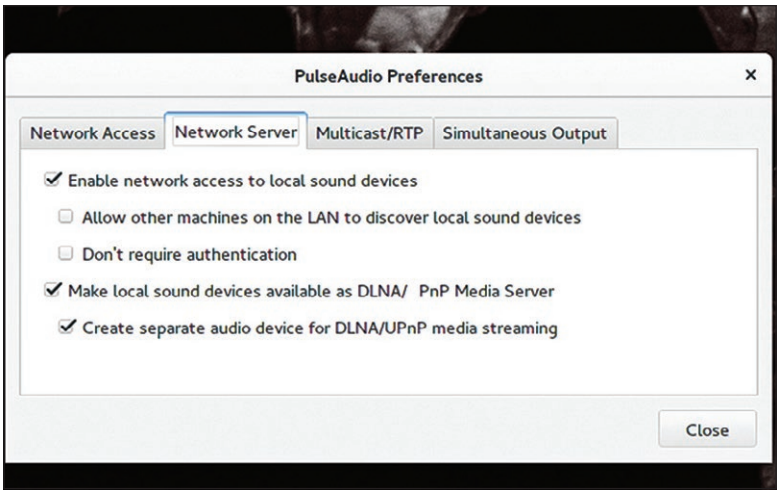
Допустим, например, что вы хотите объединить аналоговый и цифровой выход, т. е. сделать так, чтобы они работали одновременно. С *PulseAudio* это проще простого. GTK-микшер *pavucontrol* позволяет полностью контролировать *Pulse*, поэтому для начала с его помощью убедитесь, что внутренним устройством вывода по умолчанию является HDMI. Всё, что передаёт аудиосигналы в *PulseAudio*, называется источником [source], а всё, что преобразует их в шумы — приёмниками [sink]. Их список выведет команда

```
$ pacmd list-sinks
```

Она покажет всю информацию о ваших устройствах, и для вас самое здесь интересное — это номера устройств и подустройств ваших аналоговых и цифровых выходов. Наше аналоговое устройство значилось как 0, с подустройством, опять же, 0, следовательно, ниже оно фигурировало как **hw:0.0**. А наше цифровое аудиоустройство — **hw:1.3**. Теперь создадим новый приёмник под названием **combined** и сделаем его устройством по умолчанию, добавив следующие строки в **/etc/pulse/default.pa**, один из основных конфигурационных файлов *PulseAudio*.

```
load-module module-alsa-sink device=hw:0,0
sink_name=analogue
load-module module-alsa-sink device=hw:1,3 sink_name=digital
load-module module-combine-sink sink_name=combined
slaves=analogue,digital
set-default-sink combined
```

Тогда, вернувшись к *pavucontrol*, вы увидите новый выход, который будет назван как-то наподобие «Одновременный вывод на внутренний цифровой стереоканал (HDMI), внутреннее аудио». И теперь, во время воспроизведения аудио, на вкладке Playback [Воспроизведение] вы можете выбрать этот комбинированный выход для любых приложений. Если у вас достаточно новая версия *PulseAudio*, всё это можно сделать в программе *paprefs* — просто отметьте флажок Simultaneous Output [Одновременный вывод]. Если вы не поклонник графических удобств на базе GTKMM, предлагаемых *paprefs* и *pavucontrol*, то имеются и консольные версии: попробуйте *ponymix* и *pamixer*. Также стоит заглянуть в man-страницы *pacmd* — у этой оболочки масса возможностей.



› *Paprefs* даёт доступ к сетевым функциям *PulseAudio* без мороки с командной строкой.

Аудио в сети

Одно из самых приятных свойств *PulseAudio* — его умение воспроизводить аудиопотоки с удаленных машин. Потоковое аудио без сжатия требует надежного сетевого соединения; для воспроизведения необработанного аудио PCM в CD-качестве (стерео, 16 бит, 44,100 кГц) нужна стабильная пропускная способность около 1,4 Мбит/с. И если у вас есть только нестабильный Wi-Fi, стоит настроить MPD-сервер или нечто подобное, для удалённого управления локально хранящейся музыкой — дабы не доверять каждую фразу и каждый нюанс Пятой симфонии Бетховена эфиру, где обитают гномы, крадущие пакеты.

Поскольку *PulseAudio* является примером классической архитектуры клиент–сервер, эта функция реализуется довольно просто. Поясним, что машина, на которой воспроизводится аудио —

Передача на/с других медиа-устройств

Paprefs также позволяет клиентам передавать аудио на устройства Apple, AirTunes (на вкладке Network Access [Доступ к сети]). Если эта опция неактивна (а вам ну очень нужно передать музыку на свои фруктовые устройства или Kodi), возможно, потребуется установить пакет *pulseaudio-module-raop*. *Paprefs* также позволяет настроить машину как сервер DLNA/UPnP, чтобы звук с него воспроизводился на любом количестве домашних устройств для развлечений. Соответствующие параметры будут неактивны, пока вы не установите пакет *rygel*, обеспечивающий функциональность сервера UpnP. Сделав это, вы можете включить обе опции; вторая предоставляет отдельный регулятор громкости для вашего потока. *Rygel*, чтобы от него был толк, требует кое-какой настройки: надо включить плагины External или GstLaunch в **/etc/rygel.conf** и, в зависимости от вашего дистрибутива, стартовать *rygel* вручную.

Rygel может сжимать аудио в любой формат, что может облегчить все беды с пропускной способностью. Даже при сжатии FLAC lossless потребуется вдвое меньше бит в секунду. Чтобы настроить *rygel* помощью плагина Gstlaunch, добавьте следующий код в ваш **/etc/rygel.con**:

```
[GstLaunch]
enabled=true
launch-items=mypulseaudiosink
mypulseaudiosink-title=Audio on @HOSTNAME@
mypulseaudiosink-mime=audio/flac mypulseaudio-
sink-launch=pulsesrc device=upnp.monitor ! Flacenc
```

Это позволит воспроизводить аудио с вашего сервера PulseAudio, например, в Kodi или RaspBMC.

это сервер, а та, что передаёт — клиент. Можно передавать поток и из не-Linux систем, а также использовать zeroconf, чтобы сервер было проще обнаружить (без утомительного вбивания IP-адресов). На самом деле, всё это можно сделать и без редактирования файлов настройки, но нам это дело по душе, а кто-то и вообще захочет передать задачу удалённому серверу; так что начнём с методики работы «по-взрослому».

Первое и главное, что нужно *PulseAudio* — активировать модуль TCP. Делается это легко, добавлением в файл **/etc/pulse/default.pa** следующей строки:

```
load-module module-native-protocol-tcp
```

В целях безопасности, *PulseAudio* по умолчанию не принимает соединения от удаленных хостов, если их cookie не совпадает с имеющимся на сервере. Этот небольшой файл можно найти в **~/.config/pulse/cookie**, так что скопируйте его с сервера на клиент, или наоборот. Если у вас есть SSH-доступ к серверу (допустим, тот имеет IP-адрес 10.0.1.1), это можно сделать с машины-клиента вот так (изменив **user** на имя вашего пользователя на сервере):

```
$ scp user@10.0.1.1:/home/user/.config/pulse/cookie ~/.config/
pulse/cookie
```

Если безопасность вас не волнует, или волнует не настолько, чтобы синхронизировать эти cookies, можете вместо этого внести IP-адрес вашего клиента (допустим, 10.0.1.2), в белый список, добавив следующие параметры в строку load-module:

```
auth-ip-acl=127.0.0.1;10.0.1.2 auth-anonymous=1
```

Можно также указать маски подсети в списке управления доступом: например, для добавления адресов вида 10.0.1.x используйте **10.0.1.0/24**.

Применение Avahi для обнаружения

Чтобы обе машины стали видны другим клиентам, надо включить на обеих сервис *avahi*. В порядке отступления, Avahi — это протокол обнаружения, наподобие Bonjour от Apple, соавтором которого был не кто иной, как Леннарт Пёттеринг [Lennart Poettering], ведущий разработчик как *PulseAudio*, так и *Systemd* (см. стр. 54). В отличие от последних двух, его вклад в *Avahi* совершенно неоспорим. *Avahi* надо включить и на клиенте, и на сервере. Если вы работаете с *Systemd*, это делается так:

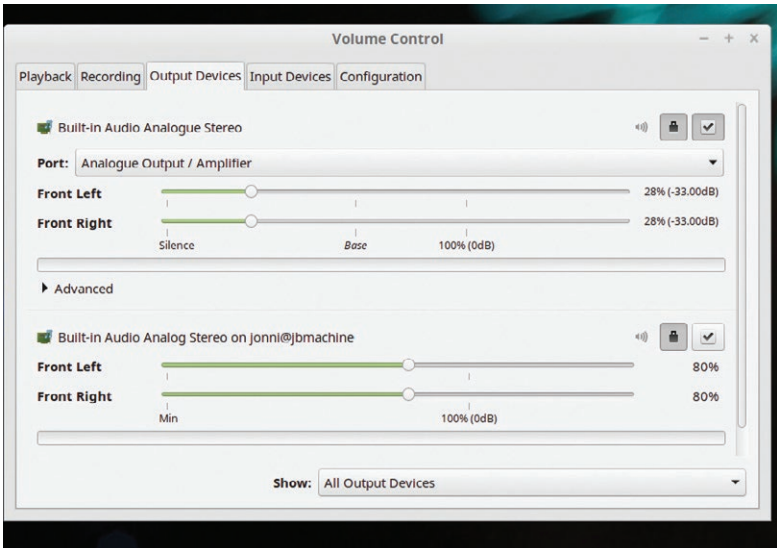
```
$ sudo systemctl start avahi-daemon
```

а для других систем инициализации — так:

```
$ sudo service avahi-daemon start
```

Теперь надо настроить использование *Avahi* в *PulseAudio* для обеих машин. На сервере добавьте следующую строку в файл **etc/pulse/default.pa**:

```
load-module module-zeroconf-publish
```



› При хорошем раскладе, ваш звуковой сервер будет выглядеть в *pavucontrol* так. Теперь вы можете направить туда все аудиовыходы, или только от отдельных устройств.

а на клиенте — такую:

```
load-module module-zeroconf-discover
```

Если всё хорошо, то при запуске **pavucontrol** на клиентской машине вы увидите сервер во вкладке Output Devices [Устройства вывода]. Можно перенаправить на него и всё аудио, и вывод отдельных приложений.

Как уже упоминалось, тут можно обойтись и без редактирования файлов конфигурации, и это правда так. Однако тогда и на клиенте, и на сервере надо установить *paprefs*. На клиенте отметьте первое поле во вкладке Network Access [Доступ к сети] для использования серверов *PulseAudio* в сети. На сервере, во вкладке Network Server [Сервер сети], отметьте первый — Enable network access [Включить доступ к сети], второй (чтобы быть видимым для других машин) и (если вы не хотите возиться с cookies авторизации *PulseAudio*) третий флажок.

В разных дистрибутивах используются различные версии *PulseAudio*, но при возникновении трудностей всегда можно попробовать перезапустить службу *avahi-daemon*. TCP-порт прослушивания сервера *PulseAudio* — 4713, поэтому убедитесь, что нет никаких препятствий типа брандмауэров. Наконец, можно попробовать форсировать запуск, используя переменную среды PULSE_SERVER. Например, для запуска медиа-плейера *Totem* с нашего сервера —

```
$ PULSE_SERVER=10.0.1.1 totem
```

Вот и всё, мы настроили передачу аудио через удалённый клиент. LXF

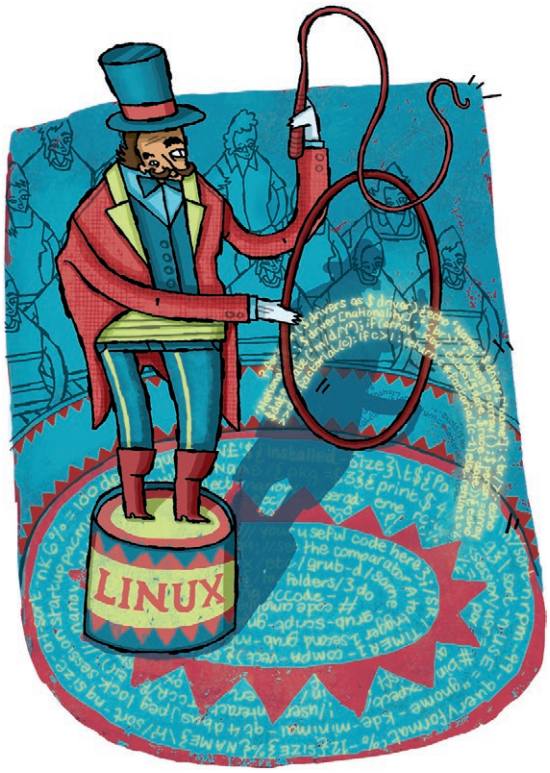
» **Подпишитесь на печатную или электронную версии** на www.linuxformat.ru/subscribe!

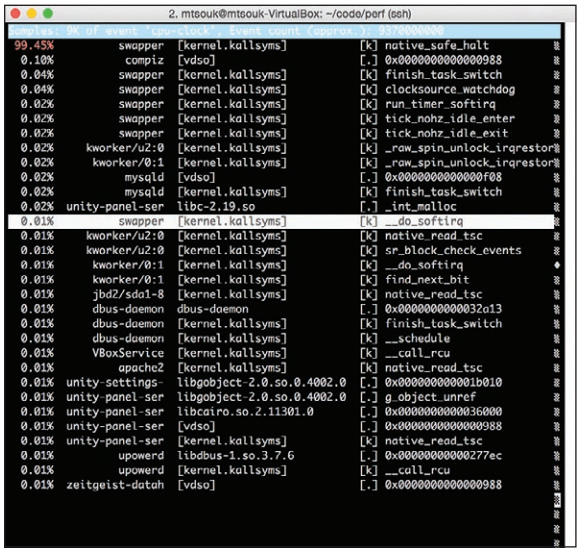
Perf_events: Трассируем ядро

Михалис Цукалос объясняет, как начать анализировать производительность вашей системы Linux с помощью *perf_events*.

Наш эксперт

Михалис Цукалос — администратор Unix, программист, математик и администратор базы данных, развивший в себе способность делиться начетверо, чтобы выступать во всех этих ролях одновременно.





› Это первая порция выведенного отчёта команды perf с данными, собранными с perf record -a –F 1000 sleep 10. Вы можете получить больше информации о каждой строке, выделив её и нажав Enter.

На этом уроке мы рассмотрим *perf_events*, инструмент для составления профилей и трассировок Linux. Одной из важнейших причин профилирования Linux является возможность быстро и эффективно разобраться с потреблением CPU. Настройки производительности позволяют не только решать проблемы, но и использовать ваше ПО более эффективно без напрасной траты ресурсов.

Команда **perf**, изначально именовавшаяся Performance Counters for Linux (PCL, счетчики производительности) — это целый набор инструментов для создания профилей и протоколов, что сейчас называется Linux Performance Events (LPE). Каждый отдельный инструмент используется в качестве подкоманды **perf**, но мы не будем рассматривать все подкоманды, а ограничимся теми, которые вы будете применять чаще всего. Иногда самым сложным моментом в их применении является выбор, какую подкоманду предпочесть; желая узнать больше об определённой подкоманде, просто используйте **perf help <имя_подкоманды>**.

Установить *perf_events* довольно просто, поскольку он является частью основного ядра; к примеру, в системе Ubuntu, работающей на ядре Linux 3.13.0-46, *perf_events* устанавливается с помощью

```
$ apt-get install linux-tools-common linux-tools-3.13.0-46-generic linux-cloud-tools-3.13.0-46-generic
```

от имени root, что обеспечит все необходимые пакеты. Примечание: как вы, скорее всего, догадались, точные имена пакетов, которые вы установите, зависят от версии ядра.

Perf_events не только проще в установке, чем *DTrace* и *SystemTap*, но и проще в изучении. Тем не менее, все три инструмента потребуют определённого привыкания, прежде чем вы сможете эффективно их использовать.

После успешной установки вы можете вызывать *perf_events* командой **perf**. Почти все подкоманды требуют прав root для запуска, но изучать используемую версию *perf_events* можно с помощью **perf --version** без привилегий root.

Perf_events передаёт данные на уровень пользователя для пост-обработки с помощью среды скриптов, которая добавит некоторые непроизводительные издержки, особенно при отслеживании часто совершаемых событий, тогда как *DTrace* и *SystemTap* умеют производить обработку и сбор данных одновременно.

Perf_events поддерживает и статическую, и динамическую трассировку. Статическая трассировка может производиться с использованием любых доступных точек трассировки, а динамическая трассировка выполняется посредством определения

Скорая помощь

Если вы администрируете только системы Linux, то просто обязаны изучить и *SystemTap*, и *perf_events*. Если же вы администрируете различные машины UNIX, лучшим выбором, возможно, будет изучение *DTrace*.

Дым от огненных диаграмм

Огненные (пламенеющие) диаграммы сто́ят знакомства с ними, поскольку у них интересные и уникальные характеристики. Огненные диаграммы визуализируют профиль стекового фрейма и позволяют понять, как используется CPU, простым и понятным образом.

Каждый блок в огненной диаграмме представляет собой функцию в стеке (см.рисунок на стр. 70). Значения по оси Y — это количество фреймов в стеке. Функции в верхнем краю — это те, что запускаются непосредственно CPU. Функция, которая находится под другой функцией, является её родительской функцией; именно это и показывает трассировка стека.

Значения по оси X охватывают обследуемую совокупность. Более широкий блок функций означает три вещи: функция вызывалась чаще либо была медленнее, чем узкий блок функций или комбинация предыдущих двух вариантов. Если во время получения данных для вашей огненной диаграммы будет запущено много программ, основание диаграммы будет состоять из большего числа отдельных блоков.

Огненная диаграмма не показывает количество вызовов или прошедшее время, поскольку расположение слева направо не имеет иного значения, кроме алфавитного порядка. Да и цвета в огненной диаграмме роли не играют (хотя они и красивы), то есть можно для своей диаграммы выбрать цвета, приятные лично вам.

Созданные огненные диаграммы SVG имеют ещё одно интересное свойство: они интерактивны. Если вы наведёте мышь на различные элементы, встроенный код JavaScript покажет детали внизу изображения.

Огненные диаграммы, как и другие наглядные изображения, решают проблемы реального мира, если вы способны понимать и интерпретировать выводимую ими информацию. И они спасают вас от необходимости разбирать обильные данные трассировки стека! Их также можно использовать для визуализации памяти, ввода/вывода и т.д.

и создания специальных зондов для сбора информации ядра на лету! Следующий вывод показывает число событий точек трассировки за сеанс:

```
$ perf list | grep "Tracepoint event" | awk -F: {'print $1'} | sort | uniq -c | sort –rn
582 syscalls
89 ext4
35 xen
25 writeback
...
```

Как вам должно быть очевидно из приведенного вывода, большинство событий точек трассировки относится к системным вызовам.

Применим perf_events

Perf_events работает в две фазы: первая фаза — сбор данных и создание файла с данными, а вторая — анализ данных и обработка этого файла.

Следующий вывод показывает, как собрать данные:

```
$ sudo perf record -a -F 1000 sleep 10
[perf record: Woken up 2 times to write data]
[perf record: Captured and wrote 0.990 MB perf.data (~43268 samples)]
$ ls -l
total 988
-rw------- 1 root root 1041188 Apr 14 13:04 perf.data
```

Команда **perf** запускается от имени root через **sudo**, поскольку она использует различные системные ресурсы для сбора данных. Параметр **-a** велит **perf** осмотреть все CPU, а **-F** определяет частоту профилирования. Выполняемая команда — **sleep 10**, которая диктует, что данные собираются в течение 10 секунд. Если вы захотите вручную завершить выполнение в любое время, достаточно нажать Control+C.

Если у вас уже есть файл **perf.data**, старый будет переименован в **perf.data.old**. Если у вас есть и **perf.data**, и **perf.data.old** и вы запускаете **$ perf record**, старый **perf.data.old** будет потерян, поскольку его место займёт существующий **perf.data** и будет создан новый **perf.data**. Вы можете проанализировать файл **perf.data**, используя **$ perf report**. [См. вывод, изображённый на стр. 68.]

Строка вывода, включающая фразу <not supported [не поддерживается]>, показывает, что в ядре, с которым вы работаете, отсутствует поддержка функций, необходимых **perf**, чтобы составить профиль этих аспектов вашей программы. Вам не следует беспокоиться о них, поскольку это не ваша вина. Такое обычно встречается в виртуализированных сетях.

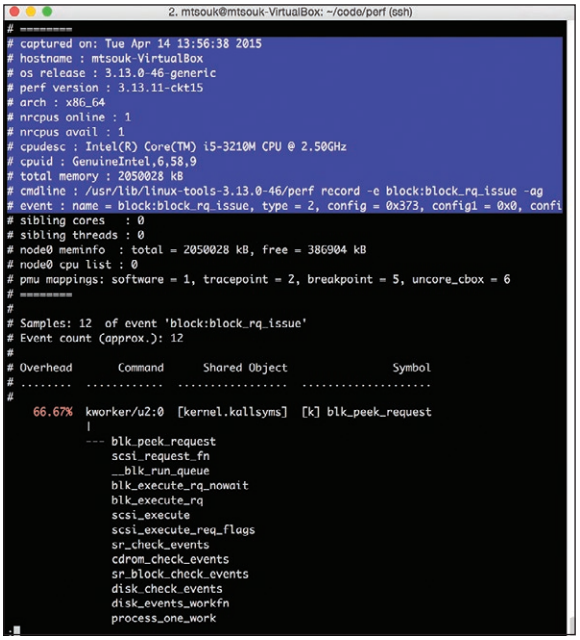
Получение информации о вводе/выводе

Бывает, что диски ведут себя плохо; тогда вам надо проверить их производительность. Зонд **block:block_rq_issue** запускается при каждом запросе на ввод/вывод в блочное устройство. Следующая команда **perf** отслеживает зонд **block:block_rq_issue**:

```
$ sudo perf record -e block:block_rq_issue -ag
[perf record: Woken up 1 times to write data]
[perf record: Captured and wrote 0.562 MB perf.data (~24543 samples)]
```

Нажмите Ctrl+C, чтобы остановить выполнение вышеупомянутой команды **perf**. Параметр **-g** даёт команду *perf_events* отследить также следы пакета [см. снимок экрана внизу с примером вывода **sudo perf report --stdio -l**]. Здесь очень важно, что вы можете увидеть в деле полную команду **perf**, а также информацию о хосте, дате, памяти, версии ядра и CPU машины Linux, применяемой во время процесса сбора данных. Информацию об ошибках страницы легко получить следующим образом:

```
$ sudo perf record -e page-faults -a
$ sudo perf report --stdio
Samples: 231 of event 'page-faults'
Event count (approx.): 28551
...
```



Скорая помощь

Вы можете узнать больше на официальной вики http://bit.ly/PerfWiki. Чем больше вы узнаете о доступных зондах, тем проще всё это становится, так что время от времени выполняйте **sudo perf list** и читайте вывод.

Этот вывод создан с помощью sudo perf report --stdio -l. Параметр --stdio отключает интерактивный режим подкоманды отчёта и позволяет обрабатывать или фильтровать вывод текста, используя команды Unix.

```
37.88 % apache2 libphp5.so [.]
0×000000000038a3e6
30.94 % sh ld-2.19.so [.] _dl_relocate_object
2.61 % bash [unknown] [.]
0×000000000045cff0
```

Первая команда записывает все ошибки страницы и (опять-таки) может быть остановлена нажатием Ctrl+C. Запуск команды отчёта в виде **$ sudo perf report -n --stdio** добавит столбец с образцом отсчёта вывода. Вывод показывает, из-за каких команд произошло наибольшее число ошибок страницы. Если вы захотите найти все зонды (т.е. пункты трассировки), относящиеся к ошибкам страницы, выполните

```
$ sudo perf list | grep faults
```

Огненные диаграммы

Огненные диаграммы, тоже созданные разработчиком **perf**, Брэнданом Греггом [Brendan Gregg], являются впечатляющим способом представления данных, собранных с помощью *perf_events*. Чтобы создать огненные диаграммы, понадобятся два внешних скрипта Perl, но весь процесс относительно прост, а его результат одновременно впечатляющ и информативен. Следующие команды **perf** иллюстрируют процесс:

```
$ sudo perf record -F 99 -a -g -- sleep 60
$ sudo perf script | ./stackcollapse-perf.pl > flame.perf
$ ./flamegraph.pl flame.perf > perfFlame.svg
```

Созданный результат зависит от собираемой информации [см. пример результата ниже].

Объяснение огненной диаграммы — задача не всегда тривиальная, это приходит с опытом. Вы можете получить оба скрипта Perl (*stackcollapse-perf.pl* и *flamegraph.pl*) на https://github.com/brendangregg/FlameGraph.

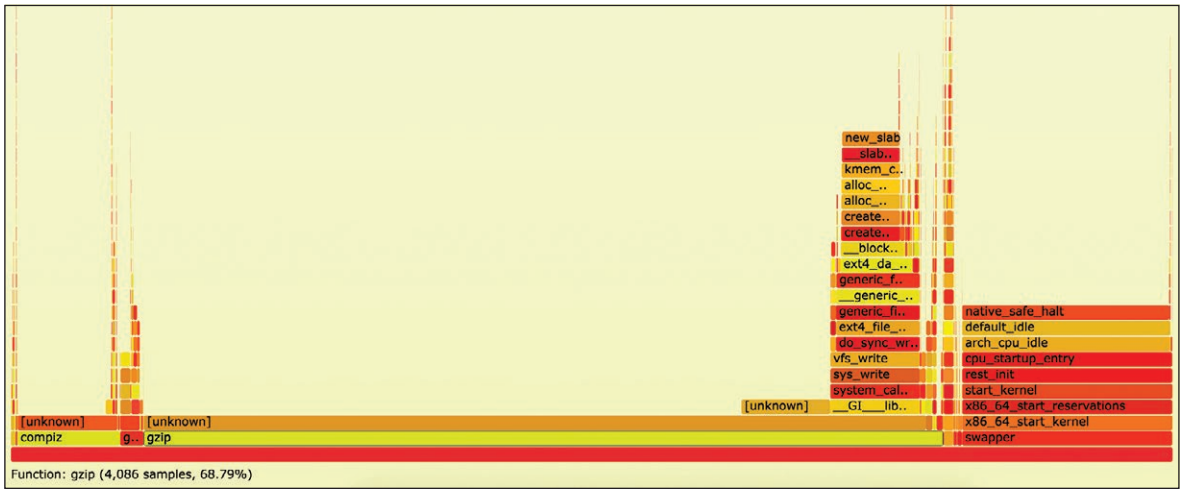
Пожалуйста, помните, что если ваше ядро Linux не содержит необходимой информации по отладке, то при выполнении команды **$ sudo perf script** вы можете увидеть следующие предупреждения (тогда ваша огненная диаграмма будет содержать не столь детальную информацию):

```
no symbols found in /bin/gzip, maybe install a debug package?
[Не найдено символов в /bin/gzip, м.б. установить отладочный пакет?]
Failed to open /tmp/perf-2294.map, continuing without... no symbols found in /usr/lib/x86_64-linux-gnu/dri/swrast_dri.so, maybe install a debug package?
[Не удалось открыть /tmp/perf-2294.map, продолжено без этого, не найдено символов в /usr/lib/x86_64-linux-gnu/dri/swrast_dri.so, м.б. установить отладочный пакет?]
no symbols found in /usr/sbin/VBoxService, maybe install a debug package?
[Не найдено символов в /usr/sbin/VBoxService, м.б. установить отладочный пакет?]
```

Создание гистограммы

Ещё один очень удобный способ получения высококлассного обзора происходящего в вашей системе — гистограммы. Хотя *perf_events* не поддерживает гистограммы напрямую, он может достаточно просто создавать их с помощью скрипта оболочки, написанного Брэнданом Греггом.

Мы бы рады сказать, что гистограммы с помощью *perf_events* так же легко создаются и настраиваются, как с помощью *DTrace*



› Огненная диаграмма — впечатляющий способ визуализации данных, полученных с помощью *perf_events*. Её также можно применить к данным, созданным другими инструментами анализа производительности, включая *DTrace*, *OS X Instruments*, *SystemTap* и *Google Chrome Developer Tools*.

Узнаём больше о разных подкомандах

Запустив команду **perf** без аргументов или опций, вы увидите подсказку, включающую все доступные подкоманды, так что давайте пробежимся по ним.

» **kmem** имеет две опции: **perf kmem stat** сообщает широкую статистику памяти системы, а **perf kmem record <команда>** записывает события kmem произвольной нагрузки.

» **diff** читает два файла **perf.data** и показывает различия между ними.

» **stat** запускает команду и собирает статистику производительности. Это подкоманда не создаёт файла **perf.data**; она немедленно возвращает запрошенную информацию.

» **trace** показывает события, связанные с целью, включая системные вызовы, а также другие системные события, такие как ошибки страницы. Эта подкоманда не создаёт автоматически файл **perf.data** и может использоваться взамен команды **strace(1)**.

» **record** предназначена для взятия образцов активности CPU и создаёт файл **perf.data** для дальнейшей обработки. Эта подкоманда может использоваться в комбинации с другими подкомандами.

» **report** применяется для анализа созданных ранее файлов **perf.data**. На входной файл указывает параметр **-i**, за которым следует имя файла. Без **-i perf** будет искать в текущей директории для **perf.data**.

» **timechart** — подкоманда, являющаяся инструментом для визуализации общего поведения системы под нагрузкой. Сгенерированное изображение создаётся в формате SVG.

» **top** — подкоманда, создающая и отображающая профиль счётчика производительности в реальном времени. Её вывод напоминает утилиту *top*.

» **annotate** — подкоманда, читающая файл **perf.data** и отображающая аннотированную версию кода. Если в файле объекта есть отладочные символы, то исходный код будет отображён вместе с кодом сборки. Если в объекте нет отладочной информации, отобразится только аннотированная сборка.

или *SystemTap*, но это не так. Приведённый ниже вывод показывает процесс попытки создания:

```
$ sudo ./perf-stat-hist syscalls:sys_enter_write count 15
Tracing syscalls:sys_enter_write, power-of-4, max 1048576, for 15 seconds...
Трассировка системных вызовов sys_enter_write, степень 4-х, за 15 секунд...
Range      : Count      Distribution
[Диапазон            : Число      Распределение]
-> -1      : 0          |          |
0 -> 0     : 0          |          |
4 -> 15: 5753 | ########### |
...
```

Необходимый вам скрипт оболочки *Bash*, *perf-stat-hist*, имеется на https://github.com/brendangregg/perftools/blob/master/misc/perf-stat-hist.

К гистограммам относится режим счётчика *perf_events*, который суммирует события, происходящие в ядре Linux, и передаёт результат в область пользователя. Например, команда

```
$ sudo perf stat -e 'syscalls:sys_enter_*' -p 1339
```

сосчитает системные вызовы для процесса с PID 1339, тогда как

```
$ sudo perf stat -e 'syscalls:sys_enter_*' -a sleep 15
```

сосчитает системные вызовы для всей системы за период в 15 секунд. Очень удобная команда —

```
$ sudo perf top -e raw_syscalls:sys_enter -ns comm
```

Она показывает системные вызовы за процесс, обновляя вывод каждые 2 секунды.

Следующая команда сосчитает системные вызовы, необходимые для выполнения определённой команды, и напечатает ненулевые значения счётчика с помощью awk:

```
$ sudo perf stat -e 'syscalls:sys_enter_*' gzip <a_command> 2>&1 | awk '$1!= 0'
```

Использование подкоманды **$ perf** при возможности лучше, поскольку её издержки меньше.

Создание теплокарт

Тепловая карта — это разумный способ визуализации числовой таблицы, в которой вы заменяете реальные значения цветными ячейками. Хорошая новость в том, что можно создавать теплокарты, используя данные, собранные *perf_events* с помощью другого скрипта Perl. Этот процесс требует следующих действий:

```
$ sudo perf record -e block:block_rq_issue -e block:block_rq_complete -a sleep 120
$ sudo perf script | awk '{ gsub(/:/, "«) } $5 ~ /issue/ { ts[$6, $10] = $4 } $5 ~ /complete/ { if (l = ts[$6, $9]) { printf " %.f  %.f\n", $4 * 1000000,($4 — l) * 1000000; ts[$6, $10] = 0 } }' > out.perf
$ ./trace2heatmap.pl --unitstime=us –unitslat=us --maxlat=50000 out.perf > perfHM.svg
```

Пример, который мы создали (см. внизу), по оси X показывает временной отрезок, по оси Y — латентность, а ось Z — это количество вводов/выводов в течение данного диапазона времени и латентности, которое представлено различными цветами.

Вы найдёте скрипт Perl *trace2heatmap.pl* на https://github.com/brendangregg/HeatMap. Выполните **$ ./trace2heatmap.pl –help**, чтобы увидеть все доступные опции.

Создание точки трассировки

В заключение вкратце расскажем, как определить новую точку трассировки. Для начала используем команду **$ perf probe –add <новая_точка>** для добавления новой точки трассировки под названием **новая_точка**, а **$ perf probe -d <имеющаяся_точка>** удалит существующую точку трассировки. Если вы хотите найти все динамические устройства сбора информации, выполните **sudo perf probe -l**. Если точка трассировки, которую вы пытаетесь добавить, не может быть найдена в ядре, вы получите сообщение об ошибке, похожее на это:

```
$ sudo perf probe --add некая_точка
Failed to find path of kernel module.
[Не удалось найти путь к модулю ядра.]
Kernel symbol 'некая_точка' not found.
[Символ ядра 'некая_точка' не найден.]
Error: Failed to add events. (-2)
[Ошибка: Не удалось добавить события. (-2)]
```

Иногда для поддержки новой точки трассировки может понадобиться перекомпилировать свое ядро Linux из исходника, чтобы включить необходимую информацию.

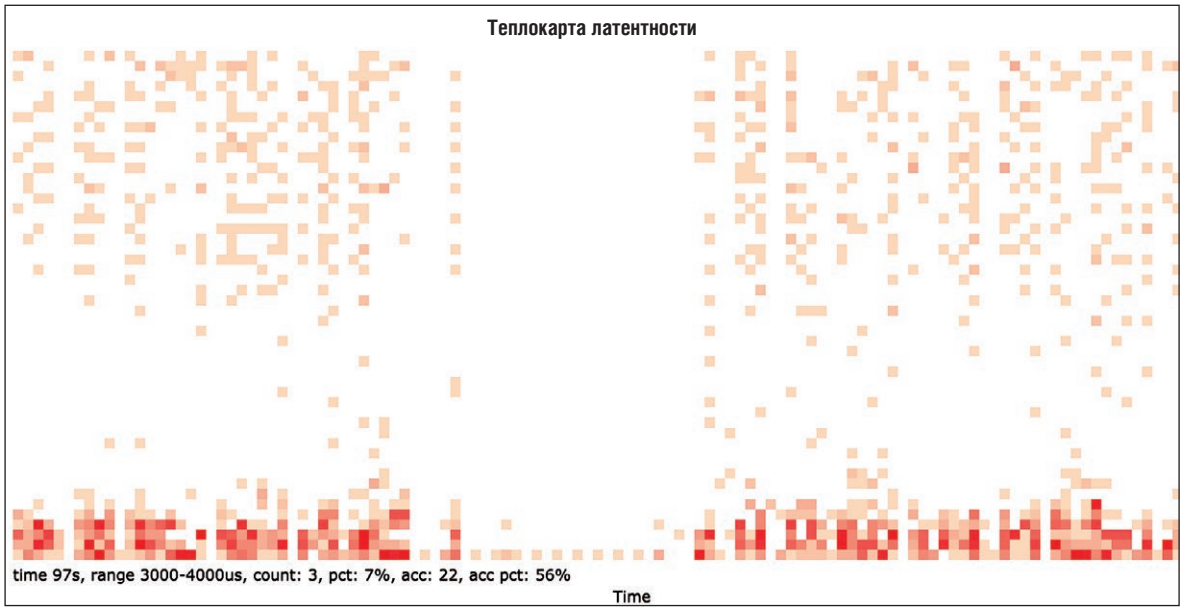
Perf_events продолжает развиваться и улучшаться от версии к версии. Было бы здорово, имей он все функции *DTrace*, но на данный момент у *DTrace* есть преимущество. Сможете ли вы выполнить свою работу с помощью *perf_events* или вам понадобятся расширенные и зрелые функции *DTrace*, решать вам. Помните, что *DTrace* отличается сложностью установки и не всеми дистрибутивами Linux поддерживается в полной мере.

Настройка производительности и решение проблем похожи на отладку, и эта задача требует большого опыта, так что будьте терпеливы и упорны. Знание *perf_event* дает отличную базу. LXF

Скорая помощь

Хорошая книга, посвящённая теме производительности систем, которая также немного затрагивает *perf_events* — Systems Performance: Enterprise and the Cloud [Производительность систем: Предприятие и облако] Брэндана Грэгга.

Через месяц: Внутри ядра 4.0



› Это теплокарта, альтернативный способ представления чисел на экране. Данные, собранные *perf_events*, можно превратить в такую карту, обработав их с помощью внешних программ или скриптов.

Tcpdump: Улов данных

Шон Конвэй поясняет азы протокола сетевого соединения TCP/IP и учит собирать данные утилитой командной строки *tcpdump*.

Наш эксперт

Шон Конвэй применяет на практике свой 30-летний опыт работы с электронными технологиями, чтобы помочь системным администраторам отточить свои навыки.

Скорая помощь

Есть мнение, что *tcpdump* для сетевого анализа — то же, что *vi* для редактирования. Оба валят работу на вас, когда другие инструменты помогают жить.

Лучший способ освоить *tcpdump* — использовать его на практике, чтобы осознать, как компьютеры взаимодействуют друг с другом в сети. Наш урок объединяет команды *tcpdump* и теорию работы в сети. С помощью этих команд мы выудим данные из сети и покажем основные компоненты, применяемые для сетевого взаимодействия TCP/IP.

Данные передаются по сети от устройства к устройству в блоках, именуемых блоками данных протокола (БДП [PDU, Protocol Data Units]), и состав БДП зависит от самого сетевого протокола. Считайте один БДП товарным вагоном на сортировочной станции, который перевозит грузы в виде данных, и каждый БДП отличается от остальных по типу переносимых данных.

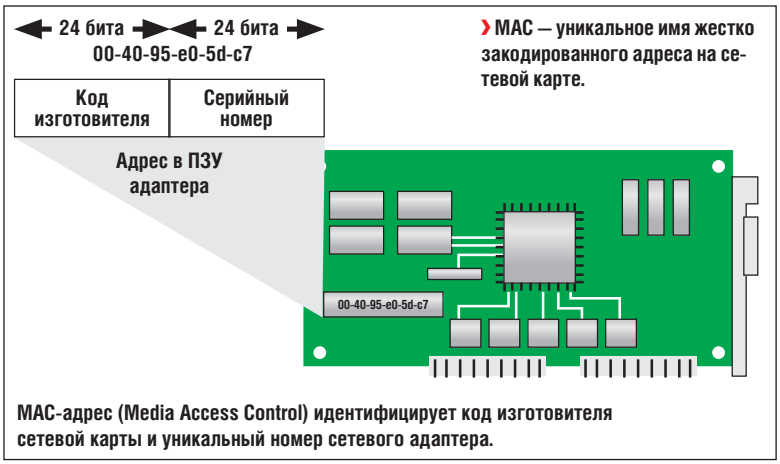
Tcpdump — это утилита/инструмент/программа интерфейса командной строки, используемая для сбора (например, снифинга) контента БДП, передаваемых через сеть. Этот инструмент не воздействует на содержимое БДП во время передачи данных по сети; это немного похоже на наблюдение за рыбкой, которая плавает в аквариуме.

Прежде чем приступать к использованию *tcpdump*, стоит обзавестись некой основой знаний о передаче данных. Эта основа будет ориентиром для получения новых знаний по передаче структурированных данных. А потом приобретенные знания можно использовать для объяснений работы *tcpdump* с БДП.

Один из вариантов сквозной передачи структурированных данных предлагает базовая эталонная модель Open Systems Interconnection (OSI). Данная модель является общей базой для координации стандартов, с целью взаимосвязи систем. Функции эталонной модели OSI организованы в иерархию уровней. Каждый уровень выполняет поднабор общих функций, необходимых для передачи, а также использует сервисы, предоставляемые более низкими уровнями, и, в свою очередь, предоставляет сервисы более высоким уровням. Эта модель — удобный инструмент для обсуждения сетевой коммуникации и изучения определенных протоколов (см. рис. на стр. 73 вверху), который демонстрирует все уровни модели OSI. Чертеж был обработан, чтобы сосредоточиться на четырех уровнях протокола TCP/IP, и *tcpdump* может рассмотреть эту информацию.

Вверху слева на схеме располагается хост, которому нужна коммуникация в сети. Сообщение входит в систему через программу в Application Layer [Уровень приложений], и протокол этого уровня упаковывает его в пакет вместе с некой контрольной информацией и передает БДП на Presentation Layer [Уровень представления]. Программы Уровня представления трамбуют данные, упаковывают некую информацию в заголовок и передают весь БДП на Session Layer [Уровень сессии]. Как и следует ожидать, Уровень сессии устанавливает сессию с контрольной информацией в заголовке и передает весь БДП на Transport Layer [Транспортный уровень]. Транспортный уровень разбивает информацию более высокого уровня на сегменты и упаковывает контрольную информацию в заголовок, и передает сегмент на Network Layer [Уровень сети]. Этот уровень упаковывает сегменты более высокого уровня с сетевыми адресами, пригодными для маршрутизации, и передает пакеты на Data Link Layer [Уровень передачи данных]. Уровень передачи данных надставляет информацию более высокого уровня заголовком и трейлером, применяет физический адрес и информацию контроля за ошибками и передает фреймы на Physical Layer [Физический уровень], который и помещает информацию фрейма БДП в сеть. Уф-ф...

Теперь перейдем на схеме стр. 73 вправо и рассмотрим движение сообщения вверх по модели до пункта назначения, изучая первые четыре уровня, поскольку они относятся к протоколу TCP/IP. Эта сторона предоставляет информацию о том, как создаются БДП с разными типами поля, размерами поля в байтах и часто используемыми именами.



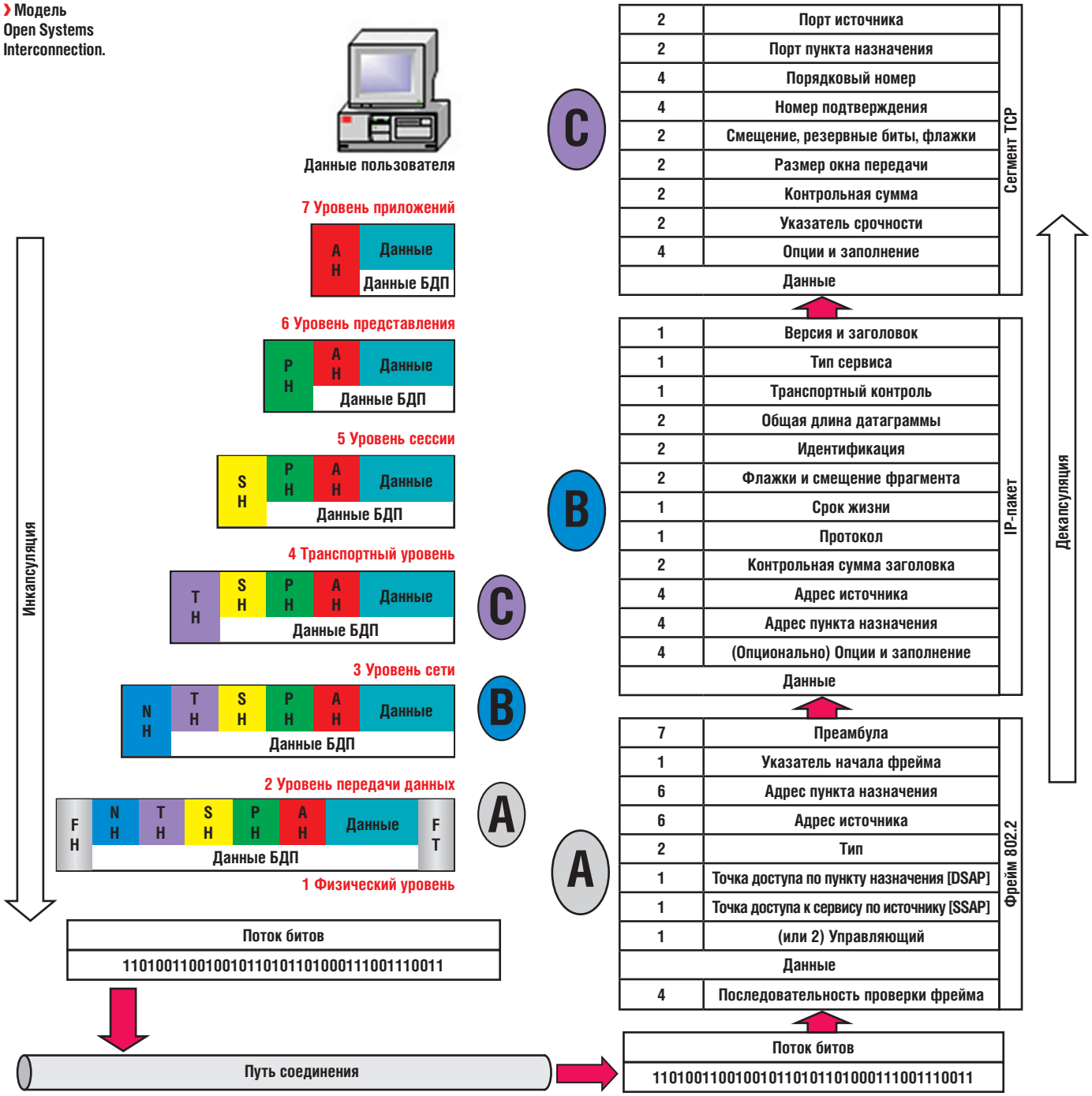
› MAC — уникальное имя жестко закодированного адреса на сетевой карте.

MAC-адрес (Media Access Control) идентифицирует код изготовителя сетевой карты и уникальный номер сетевого адаптера.

Изучаем уровни

Физический уровень — это место, где поток нулей и единиц передается на носитель. Этот уровень обеспечивает функции и процедуры для перемещения двоичных единиц на физический носитель. В сети LAN двоичные единицы могут быть электрическими сигналами или световыми импульсами, в зависимости от того, используется ли медный провод CAT5 или оптоволокно, и за поддержку этой функции отвечает сетевая карта компьютера (NIC). Уровень передачи данных (см. А на схеме стр. 73) обеспечивает надежный коммуникационный сервис в физическом канале, передавая данные в блоках БДП, которые называются фреймами. Фреймы обеспечивают контроль за временем, потоком и ошибками в поле разного размера, определенном стандартами фреймов. Уровень сети (см. В на схеме стр. 73) обеспечивает сервисы

› Модель Open Systems Interconnection.



переключения и передачи между конечными системами. Это делается посредством адресации, использованной в пакетах БДП. Транспортный уровень (см. C на схеме вверху) обеспечивает надежную сквозную передачу благодаря исправлению ошибок и контролю потока в вызываемых сегментах БДП.

Схема демонстрирует концепцию инкапсуляции данных. Это делается показом текущих битов во фрейме Ethernet, который содержит IP-пакет, содержащий TCP-сегмент. Фрейм, пакет и сегмент БДП следуют спецификации, которая активирует инструмент, такой, как *tcpdump*, чтобы он обратил внимание на определенный участок (например, байт).

Данные перемещаются по сетевому носителю в виде нулей и единиц в электрическом медном носителе или в виде вспышек света в оптоволоконном носителе. Нули и единицы в проводе образуют фрейм, и фреймы перемещаются между шлюзами с помощью MAC-адресации. MAC-адреса, физический адрес, адрес устройства или адрес Ethernet — все они относятся к одному и тому же адресу, присвоенному физическому устройству.

Каждое отдельное соединение в интерфейсе сетевой карты имеет MAC-адрес, и фрейм, как уже говорилось, перемещается от одного шлюза к другому, используя MAC-адрес. Когда фрейму надо перейти в другую сеть (например, через роутер), MAC-адрес фрейма изменяется, отобразив эту точку.

Хост X намерен обратиться к хосту Y. IP-адреса, если их предъявить, укажут, что хосты находятся в разных сетях. Хост X отправляет фрейм с MAC-адресом пункта назначения роутер A. »

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

Затем фрейм с MAC-адресом пункта назначения следующего перескока отправляется с роутера A на роутер B. Роутер B отправляет фрейм с MAC-адресом на хост Y. И роутер A, и роутер B принимают решения относительно того, по какому маршруту фрейм пойдет, в зависимости от IP-адреса в инкапсулированном пакете. Фрейм создается с разными MAC-адресами источника и пункта назначения, но информационное наполнение пакета остается одним и тем же по всему пути. MAC-адрес иногда называют «адресом следующего перескока [next hop address]», поскольку он доводит фрейм только до очередного перескока.

Вы можете отобразить MAC-адреса хоста Linux, в котором находитесь, набрав в консоли команду **ifconfig** (получившийся вывод сокращен):

```
# enp3s0: flags=4163<UP,BROADCAST,RUNNING,MULTIC AST>
mtu 1500
inet 192.168.2.253 netmask 255.255.255.0 broadcast
192.168.2.255
ether 30:85:a9:49:94:10 txqueuelen 1000 (Ethernet)
.....
```

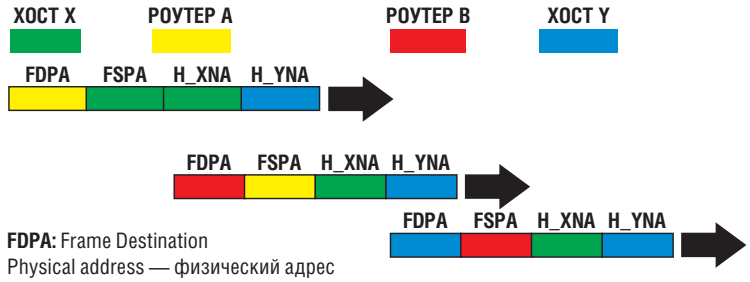
MAC-адрес — это текст, который следует за заголовком вывода **ether**. Чтобы отобразить MAC-адреса роутера в сети, используйте

```
$ arp -a
# homeportal (192.168.2.1) at 60:c3:97:e8:f3:31 [ether] on enp3s0
```

Прежде чем приступать к использованию *tcpdump*, вам обязательно следует прочитать наше предупреждение (см. врезку *Политика компании*, стр. 75).

Для данных нашего урока исходная система — это рабочий стол, где используется браузер, а пункт назначения — сервер, на котором запущен web-сервер. IP-адреса источника и назначения хоста находятся в одной и той же сети. (В целях экономии места вывод от множества команд был сокращен до данной конкретной части.)

› Фрейм с чисто физическим адресом вбирает пакет с IP-адресацией.



FDPA: Frame Destination Physical address — физический адрес пункта назначения для фрейма.
FSPA: Frame Source Physical address — физический адрес источника фрейма.
H_XNA: Host_X Network address (datagram) — сетевой адрес хоста X (датаграмма).
H_XNA: Host_Y Network address (datagram) — сетевой адрес хоста Y (датаграмма).

Примечание: Физический адрес изменяется при переходе датаграммы от шлюза к шлюзу, тогда как сетевой адрес остается одним и тем же.

Если вы хотите следовать нашему уроку, надо будет заменить имя NIC (например, enp3s) нашего тестового компьютера именем, которое покажет вывод команды **ifconfig** на вашем компьютере. Более полное объяснение опций см. на www.tcpdump.org.

При использовании *tcpdump* MAC-адреса вашего хоста (т.е. источника) и роутера (т.е. пункта назначения) отображаются с помощью фрейма данных. Согласно схеме OSI (см. стр. 73 вверху), MAC-адресация — это адресация источника и пункта назначения, которая появляется во фрейме на уровне передачи данных:

```
$ sudo tcpdump -e -i enp3s0
# 16:12:54.519026 30:85:a9:49:94:10 (oui Unknown) >
60:c3:97:e8:f3:31 (oui Unknown), ethertype IPv4 (0x0800), length
84: unknown3085A9499410.38548 > homeportal.domain: 23184+
PTR? 5.2.168.192.in-addr.arpa. (42)
```

Стрелки можно использовать, чтобы определить, откуда (источник) и куда (пункт назначения) идет БДП. Не пугайтесь, если выводимая информация беспрестанно прокручивается по экрану. Помните, что интерфейс работает свободно в неизбирательном режиме, поэтому отображает всё, что находит. Чтобы выйти из команды *tcpdump*, нажмите и удерживайте Ctrl+C.

Созерцаем проплывающие БДП

С помощью *tcpdump* можно посмотреть IP-адрес хоста (т.е. mybox) и сайта (например, web server). Обратите внимание, на иллюстрации OSI (см. стр. 73) IP-адреса — это адресация источника и пункта назначения, которые появляются в пакете.

```
$ sudo tcpdump -n -i enp3s0
# 17:34:36.468664 IP 192.168.2.253.38412 > 192.168.2.225.
http: Flags [S], seq 2642752770, win 29200, options [mss
1460,sackOK,TS val 332382530 ecr 0,nop,wscale 7], length 0
```

Если добавить в команду вторую **n** (**-nn**), то в дополнение к обращению DNS к поиску IP-адреса также произойдет и разрешение имен портов в номера. Имя сервиса HTTP будет разрешено в порт номер 80.

Если в команде *tcpdump* убрать опцию **-n**, отобразятся имена хоста (т.е. mybox) и сайта (например, web server).

```
$ sudo tcpdump -i enp3s0
# 17:30:27.428424 IP mybox.38409 > web server.http: Flags [S],
seq 1847458607, win 29200, options [mss 1460,sackOK,TS val
332133490 ecr 0,nop,wscale 7], length 0
```

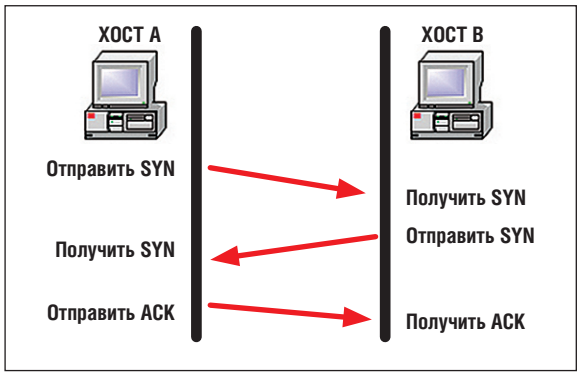
Возникли проблемы с поиском того самого БДП с определенным IP-адресом? Добавьте в команду параметр хоста, и будет отображаться только трафик, содержащий определенный IP-адрес. Вот основные части одного БДП, отображенного в HEX и ASCII для определенного IP-адреса хоста с использованием *tcpdump*.

```
$ sudo tcpdump -X -i enp3s0 host 192.168.2.225
# listening on enp3s0, link-type EN10MB (Ethernet), capture size
65535 bytes
17:53:24.208005 IP mybox.38422 > web server.http: Flags [S],
seq 458400957, win 29200, options [mss 1460,sackOK,TS val
333510270 ecr 0,nop,wscale 7], length 0
```

Непоследовательно последовательный

Если один и тот же термин не используется постоянно, или используется несколько терминов для обозначения одного и того же, велика вероятность, что начинающий системный администратор может запутаться. Пример — говорят, что по сети передаются пакеты, и это точно лишь до известной степени: фактически, по сети передаются фреймы, которые несут информацию пакета. На схеме фрейма БДП (выше) адресацией считается адресация источника и пункта назначения. Этот адрес может называться адресом управлением средства доступа [media access control] (MAC-адресом), физическим адресом, адресом оборудования или адресом Ethernet, и все термины будут верными в зависимости от того, откуда происходит информация.

Для сисадминов лучше всего будет употреблять правильный термин, а не тот, что нынче в моде. Неплохой отправной точкой могут стать стандарты типа модели OSI, если вам нужен эталон, поскольку она помогает внести большую ясность. Применив свои знания и *tcpdump*, мы можем изучить функции управления трафиком в протоколе TCP/IP. Функции имеют структуру поля в определенном месте, которое *tcpdump* определяет и показывает. Функции управления трафиком обеспечивают доставку сервиса, подобно созданию соединения с сайтом или доставке данных из одного пункта в другой.



› Проследуйте по стрелкам через три БДП данных и найдите компоненты трехэтапного квитирования.

```
0x0000: 4500 003c 52c3 4000 4006 60ca c0a8 02fd
E..<R.@.@.`.....
```

На Уровне сети IP-адресация является ключевым моментом, который следует изучить в пакете. Можно подготовить *tcpdump* к поиску определенных IP-хостов с помощью опций, описывающих адрес. Например, ниже мы заставили *tcpdump* смотреть на определенное поле адреса, источника или назначения, или на трафик в определенной сети:

```
$ sudo tcpdump -i enp3s0 src 192.168.2.253
$ sudo tcpdump -i enp3s0 dst 192.168.2.253
$ sudo tcpdump -i enp3s0 net 192.168.2.0/24
```

Давайте переместим протокол на уровень выше уровня, который занимается IP-адресами — на Транспортный уровень. На этом уровне используется сегментация наряду с портами источника или пункта назначения. Поля портов — это отдельные точки доступа к определенным сервисам. Сегментация делит данные на сегменты, помещая в контрольное поле порядковый номер и подтверждение, чтобы способствовать реконструкции данных.

Определение портов

Два примера портов — 80 для HTTP и 22 для SSH. Web-браузер может соединиться с web-сервером через порт 80, но если сервер безопасен, это будет порт 443. Эти порты являются стандартными. Некоторые сайты любят вносить путаницу, предлагая сервисы (например, SSH) на другом порте, чтобы злоумышленнику пришлось поискать сервисы где-то в нестандартном месте. Информация в **/etc/services** перечисляет сервисы и соответствующие им порты, и этот файл используется для обеспечения имен портов, когда в *tcpdump* используется опция разрешения (т.е. **-nn**). Порты — это точки, где устанавливаются соединения с сервисами, и с помощью **netstat** мы можем увидеть их действия:

```
$ netstat -an | more
# Active Internet connections (servers and established)
# Proto Recv-Q Send-Q Local Address Foreign Address State
# tcp 0 0 0.0.0.0:22 0.0.0.0:* LISTEN
# tcp 0 0 192.168.2.225:22 192.168.2.253:53643 ESTABLISHED
```

В первой строке сервис SSH слушает запросы соединения, а в другой произведено соединение SSH с IP-адресом web-сервера. Можно поставить фильтр номера порта в *tcpdump*, используя переключатель ссылки порта:

```
$ sudo tcpdump -X -i enp3s0 port 80
```

Опционно вы также можете добавить к портам описатели **scr** и **dst**, чтобы сузить поиск:

```
$ sudo tcpdump -X -i enp3s0 src port 8080
$ sudo tcpdump -X -i enp3s0 dst port 50001
```

TCP/IP — надежный протокол, за что следует благодарить функции, которые находятся на Транспортном уровне модели OSI в пакете протокола управления транспортом [transport control protocol] (TCP). В дополнение к поддержке портов для соединения с известными сервисами, есть также флажки для установления соединений и порядковых номеров, а также подтверждений, для обработки сегментов (см. схему вверху стр. 73).

Чтобы расширить наши знания, рассмотрим, как хост устанавливает соединение с web-сервером с помощью TCP/IP.

Перед отправкой данных нужно установить соединение между двумя конечными точками. Для транспортировки отправляемые данные должны быть разбиты на БДП. На транспортном уровне поле SYN в сегменте открывает запрос соединения, а поле FIN заканчивает соединение. Отправляемые сегменты подчиняются порядковым номерам и используют подтверждения для уведомления других отправителей, что получены все необходимые сегменты. Если во время транспортировки сегменты теряются, то TCP может запросить повторную передачу потерянных сегментов.

TCP/IP устанавливает надежное соединение посредством трехэтапного квитирования [handshake]. Флажки, порядковые номера и поля подтверждений из схемы вверху слева заполняются определенными данными, собранными с помощью *tcpdump*, и соединение начинается, когда Хост A отправляет сегмент, содержащий флаг SYN.

Одним из способов применения *tcpdump* для анализа потенциальной проблемы является сбор данных в файл и его последующее изучение. Например, мы создали двоичный файл *tcpdump* с именем **tcpdump_output** при помощи

```
$ sudo tcpdump -nnvvvXS -i enp3s0 src 192.168.2.253 or src
192.168.2.225 -w /tmp/tcpdump_output
```

Чтобы прочитать содержимое файла через *tcpdump*, наберите

```
$ sudo tcpdump -nnvvvXS -r /tmp/tcpdump_output
```

Данные, произведенные *tcpdump*, используют двоичный формат, поддерживаемый и другими инструментами, например, такими, как *Wireshark* и *Snort*.

Во время занятия в одном техническом колледже преподаватель попросил студентов сформулировать представление о том, каких результатов они ожидают, прежде чем обратиться к оборудованию для опытов. Чтобы выполнить это задание, студентам надо было понять, что они ищут, а не надеяться найти «что-нибудь». Предложение преподавателя было и остается отличным способом мотивировать студентов использовать свои знания прежде, чем приступать к исследованиям. Если результат исследований соответствует вашим ожиданиям, можете двигаться дальше и взглянуть с другой точки зрения.

На то, чтобы овладеть этим навыком, нужно время. Анализ сетевого трафика — дело непростое, и эффективное использование *tcpdump* требует хорошего понимания методов передачи структурированных данных. И нет лучшего способа обретения знаний, чем использовать инструмент, который требует от вас определить, что вы ищете, перед тем, как вы окунетесь в работу. *Tcpdump* — это точильный камень, который поможет вам освоить и отточить свои навыки сетевой коммуникации. LXF

Скорая помощь

Собранные данные, предназначенные для сайта, чей URL начинается с HTTP, могут содержать имена пользователей и пароли в разделе информации БДП.

Политика компании

Чтобы *tcpdump* собрал данные по сети, он работает в неизбирательном режиме, то есть вместо отбора данных, предназначенных исключительно для этого хоста, интерфейс отображает все собранные данные подряд.

Большая часть сетевого трафика не зашифрована. Этот инструмент может дать нежелательный доступ к данным, таким, как имена пользователей и пароли, передаваемые по сети. Поэтому использование инструмента *tcpdump* в рабочей среде может стать нарушением политики безопасности компании.

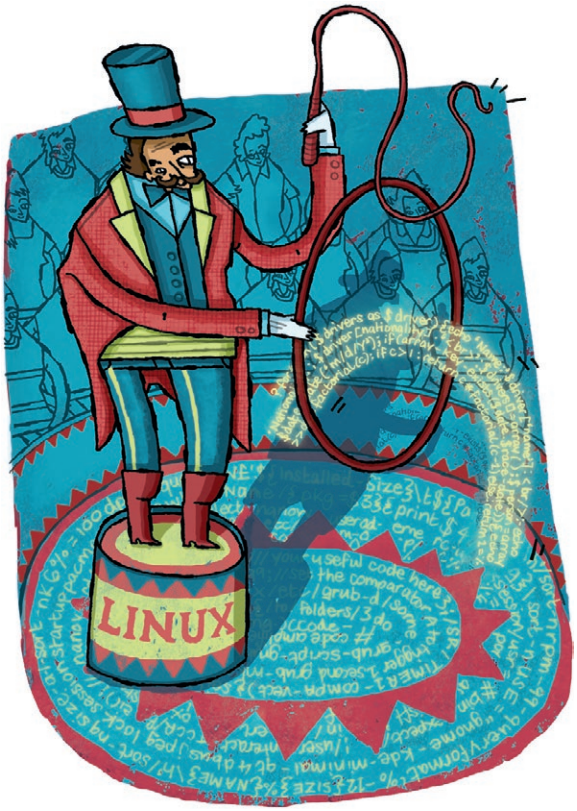
Перед использованием этого инструмента в рабочей сети ознакомьтесь с политикой безопасности вашей компании. Без получения вами должных разрешений у компании может появиться основание для вашего увольнения.

Система: Время и его Функции

В четвертой части своей серии уроков по системному программированию доктор **Крис Браун** решил коснуться краткой истории времени.

Наш эксперт

Доктор **Крис Браун** обучает, пишет и консультирует по Linux. Ученая степень по физике элементарных частиц ему в этом совсем не помогает.



Эту статью я сдал с опозданием. Редактор просмотрел ее и сказал: «Ага, тут про время». Как точно он подметил! Вы думаете, что время — довольно простая штука. Если вы не путешествуете во времени к моменту Большого взрыва, не живете рядом с черной дырой и не перемещаетесь со скоростью, близкой к скорости света, то время просто течет секунда за секундой, не так ли? Да, так, но время может стать сложнее, чем вам кажется.

Начнем с начала... время в Unix и Linux традиционно измеряется в секундах относительно момента, который называется «эпохой» и представляет собой самое начало 1970 г. (т.е. полночь 1 января) в часовом поясе UTC+0. Это количество секунд хранится в виде целого числа и определяется с помощью типа **time_t**. В 32-битных системах **time_t** — 32-битное знаковое целое; в 64-битных системах — 64-битное. Системный вызов **time()** возвращает системное время Unix в соответствии с моим определением. И мы можем напечатать его так:

```
#include <stdio.h>
#include <time.h>
main()
{
  printf("%ld\n", time(NULL));
}
```

Другой способ — сделать это в командной строке с помощью **date**:

```
$ date +%s
1428834498
```

В какой точке земного шара и в каком часовом поясе вы ни находились бы и на каком языке ни говорили бы, если все мы запустим программу в один и тот же момент времени, мы все получим одинаковый результат... по крайней мере в той степени, в которой верны часы наших компьютеров. Правда, такое представление времени не слишком полезно для тех из нас, кто пытается (например) вспомнить, какой сегодня день недели, или понять, как не опоздать на поезд от Кингс-Кросс до Кембриджа в 11:53. Но такое представление важно — не в последнюю очередь потому, что оно используется для временны́х штампов, записываемых в inode файла [см. «Академия кодинга», стр. 88 **LXF197**].

Если вам нужно разрешение точнее, чем одна секунда, воспользуйтесь вызовом **gettimeofday()**, который вернет структуру **timeval** со временем в секундах и микросекундах:

```
struct timeval {
  time_t tv_sec; /* seconds */
  suseconds_t tv_usec; /* microseconds */
};
```

Поэтому мы можем написать «более точную» версию нашей программы:

```
#include <stdio.h>
#include <sys/time.h>
main()
{
  struct timeval tv;
  gettimeofday(&tv, NULL);
  printf("%ld.%06ld\n", tv.tv_sec, tv.tv_usec);
}
```

которая даст примерно такой результат:

```
1428854228.042352
```

Однако не следует путать разрешение с точностью. Часы вашего компьютера способны определить время с точностью в лучшем случае в несколько десятых миллисекунды. Конечно, если с помощью **gettimeofday()** вы измеряете временной интервал, вызывая ее перед началом какой-то операции и сразу после ее завершения и вычитая результаты, абсолютная точность часов не имеет значения.

Время как человеко-читаемая строка

Существует несколько способов преобразования **time_t** в более удобное (для восприятия человеком) представление. Один из самых простых — **ctime()**. Давайте расширим нашу исходную программу:

Скорая помощь

Демон протокола NTP (Network Time Protocol — сетевой протокол времени) следует запускать не менее чем на трех серверах времени. Его легко настроить, и он сохранит часы вашего компьютера синхронизированными с точностью до нескольких десятых миллисекунды.

```
#include <stdio.h>
#include <time.h>
main()
{
  time_t t;
  t = time(NULL);
  printf("%s\n", ctime(&t));
}
```

которая напечатает строку вроде

```
Sun Apr 12 18:36:12 2015
```

При этом преобразовании учитывается ваш часовой пояс, но не учитываются настройки локали; сокращенные названия дней и месяцев всегда отображаются по-английски.

Хотя функция **ctime()** — дешевый и приятный способ получения времени в удобном для восприятия виде, с таким временем нельзя выполнять арифметические операции. Например, предположим, что вы хотите определить, попадает ли заданный момент времени **time_t** в интервал до или после полудня. Для этого удобнее воспользоваться функциями **gmtime()** и **localtime()**. Обе функции возвращают так называемое «разобранное» время в структуре tm, которая выглядит так:

```
struct tm {
  int tm_sec; /* секунды */
  int tm_min; /* минуты */
  int tm_hour; /* часы */
  int tm_mday; /* день месяца */
  int tm_mon; /* месяц */
  int tm_year; /* год после 1900 */
  int tm_wday; /* день недели */
  int tm_yday; /* день года */
  int tm_isdst; /* летнее время */
};
```

Поля этой структуры в основном говорят сами за себя, но учтите, что не все они независимы друг от друга: некоторые, например, **tm_wday** и **tm_yday**, вычисляются по другим. Последнее поле **tm_isdst** — это флаг, означающий, действует ли в данный момент летнее время. Различие между **gmtime()** и **localtime()** в том, что первая возвращает время в часовом поясе UTC+0, а последняя — в локальном часовом поясе. (Сокращение "gm" означает "Greenwich Mean Time [Среднее время по Гринвичу]", которое предшествовало UTC).

Вот маленькая программа, которая проверяет, что сейчас вечер пятницы:

```
#include <stdio.h>
#include <time.h>
main()
{
  time_t t;
  t = time(NULL);
  struct tm *tm;
  tm = localtime(&t);
  if (tm->tm_hour >= 12 && tm->tm_wday == 5)
  printf("It's Friday afternoon!\n");
}
```

Часовые пояса

Скорее всего, у вас есть некое представление о часовых поясах, даже если какие-то детали вам незнакомы. Часовые пояса — часть той цены, которую мы платим, чтобы жить на вращающейся сфере. Полдень в Лондоне — это семь утра в Нью-Йорке, и т. д. Как я упоминал ранее, функция **localtime()** выполняет преобразование времени с учетом локального часового пояса. Но откуда она узнает, в каком мы часовом поясе? А из переменной окружения. Например, команда

```
$ export TZ=":CET"
```

устанавливает часовой пояс в центральноевропейское время [Central European Time]. Значение этой переменной просто определяет имя файла в каталоге **/usr/share/zoneinfo** (в данном случае — **/usr/share/zoneinfo/CET**); в этом файле определяется смещение часового пояса относительно UTC и правила преобразования стандартного времени в летнее и наоборот. Это двоичный файл, созданный компилятором часовых поясов (zic), но его можно просмотреть с помощью **zdump**.

Изменим нашу последнюю программу так, чтобы она выводила часы и минуты «разобранного» времени, возвращаемого функцией **localtime()**:

```
time_t t;
t = time(NULL);
struct tm *tm;
tm = localtime(&t);
printf("%d:%d\n", tm->tm_hour, tm->tm_min);
```

Давайте запустим эту программу для нескольких часовых поясов:

```
$ export TZ=":GB"
$ ./localtime
10:47
$ TZ=":EST5EDT"
$ ./localtime
5:47
$ TZ=":Europe/Paris"
$ ./localtime
11:47
```

Итак, в Великобритании, где я живу, 10:47; в часовом поясе EST5EDT (Нью-Йорк и др.) — на пять часов меньше; а в Париже — на один час больше. (Часовой пояс EST аналогичен EST5EDT, с тем исключением, что он не содержит никакой информации о летнем времени). Пример с "Europe/Paris" демонстрирует альтернативный способ указания часового пояса. Он работает точно таким же образом — значение TZ определяет имя файла в **/usr/share/zonefinfo**, но в данном случае этим файлом будет **/usr/share/zoneinfo/Europe/Paris**.

В большинстве случаев смещение часового пояса относительно UTC выражается целым количеством часов: Лос-Анджелес — UTC-9, Москва — UTC+3, и т. д. Но есть и несколько исключений. Известный пример — Непал (UTC+5:45). А Крайст-Чёрч (один из колледжей Оксфордского университета) имеет собственный часовой пояс, который отстает от UTC на пять минут. Впрочем, не думаю, что его студенты все как один переводят часы.

»

Арифметика времени

Для преобразования разобранного времени в **time_t** можно воспользоваться **mktime()**. В данном смысле эта функция противоположна **localtime()**. Она также игнорирует поля структуры **tm_wday** и **tm_yday** (которые все равно выводятся из других полей структуры).

Умная черта функции **mktime()** в том, что она «нормализует» значения: например, интерпретирует 68 минут как 8 минут плюс один час, 35 января как 4 февраля и т. д. А следовательно, с помощью этой функции можно выполнять арифметические операции над данными о времени; например, следующий фрагмент кода выведет дату и время, которые отличаются от текущего времени ровно на одну неделю и один час:

```
time_t t;
struct tm *bdtime;
t = time(NULL);
bdtime = localtime(&t);
bdtime->tm_mday += 7; /* Добавим неделю */
bdtime->tm_min += 60; /* и час */
t = mktime(bdtime);
printf("Через неделю это будет %s\n", ctime(&t));
```

С помощью **mktime()** также легко вычислить количество секунд между двумя моментами времени в «разобранной» структуре. Для этого достаточно сделать следующее:

```
time_t timespan;
timespan = mktime(bdtime1)
- mktime(bdtime2);
```

Скорая помощь

Будьте осторожны с такими функциями, как **ctime()**, которые могут возвращать результаты в статически выделенном участке памяти. Если вы хотите вызывать их несколько раз, обязательно скопируйте возвращаемую функцией строку в собственную переменную, прежде чем вызывать функцию снова. В противном случае второй вызов может перезаписать результат первого.

Если переменная окружения TZ не установлена, **localtime()** считывает информацию о часовом поясе из **/etc/localtime**. На моем компьютере этот файл — просто копия **/usr/share/zoneinfo/Europe/London**.

Весной — вперед

Часовые пояса — не единственная проблема, когда дело доходит до локального представления времени. Несколько раз я упомянул летнее время. Возможно, вы помните фразу «Весной вперед, осенью назад». Правила перевода стрелок довольно разнообразны. В ЕС летнее время продолжается с последнего воскресенья марта до последнего воскресенья октября. В США — со второго воскресенья марта до первого воскресенья ноября, за исключением Аризоны, где стрелки не переводятся. Летнее время существует в некоторых австралийских штатах, только часы там переводят, разумеется, наоборот — вперед в октябре и назад в апреле. Во многих странах вообще нет летнего времени. Поэтому неудивительно, что бóльшая часть содержимого типичного файла **zoneinfo** посвящена указанию начала и конца летнего времени для каждого года.

Пока мы говорили в основном о двоичном представлении времени. Как же напечатать нечто удобное для восприятия человеком? Ну, это удобно сделать функцией **ctime()**, о которой я уже упомянул. Но мы столкнулись с другим препятствием: в разных культурах дни недели и месяцы имеют разные названия. Январь по-французски — “Janvier”, а среда по-немецки — ”Mittwoch”. Для учета этих и других культурных различий в Linux есть локали.

Стандарт Posix сообщает, что «локаль — это определение подмножества окружения пользователя, которое зависит от языка и культурных традиций». На практике локаль задает правила отображения денежных сумм, времени и даты, а также чисел. Начальный список доступных локалей определяется географическим положением, которое вы указали при установке Linux, но можно добавить и другие локали. Например, чтобы добавить немецкую локаль в Ubuntu, выполните команду

```
$ sudo apt-get install language-pack-de
```

Список доступных локалей можно просмотреть командой **$ locale -a**.

Локали расположены в подкаталогах каталога **/usr/share/locale**. Там обязательно будут две локали, Posix и C, которые представляют собой одно и то же и соответствуют традиционному поведению Unix.

› В Linux имеется несколько способов представления времени и куча функций для преобразования между ними.

Различные культурные традиции, которые определяются в локалях — даты, числа, валюта и т. д. — являются относительно независимыми. Поэтому локали делятся на следующие категории:

Категория	Назначение
LC_TYPE	Классификации символов и правил преобразования регистра
LC_COLLATE	Правила сравнения (словарного упорядочивания) набора символов
LC_MONETARY	Правила отображения сумм в валюте
LC_NUMERIC	Правила отображения чисел
LC_TIME	Правила отображения времени и даты

В программе можно выбрать локаль для заданной категории следующим образом:

```
setlocale(LC_MONETARY, “de_DE.utf8”);
```

или установить локаль сразу для всех категорий:

```
setlocale(LC_ALL, “de_DE.utf8”);
```

Однако в настоящих программах локали так жестко не задаются. В конце концов, назначение локалей отчасти в том, чтобы позволить нам писать «интернационализированные» программы, т. е. программы без встроенных предположений о языке и культуре. Поэтому вы скорее воспользуетесь вот этим вызовом:

```
setlocale(LC_ALL, “”);
```

который получит локаль из переменной окружения **LC_ALL**. Таким образом, этими аспектами поведения программы можно управлять с командной строки, изменяя переменную окружения **LC_ALL**, например, так:

```
$ export LC_ALL=de_DE.utf8
```

Фактически можно отдельно управлять отдельными категориями локалей, устанавливая переменные окружения, которые соответствуют категории. Например, команда:

```
$ export LC_MONETARY=fr_FR.utf8
```

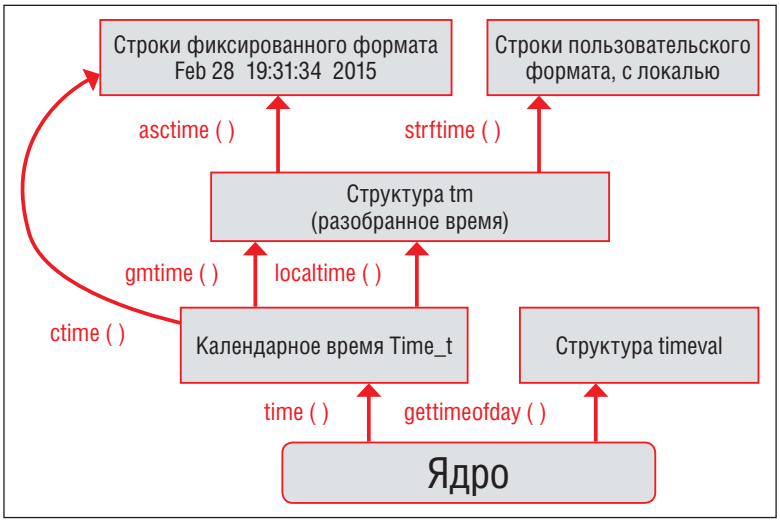
устанавливает только локаль, отвечающую за отображение денежных сумм. Но я не собираюсь показывать здесь подобные примеры. Обратите внимание, что если программа не вызывает функцию **setlocale()**, будет использована локаль Posix. Хорошо, давайте рассмотрим функцию преобразования времени, которая обращает внимание на локаль — **strftime()**. Эта функция принимает «разобранное» время и форматирует его в соответствии с заданной строкой формата. Давайте сразу возьмемся за дело:

```
/* Несложное демо strftime */
#include <stdio.h>
#include <time.h>
#include <locale.h>
int main()
{
  time_t t;
  struct tm *bdtime;
  char buf[1000];
  setlocale(LC_ALL, “”);
  t = time(NULL);
  bdtime = localtime(&t);
  strftime(buf, 1000, “It is %H:%M on %A %d %B %Y”, bdtime);
  puts(buf);
}
```

Мы уже встречались с вызовами **time()** и **localtime()**. Вызов **setlocale()** — новый для нас, а самая интересная строка — это вызов **strftime()**. Коды формата, которые я использовал здесь, таковы: **%H** для часов, **%M** для минут, **%A** для полного названия дня недели, **%d** для дня месяца, **%B** для полного названия месяца и **%Y** для четырехзначного года.

Однако кодов формата гораздо больше, и чтобы с ними ознакомиться, надо исследовать man-страницу. Так как программа использует функции **localtime()** и **strftime()**, на их поведение влияют обе переменные окружения, и TZ, и LC_ALL. Для начала попробуем запустить ее с настройками по умолчанию для моего расположения:

```
$ unset TZ; unset LC_ALL
$ ./strftime
```



```
It is 21:12 on Tuesday 28 April 2015
```

Теперь переключим часовой пояс на восточное побережье США:

```
$ export TZ=":ESTEDT"
$ ./strftime
It is 16:16 on Tuesday 28 April 2015
```

Как и ожидалось, локальное время меньше на пять часов. Мы уже видели такое поведение прежде. Теперь попробуем изменить локаль:

```
$ export LC_ALL=fr_FR.utf8
$ ./strftime
It is 16:19 on mardi 28 avril 2015
```

Названия дней недели и месяцев стали отображаться по-французски (кстати, я не подумал, что жестко кодировать "It is" и "on" в строке было не слишком хорошей идеей!). Здесь у нас странная смесь — отображается время восточного побережья США, но по-французски. В качестве последнего теста переместимся в центральноевропейское время:

```
$ TZ=»:CET"
$ ./strftime
It is 22:19 on mardi 28 avril 2015
```

Проблема 2038 года

Возможно, вы помните «проблему 2000 года», вызванную тем, что на значение года в компьютерах и базах данных отводилось только две цифры, и 1999 год должен был перейти обратно в 1900-й, что вызвало бы потенциальный хаос и внезапный спрос на программистов Cobol. Ну, а в 32-битных системах Linux, где время **time_t** хранится в виде 32-битного знакового целого и охватывает интервал от пятницы 13 декабря 1901 г. 20:45:52 до вторника 19 января 2038 г., мы сталкиваемся с проблемой 2038 года. Кстати, эти даты я вычислил не в уме, а написав программу:

```
#include <limits.h>
#include <stdio.h>
#include <time.h>
main()
{
  time_t t;
  t = INT_MIN;
  printf("самая ранняя: %s\n", ctime(&t));
  t = INT_MAX;
  printf(«самая поздняя: %s\n", ctime(&t));
}
```

Возможно, к тому времени появятся какие-то расширенные 32-битные системы Linux или базы данных, которые хранят **time_t** в виде 32-битного поля. Но большинство систем Linux в 2038 году будут как минимум 64-битными. Мы попробовали вычислить максимально возможное 64-битное время в **time_t**, изменив INT_MAX на LONG_MAX в программе выше, но **ctime()** струхнула и вернула нулевой указатель. Пришлось нам посчитать это в столбик — и мы узнали, что тогда мы получим еще примерно 250 млрд лет.

Оставшегося места нам хватит как раз на короткое упоминание системного вызова **times()**, который возвращает количество времени, потраченное выполняющимся процессом. Она возвращает структуру **tm**, которая выглядит так:

```
struct tms {
  clock_t tms_utime; /* пользовательское время */
  clock_t tms_stime; /* системное время */
  clock_t tms_cutime; /* пользовательское время дочерних процессов */
  clock_t tms_cstime; /* системное время дочерних процессов */
};
```

Вы видите, что время разбивается на пользовательское и системное время, т.е. на время, потраченное на выполнение в «пространстве пользователя», и время, потраченное на выполнение в ядре. Тактовая частота, используемая для этих измерений, обычно равна 100 Гц; если вы хотите узнать реальную частоту в своей системе, воспользуйтесь **sysconf(_SC_CLK_TCK)**. Для простой демонстрации загрузите небольшую программку отсюда: http://linuxformat.com/files/lxf199_sysadmin.c. Она включает много вызовов функции **sqrt()**, которая, как мы ожидаем, должна быть полностью в пространстве пользователя, так как ядро не занимается вычислением квадратных корней. На нашем компьютере мы получили следующие результаты:

```
$ ./timer
utime = 101
stime = 0
cutime = 0
cstime = 0
```

Как мы и ожидали, все время было потрачено в пространстве пользователя. Оба времени для дочерних процессов, разумеется, равны нулю, так как дочерних процессов мы не создавали. В качестве последнего эксперимента закомментируем цикл, который вызывает функцию **sqrt()**, и раскомментируем цикл, который вызывает **write()**. Так как **write()** — системный вызов, он будет выполняться в ядре. С этим изменением наше время выглядит так:

```
$ ./timer
utime = 19
stime = 147
cutime = 0
cstime = 0
```

Как и ожидалось, теперь выполнение по большей части происходит в ядре. На этом всё. В следующем месяце мы поговорим о процессах, уделив особое внимание ключевым системным вызовам **fork()** и **exec()**. Мы также рассмотрим с точки зрения программиста один из самых почтенных механизмов межпроцессной коммуникации — каналы. LXF

Скорая помощь

Аппаратные часы. В компьютере есть аппаратные часы, которые работают независимо от системных часов, поддерживаемых ядром Linux. С помощью команды **hwclock** можно прочесть время аппаратных часов и установить системные часы по нему, или наоборот.

Разбор даты и времени

С помощью функции **strptime()** можно преобразовать дату и время из строкового представления в «разобранное» время. Грубо говоря, эта функция обратна **strftime()**. Она использует коды формата (похожие на те, что использует **strftime**) для описания ожидаемого ввода; все, кто в прошлом имел неприятность встречаться с функциями **scanf()** или **fscanf()**, это оценят.

Например, вот маленькая программа, которая принимает дату (в формате ДД/ММ/ГГГГ) с командной строки и сообщает, какой день недели это был.

```
/* Вычисление дня недели */
#include <stdio.h>
#include <time.h>
int main(int argc, char *argv[])
{
  char *weekday[] = {
    "воскресенье", "понедельник", "вторник",
    "среда", "четверг",
    "пятница", "суббота"};
  struct tm bdtime;
  strptime(argv[1], "%d/%m/%Y", &bdtime);
  printf("Это %s\n", weekday[bdtime.tm_wday]);
}
```

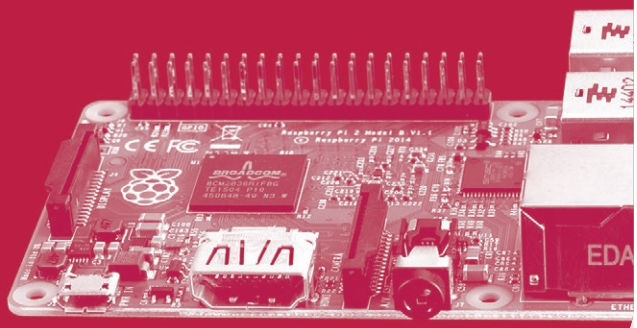
С помощью этой программы можно узнать, на который день недели выпадает Рождество в этом году:

```
$ ./day-of-week 25/12/2015
Это пятница
```

Ну вот, теперь вы знаете. Конечно, чтобы успеть приготовить традиционную рождественскую индейку, ее нужно достать из холодильника как минимум в четверг.

LINUX FORMAT Pi

Пользователям



Получите свою порцию смачнейших новостей, обзоров и руководств по Raspberry Pi.

Кэрри-Энн Филбин рассказывает о последних новостях от фонда Raspberry Pi.

Добро пожаловать...

Образовательная команда Raspberry Pi Foundation, которую я возглавляю, любит знакомить преподавателей с Raspberry Pi, и эта команда (включающая Бена Наттала [Ben Nuttall], Клива Била [Clive Beale], Джеймса Робинсона [James Robinson] и Дэйва Хонесса [Dave Honess]) всегда посещает конференции и мероприятия для обмена опытом. В прошлом году мы запустили программу обучения учителей, под названием Picademy, и пригласили преподавателей со всего мира в штаб-квартиру Pi для веселых практических занятий.

Программа была невероятно популярна, и за 8 уникальных мероприятий мы подготовили около 200 преподавателей. Отклик всегда был хорошим, и с момента появления программы в апреле 2014 г. в команду вошли такие уважаемые члены сообщества, как Лес Паундер и Мартин О'Хэнлон [Martin O'Hanlon].

Рост Piacademy

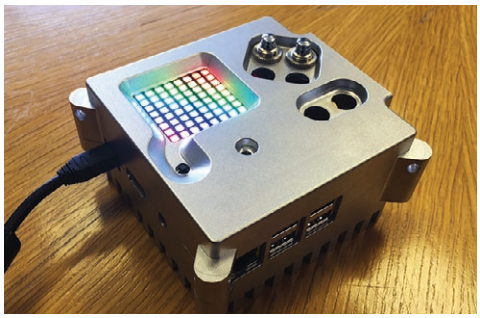
Pi Foundation преследует важную цель. Мы хотим обучить сотни тысяч преподавателей, и наших возможностей для этого не хватит, даже будь мы суперзвездами. Но благодаря щедрой поддержке Google.org мы думаем, что избрали верное направление с новой программой для преподавателей Picademy@Google, запущенной в Лидсе, Великобритания. Она позволяет учителям от начальной до старшей школ посетить семинар по Raspberry Pi в Google Digital Garage, спецпространстве в Leeds Docks. Это совсем бесплатно; гараж будет открыт 6 месяцев. Запись — на http://bit.ly/PicademyAtGoogle.

Astro Pi готов ко взлету

Pi тоскует по космосу, но для начала ему нужен специальный корпус.

Если вы все еще думаете, что в Pi нет ничего выдающегося, ваше мнение может изменить тот факт, что Pi будет доставлен на борт МКС для экспериментов.

Но просто так махнуть в космос с Pi в кармане нельзя — его нужно облачить в сверкающий космический корпус. Pi Foundation обнародовала детали и выложила снимки эксклюзивного алюминиевого корпуса, в котором Pi отправится в полет, и это серьезная штучка. На МКС нет естественного движения воздуха, и корпус должен тщательно охлаждать Pi, так как по правилам МКС ни одна из поверхностей на борту не может нагреваться выше 45 градусов. Корпус изготовлен из сплошного алюминиевого блока, и мы подробно расскажем о нем в **LXF200**. Больше снимков — на http://bit.ly/AstroPiCase.



Встречайте Modulo

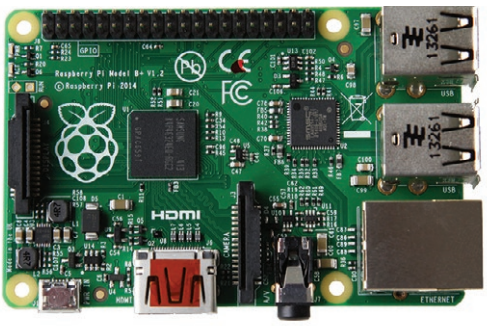
Система модульного расширения для Pi.

Встречайте Modulo, представителя следующего поколения плат расширения Pi. Это удобное решение, созданное на общественные средства, представляет собой набор компактных сменных электронных модулей, позволяющих без труда создавать электронные устройства на основе Raspberry Pi. Для этой системы на основе Arduino есть разнообразные модули, от ЖК-дисплеев до джойстиков. Уже давно собравший все средства запрос на Kickstarter можно посмотреть здесь: http://bit.ly/LXFmodulo.

Скидка на Pi B+

Дефляция дошла и до Pi — скидки на старые модели.

При миллионе уже проданных Pi 2 и поддержке предыдущих моделей со стороны Pi Foundation неудивительно, что исходная модель Raspberry Pi B+ — которая продолжала продаваться по $35, как и Pi 2 — должна была получить скидку. Вероятно, оптимизация изготовления Pi 2 вызвала лавинообразный эффект, что снизило стоимость модели B+ до $25. Подробности здесь: http://bit.ly/PiPriceCut.



Pipsta

Микропринтер для Raspberry Pi, да что он может? **Лес Паундер** исследует.

Вкратце

» **Маленький термопринтер для быстрой и бесшумной печати с Raspberry Pi. В Pipsta есть надежная библиотека Python, которую легко интегрировать в свои проекты, и самосборный корпус для защиты Pi.**

Принтеры не назовешь самыми впечатляющими из устройств, кроме, конечно, 3D-принтеров, которые сейчас вошли в моду и наращивают рынок своих потребителей. Маленький принтер, подключенный к Raspberry Pi — это необычно, но что он может предложить?

Pipsta — принтер для всех моделей Raspberry Pi, в комплекте с ним поставляется набор, который собирается примерно за час, но не требует пайки, так как содержит готовые электронные схемы. Pipsta состоит из трех основных компонентов. Принтер — обычный термопринтер, в который вставляются рулончики специальной термобумаги. Под воздействием тепла на этой бумаге печатаются изображения и текст. Под блоком печати расположена плата контроллера mini-USB, порт которой подключается к USB-порту Raspberry Pi. У контроллера также есть собственный источник питания, подключаемый в передней части блока. Последний компонент — акриловый корпус; он окружает принтер и состоит из шести частей, которые просто стыкуются друг с другом.

Raspberry Pi устанавливается в нижней части корпуса, и к модулю печати подключается один из нескольких проводников заземления, который служит для заземления печатающей головки. Принтеру Pipsta необходимо собственное питание, так как входы/выходы общего назначения Raspberry Pi (GPIO) не обладают требуемой мощностью.

После установки Raspberry Pi в корпус порты HDMI, USB, Ethernet и питания останутся по-прежнему доступными. Доступ к портам GPIO тоже возможен, но для этого придется сделать прорезь в корпусе. То же относится к слоту SD-карты, но, к счастью, корпус для более удобного доступа можно снять. Также можно подключить камеру Raspberry Pi через корпус через прорезь на задней панели для разъемов USB и Ethernet.

› **Pipsta — принтер маленький, но вырастает в ваших глазах, когда дело доходит до печати любимых фотографий.**

Малый отпечаток

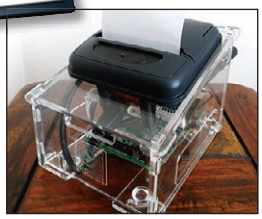
С принтером поставляется подробное руководство, где описаны все этапы установки, которое дублируется в онлайн-ресурсе на Bitbucket. Установка ПО нам показалась простой, но есть пара изменений конфигурации, способных озадачить новичка, в частности, отключение стандартного принтера Linux и разрешение всем пользователям печатать на Pipsta. После установки *pip*, менеджера пакетов Python, понадобится установить зависимости Pipsta, которые поддерживают обработку изображений и создание QR-кодов. Наконец, можно зазрузить программы и примеры Python и извлечь их в домашний каталог.

Pipsta программируется на языке Python, и менеджер пакетов *pip*, используемый во время установки, работает с Python 2.7. Команда Pipsta говорит, что сейчас они сосредоточены на Python 2.7, но в ближайшем будущем перейдут на Python 3.

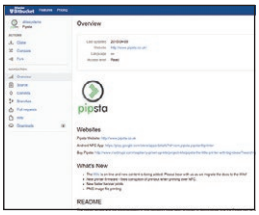
Чтобы попробовать, как все работает, мы решили пройтись по первому из предложенных примеров, под названием Basic Print. Он выполняет тестовую печать традиционной фразы "Hello World from Pipsta", и у нас сработал хорошо. В каталоге есть и другие примеры, и среди них наше внимание привлек Image Print. Он демонстрирует печать файлов PNG в оттенках серого с высокой детальностью. Нам удалось воспроизвести логотип Pi и фотографию с ним, и к нашему удивлению, фотография напечаталась даже лучше, чем логотип.

Итак, для чего же нужен принтер Pipsta? Им уже пользуются программы-предсказатели погоды и судьбы, а также некоторые приложения Twitter. Вы на нем можете напечатать бейджики для очередного Raspberry Jam. Как всегда с Pi, вас ограничивает только воображение. LXF

Свойства навскидку



Удобная сборка
Акриловый корпус для принтера Pipsta удобен в сборке, и с ним у вас сохраняется доступ ко всем портам.



Масса примеров
На сайте Pipsta вы можете отыскать массу примеров — от классического вывода "Hello World" до печати QR-кодов.

LINUX FORMAT Вердикт

Pipsta

Разработчик: Able Systems
Сайт: www.pipsta.co.uk
Цена: £84

Функциональность	8/10
Производительность	7/10
Простота в работе	7/10
Оправданность цены	7/10

» Хорошая игрушка для школ и клубов по программированию, которые хотят объединить вычислительные ресурсы с физическими проектами.

Рейтинг 7/10

GPIO: Метнем игральную кость

Лес Паундер с презрением отвергает старомодный игровой кубик, потому что у него есть имитатор кубика на Pi.

Наш эксперт

Лес Паундер — член образовательной команды Raspberry Pi Foundation. Он создает безумные проекты на Pi и Arduino и пишет о своих открытиях в блоге: http://bigl.es.

Скорая помощь

Макетные платы — это прототипы для создания электронных схем без пайки. С ними легко собрать и протестировать проект, а потом снова примениться в другом проекте. Макеты бывают разных размеров, от небольших — с кирпичик «Лего» — до формата A4. Цена их зависит от размера и спецификации (от £2 до £30).

Скромный кубик уже много лет помогает вносить элемент случайности в настольные игры по всему миру. Как же сделать старый добрый кубик современным? На этом уроке мы создадим цифровую версию кубика из Raspberry Pi и других компонентов.

Для этого проекта вам понадобится Raspberry Pi (любой модели) со свежей версией Raspbian. Из дополнительных компонентов требуются шесть светодиодов, шесть резисторов сопротивлением 220 Ом (красный, красный, коричневый, золотой), восемь соединительных проводников «мама–папа», семь соединительных проводников «папа–папа» и макетная плата на 400 отверстий.

Код всех примеров, которые мы рассмотрим, можно найти на http://bit.ly/LXF199_Pi_Beginner. Для начала подсоедините все компоненты нашего кубика к Raspberry Pi, который должен быть выключен. К макету компоненты подключаются согласно схеме, которую можно загрузить с http://bit.ly/LXF_199_Dice_Diagram.

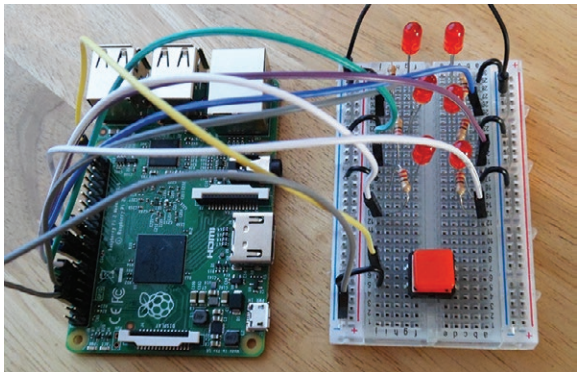
Настройка ПО

Для этого проекта мы воспользуемся Python 3 и редактором *IDLE 3*. Откройте *IDLE 3* с префиксом **sudo**, чтобы нашему проекту были доступны входы/выходы общего назначения. Откройте *LXTerminal* и наберите следующую команду (не забыв в конце нажать Enter):

```
$ sudo idle3 &
```

Открыв *IDLE*, можно приступить к программированию проекта. Выберите File > New [Файл > Создать], чтобы открыть новое окно, а в нем выберите File > Save [Файл > Сохранить], чтобы сохранить проект. Это хорошая практика, которая позволит избежать случайной потери данных.

Начнем писать программу с импортирования библиотек, благодаря которым наш проект сможет работать. Библиотеки — это внешние репозитории кода, которые можно импортировать в наш проект для расширения его свойств и возможностей. На данном уроке нам понадобятся три библиотеки: RPi.GPIO для доступа к GPIO, функция sleep из библиотеки time и функция randint из библиотеки random.



› **Схема необходимых нам соединений между Pi и макетом доступна здесь: https://github.com/lesp/LXF_199_Dice/archive/master.zip.**

```
import RPi.GPIO as GPIO
from time import sleep
from random import randint
```

Затем надо выбрать схему нумерации контактов GPIO. Здесь мы воспользуемся схемой Broadcom — это стандарт, поддерживаемый Raspberry Pi Foundation. Нумерация в этой схеме не является хронологической, контакты не нумеруются последовательно от 1 до 40, но это самая популярная и устоявшаяся схема нумерации контактов; она берет начало от схемы нумерации контактов микросхемы Broadcom, на которой работает Raspberry Pi.

```
GPIO.setmode(GPIO.BCM)
```

Теперь перейдем к следующему участку кода, в которой используются переменные с номерами контактов GPIO для светодиодов и кнопки. Переменные — это способ хранения данных,

Импортируйте эффективно

В этом проекте мы воспользовались функцией randint для получения случайного числа, которое будет результатом броска кубика. Но вместо импорта всей библиотеки random мы импортировали только одну функцию. Почему? Ну, импортировать только то, что нам нужно, немного эффективнее.

Конечно, можно импортировать и всю библиотеку random, но это увеличивает объем памяти, потребляемой проектом; к тому же тогда нам придется предварять вызов каждой функции библиотеки именем библиотеки. Например, вместо

```
randint()
```

при импорте всей библиотеки пришлось бы писать:

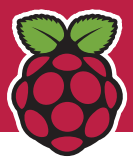
```
random.randint()
```

Это означает, что для получения того же результата придется набрать чуть больше кода. Существует и другой способ импорта библиотеки полностью, при котором вам не придется предварять вызов каждой библиотечной функции именем библиотеки.

Вам поможет шаблон при первом импорте библиотеки в начале проекта.

```
from random import *
```

Это чуть менее эффективный способ импорта только тех функций, которые нам нужны, но так с ними проще работать.



который позволяет избежать жесткого кодирования значений в проекте: например, если я пользуюсь контактом 14, но захотел изменить его на 15, то мне пришлось бы проверять весь код строка за строкой и повсюду изменять 14 на 15, что заняло бы довольно много времени. Создав переменную и присвоив ей имя, мы можем обращаться к ней по имени; при этом Python смотрит на имя переменной и извлекает ее значение. Решив изменить значение, я сделаю это один раз — там, где объявлял переменную. В простом коде ниже мы создаем шесть переменных с именами от d1 до d6 для нашего кубика и задаем переменную для контакта, к которому будет подключена кнопка.

```
d1 = 14
d2 = 15
d3 = 18
d4 = 23
d5 = 24
d6 = 25
d2 = 15
d3 = 18
d4 = 23
d5 = 24
d6 = 25
button = 21
```

Объявив переменные, настроим входы и выходы для нашего проекта. Основное средство ввода — кнопка, подключенная к контакту 21. Мы устанавливаем его в HIGH (т.е. в «единицу») с помощью внутренних подтягивающих/понижающих резисторов. Все светодиоды d1–d6 мы подключаем к выходам:

```
GPIO.setup(button,GPIO.IN, pull_up_down = GPIO.PUD_UP)
GPIO.setup(d1,GPIO.OUT)
GPIO.setup(d2,GPIO.OUT)
GPIO.setup(d3,GPIO.OUT)
GPIO.setup(d4,GPIO.OUT)
GPIO.setup(d5,GPIO.OUT)
GPIO.setup(d6,GPIO.OUT)
```

Следующий шаг — создать функцию для обработки действий кубика.

```
def dice():
  for i in range(5):
    roll = randint(1,6)
  print(roll)
```

Мы начинаем с указания имени функции, в данном случае **dice**. Как видите, строки под заголовком функции имеют отступы. Так в Python обозначается код, принадлежащий функции. Следующий шаг — обработать бросок виртуального кубика. Для этого используется цикл, выполняемый пять раз. В этом цикле мы создаем переменную **roll** и присваиваем ей случайное число от 1 до 6 — фактически это кубик с шестью гранями. Затем мы прерываем цикл и печатаем выбранное число.

Следующий фрагмент кода под названием «Суматоха [Razzle Dazzle]» (который я не привел здесь из-за экономии места — он доступен на **LXFDVD** или по ссылке http://bit.ly/LXF199_Pi_Beginner) служит единственной цели — сделать бросок кубика более эффектным. Цикл проходит три раза, включая и выключая четные и нечетные светодиоды. Включение светодиода выполняется функцией **GPIO.output(d1,1)**, и единица означает «включить». Примечание: Вместо единицы можно указать TRUE. Для выключения светодиода замените 1 на 0 или проставьте FALSE:

```
if roll == 1:
  GPIO.output(d1,1)
  sleep(3)
  GPIO.output(d1,0)
```

Светодиоды

У светодиода две ножки: длинная — анод, положительный (+) контакт светодиода; короткая — катод, земля (-), которая подключается к контакту «земля [GND]» Raspberry Pi.

Мы подключаем светодиоды так, что аноды подключаются к контактам GPIO через резистор 220 Ом, который используется для снижения тока, поступающего на светодиод; это защищает светодиод и продлевает срок его службы. Катод каждого светодиода подключается к шине «земля» макета, а та подключается к одному из контактов «земля» Raspberry Pi.

Последняя часть функции **dice()** — набор условий if...elseif, которые управляют тем, что происходит при выборе определенного номера:

```
elif roll == 2:
  GPIO.output(d1,1)
  GPIO.output(d2,1)
  sleep(3)
  GPIO.output(d1,0)
  GPIO.output(d2,0)
```

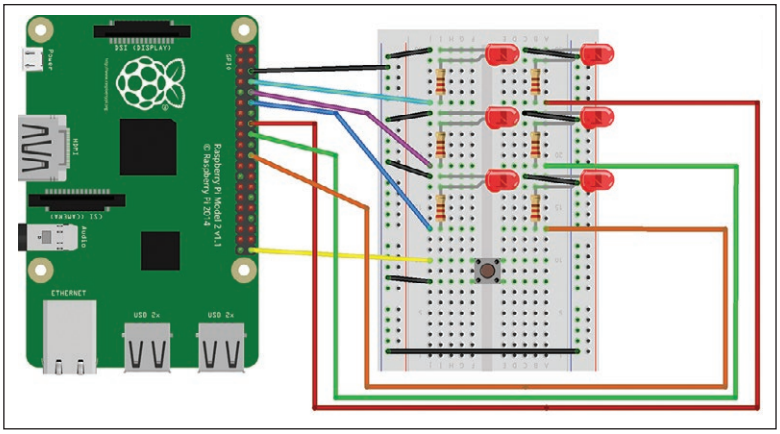
В Python else известно как elif. Рассмотрим первое условие. Если случайное число, записанное в переменной roll, равно 1, то на три секунды загорается первый светодиод, затем он гаснет. Если условие ложно, проверяется следующее условие, и так до тех пор, пока не будет найдено истинное условие и не будет предпринято соответствующее действие. Для чисел от 2 до 6 мы используем набор аналогичных условий else if (elif), изменяя только число, которое сравнивается с переменной roll. Если условие истинно, загорается светодиод с правильным номером.

Наконец, мы создаем бесконечный цикл, проверяющий нажатие кнопки. После нажатия кнопки ее состояние изменяется с «единицы [HIGH]» на «ноль [LOW]», и при этом изменении состояния срабатывает функция **dice()**:

```
while True:
  if GPIO.input(button) == False:
    dice()
```

Обязательно сохраните свою работу и выберите Run > Run Module [Запуск > Запустить модуль] для запуска проекта. Подождите несколько секунд, пока проект загрузится, и нажмите кнопку на макете. Вы должны увидеть небольшую анимацию, после чего светодиоды покажут случайное число.

Поздравляем — у вас получилась цифровая игральная кость! LXF



› Мы пользуемся макетом с шинами по бокам, что позволяет нам разделить один контакт «земля» на Pi на несколько точек на макете.

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

CamJam: Создаем монитор погоды

Лес Паундер играет в городского ученого и отслеживает погоду, не выходя из дома, с помощью Python и Raspberry Pi.

Наш эксперт

Лес Паундер специализируется в обучении британских преподавателей Raspberry Pi. Он ведет блог на http://bigl.es.

Raspberry Pi — отличная платформа для научных экспериментов, так как с ее помощью можно собирать данные и обрабатывать их; но с чего начать? На этом уроке мы создадим датчик температуры на Raspberry Pi, который будет брать замеры с датчика и сравнивать их с данными онлайн-сервиса погоды. Затем по этим данным строится график, отражающий изменение погоды с течением времени.

Для данного проекта нам понадобятся: Raspberry Pi (любой модели) с рабочим Wi-Fi-подключением, набор CamJam EduKit (№ 2), датчик температуры DS18B20, резистор 4,7 кОм, три соединительных проводника «мама–папа» и макетная плата. Для получения данных о погоде воспользуемся Open Weather Map. Этот бесплатный API предоставляет подробные данные о погоде по всему миру и текущую погоду в заданном месте, а также прогнозы и архивную информацию. Для использования API нужно получить ключ с сайта http://openweathermap.org/api. Помните, что ключ нужно хранить в безопасном месте: он связан лично с вами и не должен попасть к другим. Мы воспользуемся ключом, чтобы убедиться, что мы можем загрузить данные о погоде для своего местоположения; но сначала надо установить модуль pyowm, чтобы Python мог общаться с сайтом OpenWeatherMap. Для установки модуля воспользуемся *pip*, менеджером пакетов Python. Он уже установлен на Raspberry Pi, и чтобы им воспользоваться, откройте *LXTerminal* (щелкнув по иконке на рабочем столе) и наберите следующее:

```
sudo pip-3.2 install pyowm
```

Затем откройте приложение Python 3, выбрав его из меню Programming [Программирование]. Чтобы создать новый файл, выберите File > New [Файл > Создать]. Начнем с импорта модулей.

› Наш проект можно запитать от мобильной батареи и поместить его в подходящее место в саду. Для сохранности устройства поместите его в коробку для бутербродов или пластиковый контейнер.

Pyowm — обертка Python для OpenWeatherMap, а из модуля time мы импортируем функцию sleep, которая используется для управления скоростью выполнения кода далее в проекте:

```
import pyowm
from time import sleep
```

Теперь определим функцию, с помощью которой легко будет получить погоду простой командой вместо набора больших фрагментов кода:

```
def get_weather(n):
```

Начнем с имени функции get_weather. Ее аргументом является “n” в скобках, этот параметр/ключ позволяет изменять местоположение, для которого мы хотим получить погоду. Теперь создадим переменную owm, содержащую наш ключ API:

```
  owm = pyowm.OWM('PASTE YOUR API KEY HERE')
```

Ключ нужно поместить в одиночные кавычки, чтобы Python понял, что это строка. Заметьте: у этой строки есть отступ, который подчеркивает, что она является частью только что созданной функции. Создадим следующую переменную:

```
  observation = owm.weather_at_place((n))
```

На сей раз мы назвали ее observation и применим, чтобы сократить имя функции, которую будем использовать как местоположение, заданное в проекте.

Проверяем погоду

Две следующих переменных сокращают вызовы функций для получения погоды в указанном месте и для получения температуры в градусах Цельсия:

```
  w = observation.get_weather()
  a = (w.get_temperature('celsius'))
```

В последней строке мы выводим сообщение для пользователя функцией print, а данном случае “The current temperature at... [Текущая температура в...]”, к которому добавляется местоположение, заданное в переменной n, после чего приставляется содержимое переменной a. Переменная a весьма непростая, так как по сути это словарь Python, содержащий пары «ключ–значение». В данном случае Python должен найти в словаре ключ “temp” и вернуть его значение.

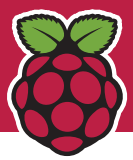
```
print(«Текущая температура в”,(n),“-”,a['temp'])
```

На этом тестовая функция закончена, поэтому сохраните свою работу и выберите Run > Run Module [Запуск > Запустить модуль] из меню IDLE. Откроется оболочка Python, но ничего не произойдет, пока мы не вызовем функцию, передав ей местоположение.

Местоположение можно передать так:

```
get_weather(“London,uk”)
```

Или в виде почтового индекса — он же ZIP-код [британский]:

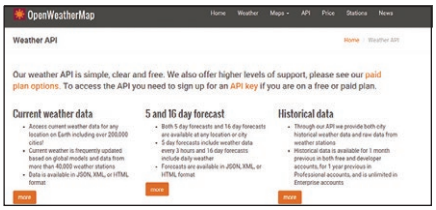


Python и API

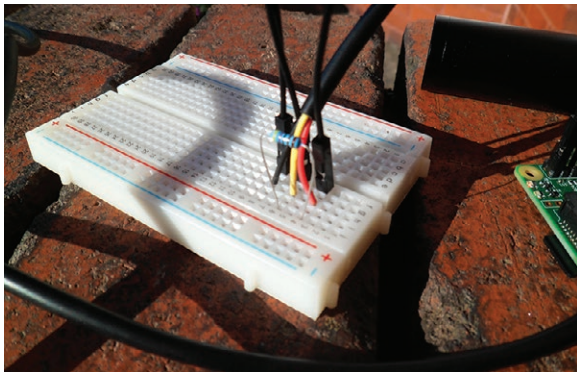
На нашем уроке нам получали данные о погоде от web-сервиса и строили график на основе созданных данных два API (Application Program Interface — программный интерфейс приложения). API — отличная штука, и с ними вы полюбите Python гораздо больше; так, можно изменить API *Plotly* на API *Minecraft* и соорудить мир *Minecraft* на основе данных погоды. Если API OpenWeatherMap сообщает, что у меня идет снег, можно учинить снегопад и в мире *Minecraft*, или даже отобразить температуру на графике блоками из мира *Minecraft*.

Далее опишем еще два популярных API. Первое — API Twitter, доступное в таких модулях Python, как Twython и Tweepy, благодаря которым ваши проекты получат доступ к Twitter. Второе API — SMS-сервис для Raspberry Pi, позволяющий отправлять SMS-сообщения из кода на Python. С ним можно, например, получать SMS-сообщения о температуре в заданное время. Данный API можно найти на www.smspi.co.uk.

Существуют различные API, с которыми можно поэкспериментировать и получить удивительные результаты, включив их в свой проект: например, построить собственную погодную станцию, которая будет писать в Twitter и отправлять SMS, чтобы вы следили за погодой.



› API — быстрый, но весьма продуктивный способ расширения возможностей вашего кода. Многие открытые проекты используют API, чтобы помочь своему сообществу создавать удивительные проекты.



› Мы создаем прототип проекта на макетной плате, но если вы хотите создать постоянную платформу, можете воспользоваться перфорированной или печатной платой, чтобы всё выглядело профессиональнее.

```
get_weather("W12 7LJ")
```

В оболочке должно появиться следующее сообщение:

```
# Текущая температура в Блэкфул,uk - 9.06
```

Мы успешно создали функцию для получения погоды от сервиса OpenWeatherMap. Пора настроить наш датчик температуры.

Настраиваем датчик температуры

Датчик температуры играет важную роль в этом проекте. Он фиксирует температуру в месте, где мы находимся, чтобы затем сравнить ее с температурой, предоставленной OpenWeatherMap. Мы используем датчик Dallas DS18B20, с виду очень похожий на транзистор. Наш датчик защищен водонепроницаемой оболочкой и металлической крышкой. Это означает, что датчик можно погрузить в жидкость и измерить ее температуру — достаточно только соблюдать осторожность, если жидкость горячая. На другом конце датчика три проводника: красный подключается к контакту 3,3 В нашего Pi, черный — к контакту «земля [GND]», а желтый — последовательная линия, подключаемая к контакту 4 (согласно нумерации Broadcom) нашего Pi. Для безопасного подключения проводников в Pi нам понадобится макетная плата, три соединительных проводника «папа–мама» и резистор сопротивлением 4,7 кОм, подключенный между входом 3,3 В и последовательной линией (см. схему на стр. 86).

Подсоединив компоненты, включите Pi и загрузите рабочий стол. Чтобы воспользоваться датчиком, надо настроить Pi. Откройте *LXTerminal* и наберите команду

```
sudo leafpad /boot/config.txt
```

Откроется конфигурационнй файл загрузки. Перейдите в конец файла и добавьте следующее:

```
dtoverlay=w1-gpio
```

Сохраните файл и перезагрузите Raspberry Pi. Теперь надо проверить, что Pi видит датчик. Откройте *LXTerminal* и выполните следующее:

```
$ sudo modprobe w1-gpio
$ sudo modprobe w1-therm
```

Эти две команды загрузят модули ядра, необходимые для использования датчика. Датчик прочтет температуру и запишет данные в файл, так что теперь надо перейти к файлу и прочесть его содержимое:

```
$ cd /sys/bus/w1/devices
```

В этом каталоге надо найти подкаталог, начинающийся с 28-, поэтому начнем с просмотра содержимого каталога.

```
$ ls
```

Обнаружив каталог, перейдите в него командой **cd**. Для этого начните печатать имя каталога, затем нажмите Tab для автоматического дополнения:

```
$ cd 28-
```

Внутри этого каталога надо заглянуть в один файл, и для этого воспользуемся командой cat.

```
$ cat w1_slave
```

Эта команда выведет массу кода; найдите в нем параметр t= . В нашем случае он был равен 17125, который, если разделить его на 1000, даст нам 17,125 градуса по Цельсию, что доказывает, что датчик работает правильно. Теперь воспользуемся датчиком для записи температуры в том месте, где мы находимся, для чего воспользуемся следующим кодом на Python.

Начнем, как всегда, с импорта модулей, необходимых для выполнения проекта. В данном случае воспользуемся модулем OS, требуемым для того, чтобы Python мог взаимодействовать с операционной системой. Модуль glob применяется для обработки путей ОС. Библиотека *time* используется для управления скоростью выполнения проекта:

```
import os
import glob
import time
```

Теперь при помощи функции system из модуля OS загрузим в ядро два драйвера: один для GPIO, а другой для датчика температуры.

```
os.system('modprobe w1-gpio')
os.system('modprobe w1-therm')
```

»

Скорая помощь

Мы подключаем резистор сопротивлением 4,7 кОм между напряжением 3 В и проводником данных датчика. Этот резистор «подтягивает» проводник данных и сохраняет подключение стабильным.

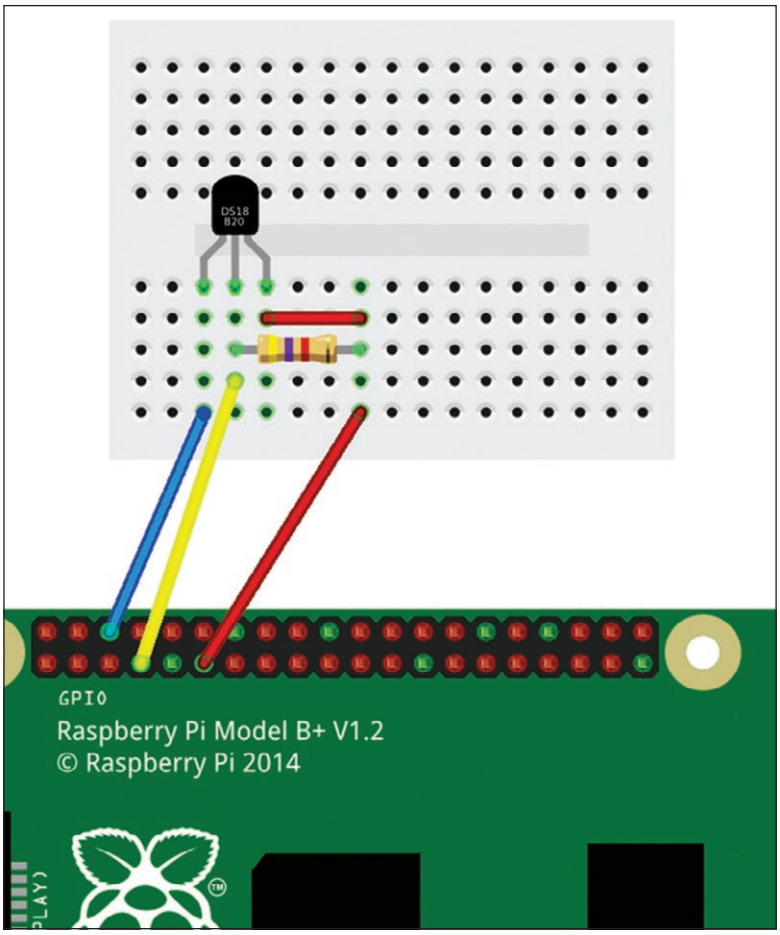
» **Подпишитесь на печатную или электронную версии** на www.linuxformat.ru/subscribe!

Затем мы создаем три переменных. Первая содержит местоположение датчика температуры в файловой системе. Вторая использует glob для поиска каталога с данными датчика. Наконец, последняя переменная нужна для добавления имени файла, содержащего данные о температуре, к предыдущей переменной:

```
base_dir = '/sys/bus/w1/devices/'
device_folder = glob.glob(base_dir + '28*')[0]
device_file = device_folder + '/w1_slave'
```

Теперь перейдем к двум особым строкам кода. В первой создается глобальная переменная **temp_c**, которой можно пользоваться внутри и за пределами созданной ранее функции. В следующей строке создается пустой список. Списки похожи на массивы в других языках, в которых можно сохранять данные о температуре.

```
global temp_c
sensor = []
```



› **Мы создали репозиторий на Github, содержащий весь код этого проекта: https://github.com/lesp/LXF_199_Temperature_Sensor/archive/master.zip.**

Далее создадим функцию, которая прочтет температуру, хранящуюся в переменной **device_file**, созданной нами ранее. Мы открываем файл в режиме только для чтения, считываем информацию, закрываем файл и выводим информацию в оболочку:

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

Следующая функция считывает «сырые» данные о температуре с датчика и обрабатывает их, удаляя ненужную информацию. Затем результат делится на 1000, оставляя нам корректное значение температуры:

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

Теперь воспользуемся циклом for, который проходит пять раз, каждый раз выводя температуру с датчика, добавляя данные о температуре к списку датчиков и затем выводя список данных, потом ожидает в течение 3600 секунд (одного часа) и повторяет процесс:

```
for i in range(5):
  print(read_temp())
  sensor.append(temp_c)
  print(sensor)
  time.sleep(3600)
```

На этом с кодом покончено. Сохраните и запустите файл, выбрав Run > Run Module [Запуск > Запустить модуль]. Для тестирования можете изменить значение последней паузы на более короткий интервал.

Настройка Plotly

Теперь нужно установить графическую составляющую проекта, которая поддерживается *Plotly*, графическое решение для web-интерфейса с прекрасным API для Python. Вам понадобится создать учетную запись *Plotly* (https://plot.ly) и получить ключ API.

Получив ключ, откройте LXTerminal на Raspberry Pi и установите модули:

```
$ sudo pip-3.2 install plotly
```

Затем надо сообщить Plotly свое имя пользователя и ключ API. В *LXTerminal* наберите следующее (в одной непрерывной строке):

Покупаем набор

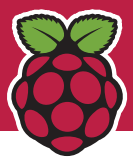
На нашем уроке мы пользовались готовым набором компонентов из набора CamJam Edukit No.2 под названием Sensors [Датчики]. У оптовиков, например, на The Pi Hut (http://thepihut.com), этот набор продается за £7, и это экономичный способ приобрести все компоненты, которые необходимы для данного проекта.

Стоимость DS18B20 — £4,5 в стандартной версии, но в наборе CamJam есть специальная влагостойкая версия, присоединенная к длинному проводнику. Такой датчик можно высунуть из окна, оставив Raspberry Pi внутри. Кроме DS18B20, набор включает: макетную плату, резисторы, проводники и другие компоненты, которые пригодятся в других проектах. (В Интернете можно накопать множество онлайн-схем макетов).

На рынке есть и другие датчики температуры, например, популярные DHT11 и DHT22, но они подключаются к Raspberry Pi иначе. DHT11/22 отправляют данные на плату через аналоговый выход.

Для платы Arduino достаточно просто подключить компоненты к аналоговым портам, но у Raspberry Pi таковых не предусмотрено. И чтобы Pi мог прочесть данные с этих датчиков, вам понадобится АЦП.

Мы обнаружили, что DS18B20 гораздо удобнее и дешевле. Набор CamJam, кроме того, упакован в удобный жестяной корпус, и вы можете безопасно и надежно хранить все свои электронные компоненты в одном месте.



```
$ python -c "import plotly; plotly.tools.set_credentials_
file(username='ВАШЕ ИМЯ', api_key='API KEY', stream_
ids=['STREAM ID', 'STREAM ID])"
```

Всю эту информацию можно найти на https://plot.ly/python/getting-started после создания учетной записи.

Сводим все это вместе

Мы создали основы для нашего проекта, и теперь пора объединить всё это вместе. Обязательно запускайте IDLE с правами sudo, это можно сделать в *LXTerminal* командой

```
$ sudo idle3
```

Начнем с импорта модулей для операционной системы, путей, API погоды и графиков:

```
import os
import glob
import pyowm
from time import sleep
import plotly.plotly as py
from plotly.graph_objs import *
```

Затем сохраним API-ключи для OpenWeatherMap и Plotly в централизованном месте.

```
py.sign_in('USERNAME', 'API KEY')
key = ('API KEY')
```

Загрузим модули ядра для датчика:

```
os.system('modprobe w1-gpio')
os.system('modprobe w1-therm')
```

Мы создаем переменные для хранения названия устройства датчика и данных о температуре, а затем создаем три списка. Наша ось X содержит периоды, для которых записываются данные: например, с 10 утра до 2 часов дня. Y — пустой список для хранения данных о температуре, которые мы будем получать из OpenWeatherMap, а список sensor пуст и используется для хранения данных нашего датчика:

```
base_dir = '/sys/bus/w1/devices/'
device_folder = glob.glob(base_dir + '28*')[0]
device_file = device_folder + '/w1_slave'
global temp_c
x = ['10:00','11:00','12:00','13:00','14:00']
y = []
sensor = []
```

Мы создаем три функции. Первая получает информацию о погоде из OpenWeatherMap и очень похожа на первую часть проекта, единственное различие в последней строке **get_weather(n)**, которая добавляет данные о температуре в список y, созданный ранее.

```
  y.append(a['temp'])
```

Следующая функция, **read_temp_raw()**, остается такой же, как в предыдущем примере, но из **read_temp()** удалено преобразование в шкалу Фаренгейта и добавлена глобальная переменная **temp_c**, которой мы можем пользоваться внутри и вне функции.

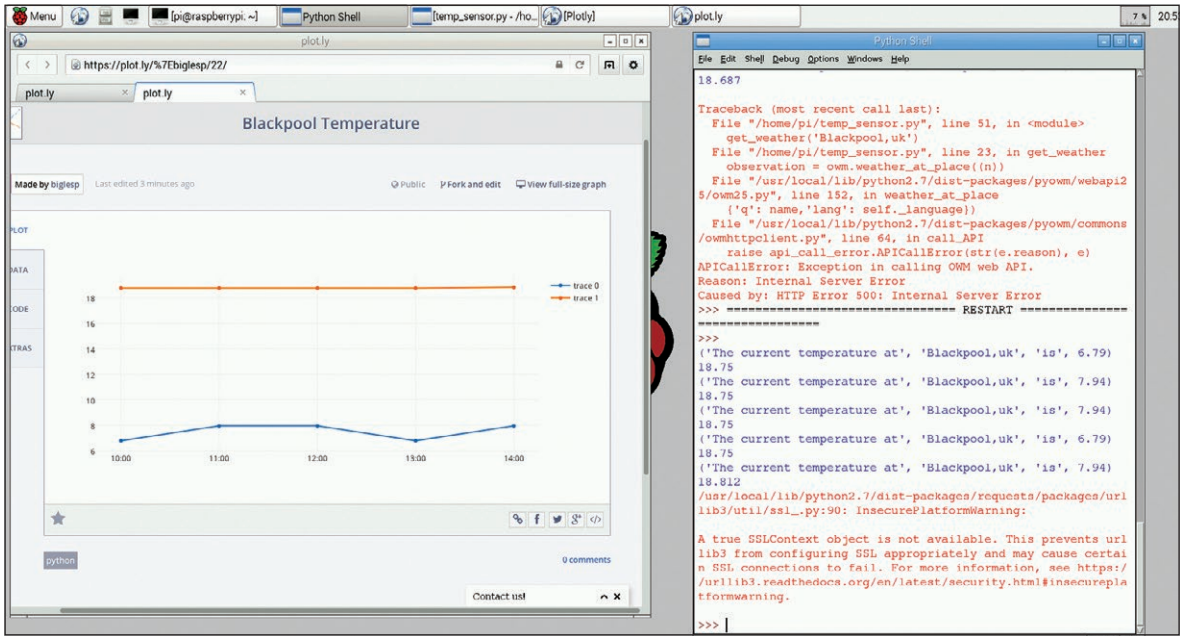
Затем перейдем к использованию кода цикла for, который выполняется пять раз. Он загружает погоду для Блэкпула (Великобритания), а потом считывает температуру через датчик. Значения температуры от датчика и из OpenWeatherMap записываются в соответствующие списки с помощью функции append. И мы выводим температуру датчика, чтобы подтвердить, что они были собраны. Наконец, мы приостанавливаем выполнение программы на 3600 с (один час) и собираем данные пять раз за каждый час:

```
for i in range(5):
  get_weather('Blackpool,uk')
  read_temp()
  sensor.append(temp_c)
  print(temp_c)
  sleep(3600)
```

В последней части кода по нашим данным строятся графики. Графиков два. В первом (Trace0) для оси Y используются данные, полученные с помощью API OpenWeatherMap, для оси X — фиксированные значения времени. Во втором графике (Trace1) для оси X используются те же значения, а для оси Y — данные из списка sensor. Затем данные упаковываются в список и размещаются на сайте https://plot.ly с именем файла "Blackpool Temperature" (его можно изменить):

```
trace0 = Scatter(x=(x),y=(y))
trace1 = Scatter(x=(x),y=(sensor))
data = Data([trace0, trace1])
unique_url = py.plot(data, filename = 'Blackpool Temperature')
```

Покончив с кодом, выберите Run > Run Module [Запуск > Запустить модуль] для запуска программы и сбора данных. По завершении программы автоматически откроется окно браузера с графиком plot.ly. Поздравляем, вы создали проект! LXF



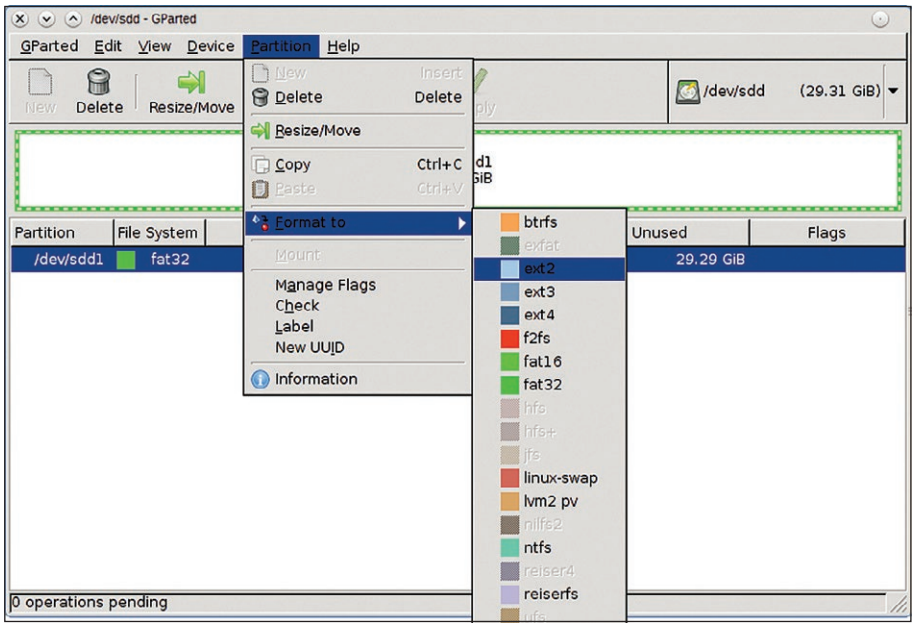
› При запуске код будет выводить извещения в оболочку Python, которая сообщит нам текущую температуру. По завершении цикла будет построен график *Plotly*, который затем откроется в новом окне браузера.

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. Резервное копирование при установке Mint
2. Создание множества каталогов
3. Установку Dracut
4. Перенаправление электронной почты
5. Обращение со спецсимволами
6. Первый опыт в Linux



› Желая копировать на флешку большие файлы, надо переформатировать ее в ext2 вместо FAT.

1 Обновление Linux Mint

В Я хотел бы обновить свой Mint 14 до версии 17.1 (с DVD LXF196). Мне не удается сделать резервную копию на свой 8-гигабайтный USB-диск Cruzer. Сначала я сделал резервную копию файла, и казалось, что все нормально, но я не могу найти файл на диске. Затем я попытался сделать резервную копию программ и столкнулся с той же проблемой. Для этого достаточно просто скопировать /home на внешний диск? Файл может быть большим, так как на диске C: у меня Windows Vista в виртуальной машине с кое-какими программами.

Колин Баллентайн [Colin Ballantyne]

О Вы не указали, какой программой делаете резервную копию, но в ней должна быть возможность указать расположение копии. Вот простой способ найти пропавший файл, даже если Вы не знаете, как он называется — командой **find**:

```
$ find /media/usbstick -type f -mtime -1
```

Она найдет все файлы, измененные за последний день, и выведет их имена. Если Вам нужен более свежий поиск, скомандуйте

```
$ touch somefile
```

Затем запустите программу резервного копирования и выполните команду

```
$ find /media/usbstick -type f -newer somefile
```

Команда **touch** устанавливает временной штамп указанного файла в текущее время (если файла не существует, команда создает его), затем команда **find** находит все файлы новее указанного. Если Вы по-прежнему не можете найти резервную копию, довольно просто создать ее из терминала командой tar:

```
$ tar cf /media/usbstick/home-backup.tar ~
$ tar czf /media/usbstick/home-backup.tar.gz ~
```

Каждая из этих команд создаст копию домашнего каталога в файле на флешке, но последняя также сожмет данные с помощью *Gzip*. Эта копия может иметь меньший размер в зависимости от типа данных, но выполнение команды потребует больше времени. Распаковать архив обратно в текущий каталог можно с помощью

```
$ tar xf /media/usbstick/home-backup.tar.gz
```

Здесь есть одна потенциальная проблема, и с программой резервного копирования, которой Вы пользуетесь, USB-флешки обычно отформатированы в файловой системе FAT. Максимальный размер одного файла в FAT составляет 4 ГБ, а у Вас он может быть больше. Простой способ это обойти — переформатировать диск с файловой системой ext2. Это можно сделать с помощью стандартной утилиты для управления дисками или с командной строки:

```
$ sudo mke2fs /dev/sdXN
```

где sdXN — Ваш USB-диск. Обязательно укажите правильное имя диска. Чтобы избежать этих проблем в будущем, можно создать отдельный раздел для **/home** при установке Linux Mint 17.1. При последующих установках раздел **/home** можно запретить трогать, и Ваши данные будут сохранены.

Обновление без переустановки в целом возможно, но Вы перескакиваете с версии 14 сразу на 17.1, а обновления обычно безопасны при последовательном переходе от версии к версии. Можно пойти этим путем, но переустановка будет быстрее и чище.

2 Создание каталогов

В Я запускаю Kodi на Raspberry Pi B+ с использованием Synology NAS. Мне на самом деле доставляет удовольствие возможность отображения нескольких рисунков для своих носителей. Моя структура файлов такова:

» Films/A/отдельные фильмы в своих собственных каталогах.

» Films/B/отдельные фильмы в своих собственных каталогах.

» И так далее до Z.

Для отображения дополнительных/альтернативных фоновых изображений надо создать в каждом из каталогов для отдельных фильмов подкаталог под названием extrafanart. Так как у меня сотни фильмов в отдельных подкаталогах, создавать там подкаталоги вручную будет утомительно. Я хотел бы узнать, как создать скрипт или выполнить команду, которые сделают следующее:

1 Просканируют (рекурсивно) эти каталоги и подкаталоги на наличие подкаталога под названием extrafanart.

2 Если подкаталога extrafanart не существует, создадут его.

GeordieJedi, С форумов

О Это очень просто или чуть сложнее в зависимости от точности Вашей структуры каталогов. Если вся Ваша структура каталогов подчиняется схеме Films/Первая_буква/Название_фильма/extrafanart, то все, что Вам нужно —

```
$ for DIR in Films/[A-Z]/* ; do
$ mkdir -p "$DIR/extrafanart"
```

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**.

```
$ done
```

Этот скрипт проходит все каталоги, которые соответствуют шаблону, т. е. все каталоги вида **Films/Первая_буква/Название_фильма**, и пытается создать подкаталог **extrafanart**. Секрет успеха — параметр **-p mkdir**, который делает две вещи. Первая — если указанный каталог уже существует, то команда ничего не сделает, тогда как без **-p mkdir** завершилась бы с ошибкой. Кавычки нужны на случай, если в именах есть пробелы. Вторая особенность **-p** в том, что с ним команда создает все промежуточные каталоги, которые могут понадобиться: так, команда

```
$ mkdir -p foo/bar
```

для пустого каталога сначала создаст **foo**, а затем **foo/bar**.

Если Ваша структура каталогов сложнее, например, с подкаталогами для отдельных частей фильма или эпизодов телепрограмм, нужно построить список каталогов с помощью **find** и затем выполнить **mkdir** для этих каталогов, например:

```
$ find Films -mindepth 2 -type d ! -name extrafanart
-exec mkdir -p "{}/extrafanart" +
```

Параметр **-mindepth** означает, что **find** должна проигнорировать каталог **Films** и все, что находится под ним (начальные буквы). Параметр **-type** ограничивает поиск каталогами, **-name** проверяет соответствие имени каталога, а **!** инвертирует это соответствие. Вы должны проверить, что подкаталог **extrafanart** существует, иначе создадутся каталоги **extrafanart/extrafanart**, **extrafanart/extrafanart/extrafanart** и т. д. Наконец, выполняется команда, следующая за **-exec**, причем **{}** заменяется каждым результатом команды **find**, а **+** замыкает часть с **-exec**. Параметр **-exec** можно использовать с **\;** или **+** . В первом случае команда запускается отдельно для каждого найденного каталога; а параметр **+** ускоряет дело, формируя более длинную команду, которая сразу создает несколько каталогов.

3 Неудача с Dracut

Я с интересом прочел статью Нейла Ботвика о *Dracut* [см. стр. 78 LXF196]. Здорово, подумал я, наконец-то загрузка будет быстрой. Но установка *Dracut* оказалась кошмаром из конфликтующих зависимостей и недостающих пакетов. Насколько я понимаю, единственный способ установить ее — это удалить *initramfs* и, кажется, нескончаемый список других файлов. Пожалуйста, проясните это для меня — я не хочу сдаваться при первой же неудаче, но на данный момент не вижу другого выхода. Отсутствующие зависимости есть у следующих пакетов:

```
dracut : Конфликтует с: initramfs-tools
но 0.103ubuntu4.2 установлен.
Конфликтует с: initramfs-tools:i386
```

который является виртуальным пакетом, и для разрешения этих зависимостей необходимо выполнить следующие действия:

```
Сохраните следующие пакеты в их текущей
версии:
1) dracut [не установлен]
```

dave_try1st, с форумов

Пакет *initramfs-tools* содержит программы/скрипты, используемые в Linux Mint для генерации файлов *initramfs* и различными другими пакетами для повторной сборки *initramfs* после установки определенных обновлений. Разработчики обозначили это как конфликт с *Dracut*, вероятно, потому, что *Dracut* может перезаписать все файлы, генерируемые *initramfs*. Лично я бы этого не делал. Если Вы хотите установить программу, способную изменить вашу систему, воля Ваша.

Идея свободного ПО в том, что Вы всё можете изменить по своему личному желанию, даже если ради этого придется что-то сломать. Но я могу понять и точку зрения разработчиков — если Вы устанавливаете программу и ломаете систему, они должны собирать все по кусочкам, отвлекаясь от другой, более продуктивной работы.

Вы можете последовать совету в сообщении и удалить *initramfs-tools*, но это довольно крутая мера ради того, чтобы просто попробовать *Dracut*. Решение простое: установить *Dracut* из исходников, тем самым миновав проверки, которые выполняются менеджером пакетов. Это означает, что будут установлены и *Dracut*, и *initramfs*, но менеджер пакетов не будет знать о *Dracut*.

Помните, что *Dracut* может попробовать создать файлы *initramfs* с теми же именами, что и исходные, хотя не перезапишет их, если не указать параметр **--force**. Если это произойдет, нужно либо велеть *Dracut* использовать другое имя (указав его в командной строке), либо переименовать исходные файлы. В любом случае нужно создать дополнительные пункты меню загрузки *Grub*, чтобы при загрузке у Вас был выбор между исходным *initramfs* и *initramfs* от *Dracut* на случай, если Ваш новый *initramfs* будет демонстрировать неоптимальные характеристики (или не будет »

Коротко про...

top

Ваш компьютер когда-нибудь начинал работать медленно, зависал или сбрасывал всю информацию с диска по непонятной причине? Если это произошло снова, запустите в терминале команду **top**.

Она отображает массу информации на небольшом пространстве. В верхней части окна по умолчанию отображается системная статистика — время работы системы, число пользователей, количество запущенных задач и т. д. Затем отображается информация об общем использовании процессора, разделенная на программы пользователя (us), системные операции (sy), процессы с измененным приоритетом (ni), простаивающие (id) и ожидающие (wa) процессы.

Очевидно, что простаивающие процессы — это скорее неиспользование ресурсов, чем их использование. Здесь отображается объем доступных ресурсов процессора. Напротив, ожидающие процессы — это процессы, захватившие ресурсы, обычно ввод/вывод на жестком диске.

Ниже располагается список выполняющихся задач, отсортированных по потреблению процессора. В нем в любой момент можно посмотреть, какая программа съедает все ресурсы процессора. Для изменения сортировки, например, по потреблению процессора или памяти, нажмите F.

Процессы не только отображаются, ими можно и управлять. Нажмите k, чтобы убить процесс, или r, чтобы изменить его приоритет — увеличить значение nice, сделав процесс менее жадным. Чем больше значение nice, тем чаще процесс говорит другим «только после вас». Для выполнения обеих операций необходим идентификатор процесса (PID), указанный в левой колонке. Опечатка здесь может привести к печальным последствиям, но если запустить **top** в X-терминале, можно дважды щелкнуть на PID необходимого процесса и нажать среднюю кнопку мыши, чтобы скопировать и вставить его и тем самым избежать неприятностей.

› Если ваш менеджер пакетов не хочет устанавливать программу, вы можете сделать это вручную, установив ее из исходников.

работать). Учтите, что эта проблема может также оказывать влияние и на другие дистрибутивы, например, на вездесущий Ubuntu.

4 Пересылка почты

В У меня Fedora 21, и я не могу заставить работать почту. Я установил SSMTP, но функция mail() PHP по-прежнему не работает. Полагаю, я должен изменить конфигурационный файл SSMTP, но не знаю, где. Где и что я должен изменить в этом файле?

barrythai, С форумов

О SSMTP — утилита перенаправления почты, похожая на *sendmail* для вызывающих ее программ, но все, что она делает — лишь перенаправляет почту через другой сервер. Это означает, что программы смогут отправлять почту без полноценного почтового сервера, установленного на Вашем компьютере. Файл настройки SSMTP — **/etc/ssmtp/ssmtp.conf**, но чтобы утилита заработала, его надо изменить. Самый важный параметр, который нужно задать — **mailhub**. Этот параметр задает адрес почтового сервера, которым Вы пользуетесь — это может быть локальный сервер или сервер в сети, но чаще всего это почтовый сервер Вашего провайдера. Если сервер использует порт, отличный от стандартного 25, его можно указать в **mailhub** следующим образом:

```
$ mailhub=mail.ваш.домен:2525
```

Возможн, Вам также пригодится параметр **Hostname**, который определяет имя хоста, используемого SSMTP при отправке почты. Некоторые почтовые серверы уровнем выше могут пожаловаться, если он не соответствует серверу в почтовом адресе. Если параметр не установлен, будет использовано имя хоста компьютера по умолчанию (которое выводится командой **hostname -f**).

Если у Вас или провайдера используется аутентификация SMTP, также понадобится указать логин и пароль в параметрах **AuthUser** и **AuthPass** для сервера, заданного в **mailhub**. Подробности можно найти в комментариях файла **ssmtp.conf** по умолчанию или на man-странице **ssmtp.conf**.

5 Копирование и распаковка

В Я переношу все файлы на новый жесткий диск и столкнулся с небольшой проблемой. У меня есть много файлов, имена которых начинаются с "United Kingdom", после чего идет цифра, слово в скобках и расширение файла, вот так: United Kingdom 1 (Old English).txt.

Проблема в том, что cp не любит скобок, и мне приходится пользоваться терминалом и sudo, так как выделение файлов и копирование правой кнопкой мыши не работает, потому что я копирую их в системный каталог. Файлы, о которых я говорю, находились в ZIP-архиве, и я распаковал их командой:

```
$ unzip file.zip -d /
каталог_для_распаковки_файлов
```

Но вместо распаковки файлов в каталог они распаковались в подкаталог, который был создан в этом каталоге. Не могли бы вы подсказать, что я делаю не так?

chris, С форумов

О Проблема со скобками в том, что они являются специальными символами оболочки, используемыми для проверки шаблонов. Поэтому оболочка пытается интерпретировать их перед передачей имен файлов команде **cp** — это делает оболочка, а не **cp**: Вы получите ту же ошибку, если замените **cp** на **ls** (что всегда стоит сделать с шаблонами перед реальным копированием). Чтобы осчастливить оболочку, Вам нужно экранировать специальные символы, как скобки, так и пробелы, вот так:

```
United\ Kingdom\ 1\ \(Old\ English\).txt
United\ Kingdom\ 2\ \(Magna\ Carta\).txt
```

Чтобы заставить оболочку добавить экранирующие символы, можно использовать автодополнение по Tab — попробуйте Uni[TAB]1[TAB]. Но это все равно довольно неудобно, не говоря уже о том, что может привести к ошибкам, если файлов много; поэтому проще скопировать их с помощью шаблона:

```
$ sudo cp -p Unit[TAB]*.txt /destination/path
```

ZIP-файл — это архив файлов в структуре каталога, при распаковке которого структура каталога сохраняется. Если указан параметр **-d**, то эта структура каталога будет просто распакована в указанный путь, поэтому если в архиве все было в каталоге, то Вы по-прежнему получите этот каталог. Если все файлы находятся в одном каталоге без подкаталогов, можете воспользоваться параметром **-j** команды **unzip**. С ним все файлы архива будут распакованы в каталог назначения независимо от их расположения в иерархии архива. Если Вам нужно сохранить структуру каталогов, но родительский каталог не нужен, можете просто распаковать архив и затем перенести все на уровень выше.

```
$ sudo unzip -d /dest/path/ yourarchive.zip
$ sudo mv /dest/path/yourarchivedir/* /dest/path
$ sudo rmdir /dest/path/yourarchivedir
```

Командой **rmdir** нельзя удалить непустой каталог, поэтому последний шаг также проверяет, что перемещены все файлы — шаблон * по умолчанию не включает файлы с точками.

6 Пробуем Linux впервые

В Я хочу установить на свой компьютер Linux и хочу поместить его на отдельный диск. Для загрузки я хочу нажимать F8 и затем загружать Linux с этого диска. Мне хотелось бы, чтобы Linux распознавал остальные диски под буквами C, D, E и т.д., чтобы я мог воспользоваться документами на других дисках.

У меня Windows 7 Professional на восьмиядерном AMD FX с 16 ГБ оперативной памяти. У меня шесть жестких дисков внутри компьютера, один — съемный, и еще один можно разместить в верхней части корпуса. Чтобы установить Linux и сохранить его отдельно от Windows, нужно ли отключать другие диски во время установки?

Деннис Робб [Dennis Robb]

О Похоже, Вы точно знаете, чего хотите. Linux можно установить на любой диск в компьютере, и он может не иметь четкого номера в списке операционных систем, при этом установка на отдельный диск сделает жизнь проще. Вам подойдет почти любой популярный дистрибутив Linux, но я бы выбрал один из самых популярных, пусть только и потому, что при возникновении проблем Вам сможет помочь большее по количеству сообщество пользователей.

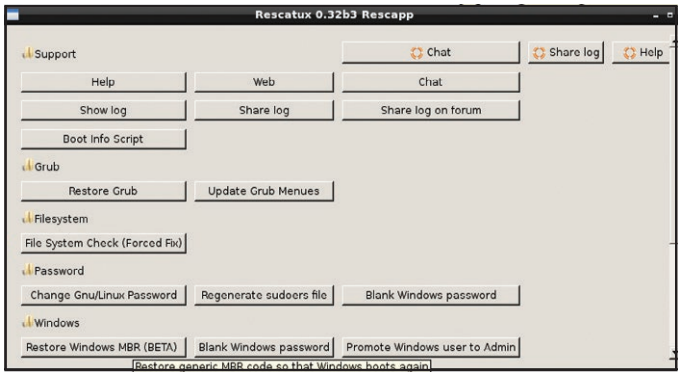
Установщик дистрибутива должен определить Ваши разделы с Windows и настроить их монтирование при загрузке системы Linux, так что они будут доступны в каталогах вроде **/mnt/windows_C**, **/mnt/windows_D** и т.д. Точное расположение может отличаться, но эти каталоги будет легко найти в файловом менеджере или в меню Places на рабочем столе.

Так как у Вас Windows 7, то диски, вероятно, отформатированы в NTFS. В Linux хорошо поддерживается FAT, но поддержка NTFS ядром довольно ограничена — часто тем, что вы можете только читать данные с дисков, но не записывать их. Решение — установить пакет *ntfs-3g* с помощью менеджера пакетов дистрибутива (иногда он называется «программным центром» в меню). Этот пакет решит проблему с NTFS.

Отключать свои диски с Windows не требуется: если это сделать, то Linux не сможет определить файловые системы Windows. Любой установщик дистрибутива Linux увидит Ваши диски и спросит, на который из них Вы предпочитаете его установить.

А потом есть два варианта.

Первый — установить загрузчик на дополнительный диск и потом выбирать его из меню



› Если вы захотите удалить диск или раздел с Linux из двойной загрузки, то в Rescatux есть кнопка, которая вам пригодится — она восстановит стандартный загрузчик Windows.

загрузки, как Вы и хотели. Это можно сделать в большинстве дистрибутивов, но соответствующая опция нередко скрывается под кнопкой Advanced options [Дополнительно].

Второй вариант — установить загрузчик на главный диск Вашей системы.

Обратите внимание, что это не программа Linux, а только код загрузчика, который может загрузить любую операционную систему. Затем во время загрузки Вы получите меню, которое позволит выбирать операционную систему. По умолчанию будет загружаться Linux, но это легко изменить, если Вы хотите по умолчанию загружать Windows или подождать, пока пользователь сам выберет операционную систему — обычно пользователю дается 10 секунд на выбор, и затем, если он не нажал любую кнопку для остановки таймера и не выбрал другого варианта, загружается ОС по умолчанию.

Единственный недостаток последнего подхода в том, что если отключить диск с Linux, Вы потеряете меню загрузки.

Однако это легко исправить, либо с помощью загрузочного диска Windows в аварийном режиме (найдите вариант fixmbr), либо с помощью диска восстановления системы, такого как Rescatux, который восстановит исходный загрузчик Windows по одному щелчку мыши. 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/HardwareLiSter). Одна из указанных программ непременно должна быть включена в ваш дистрибутив (а иногда и обе).

Если вы не хотите или не можете их установить, выполните 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

WPA

› Что такое WPA?
WPA — более надежный потомок WEP.

› Хорошо, спасибо за лаконичный ответ, но все же ЧТО это?
WPA (Wi-Fi Protected Access — защищенный доступ по Wi-Fi) — система, гарантирующая приватность данных, передаваемых по беспроводным сетям.

› А оно мне надо?
Если вы живете не в пещере и не в экранированном бункере, то почти наверняка да. В отличие от традиционной проводной сети, в которой ваши данные никогда не покидают проводов, соединяющих компьютеры, беспроводная сеть распространяет их на сотни метров. Это значит, что любой, у кого есть ноутбук и сетевая карта для беспроводной сети, может так же легко прочесть ваши данные, как при подключении к вашему коммутатору напрямую.

› У меня большой дом, ноутбук даже не подключен к сети в остальных комнатах, поэтому я в безопасности, так ведь?
Нет! Хотя начиная с определенного расстояния ваш ноутбук может не подключиться к точке доступа, могут обнаружиться другие устройства, которые смогут. Если прикрепить к беспроводной сетевой карте антенну, вы получите гораздо больший диапазон.

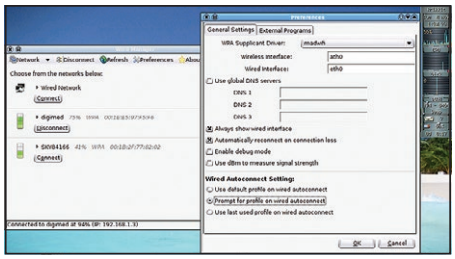
В этом может не быть ничего страшного, если вы пользуетесь Wi-Fi только дома для выхода в Интернет с ноутбука. Но что если ваша компания пользуется сетью Wi-Fi со слабой защитой (или вообще без нее), а кто-то в припаркованном снаружи автомобиле нахально считывает все конфиденциальные данные?

› Раз уж вы упомянули это: что такое WEP?
WEP — Wired Equivalent Privacy [приватность, эквивалентная проводной сети]. Это была предыдущая попытка защиты беспроводных сетей. Хотя это лучше, чем вовсе не иметь защиты, взломать ее не так трудно. Она может спасти вас от соседей, которые случайно подключатся к вашей точке доступа вместо своей собственной, но не спасет от того, у кого есть намерение взломать ваше соединение.

› Какие устройства необходимы для WPA?
Все новые беспроводные сетевые карты и точки доступа поддерживают WPA и WEP. Если у вас более старое устройство, может оказаться, что оно поддерживает только WEP.

› Как насчет ПО?
Вам нужна поддержка WPA на двух уровнях — на уровне драйверов устройства и в отдельной программе, реализующей WPA. Самая популярная из таких программ — *wpa_supplicant*, она должна быть в вашем дистрибутиве. Если ее там нет, загрузить ее можно с сайта http://hostap.epitest.fi/wpa_supplicant. Это бесплатная программа, которая поддерживает обмен ключами и аутентификацию.

› Небось, чтобы воспользоваться WPA в Linux, придется читать man-страницы?
Можете сделать и так, но к счастью, есть и графические утилиты, способные упростить вам жизнь. Самая продвинутая из них, пожалуй, *Network Manager* в Gnome (http://www.gnome.org/projects/NetworkManager). Хотя это проект для Gnome, у него есть клиенты и для KDE, и для других рабочих столов.



› *Network Manager* — не единственный возможный вариант: *wicd* также поддерживает беспроводные (и проводные) подключения.

LXFHotPicks

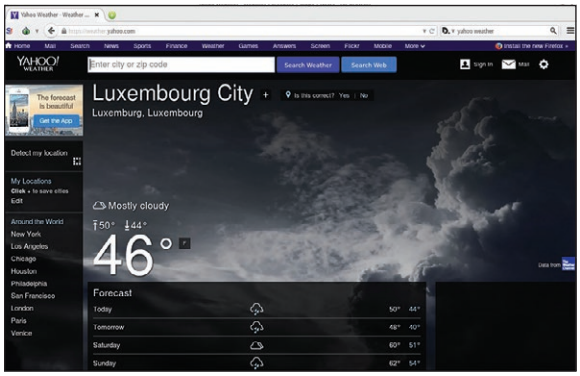
Александр Толстой
подает на стол еще один горшок горячего супчика с открытым кодом, приправленного пряным соусом свободы; на месяц вам хватит.

Tor Browser » GNU Nano » ZFSonLinux » Plank » GNU Hurd » QmapShack » Guake » Minilens » MegaGlest » Vera » DNSCrypt

Web-браузер

Tor Browser

Версия: 4.5 **Сайт:** www.torproject.org



› Этот метеосайт не смог засечь наше местонахождение. Отлично! Правда, погоды нам теперь не узнать...

Браузер *Tor* хорошо знаком пользователям Linux, озабоченным сохранением своей конфиденциальности. Он неоднократно упоминался в нашем Сравнении дистрибутивов-анонимайзеров [см. **LXF196**, стр. 30] в качестве основного компонента разных решений обеспечения анонимности. *Tor* — не просто браузер: он также включает группу серверов, чья работа обеспечивается добровольцами, и эти сервера позволяют людям обеспечить конфиденциальность и безопасность в Интернете; причем пользователю не обязательно быть технарем, потому что все тихо работает внутри. Сюда входит подключение через ряд виртуальных каналов вместо установки прямого соединения, что означает возможность делиться информацией в общественных сетях без угрозы конфиденциальности. *Tor* также послужит эффективным средством обхода цензуры, поскольку способен предоставить пользователям доступ к заблокированным сайтам или контенту.

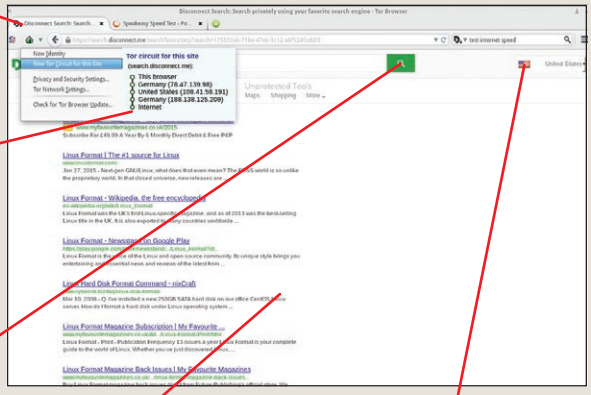
Tor основан на последнем стабильном релизе *Firefox* плюс несколько важных плагинов и расширений, которые устанавливаются и настраиваются по умолчанию. Среди них: расширение HTTPS Everywhere для шифрования трафика на всех сайтах; NoScript для отключения JavaScript по умолчанию; и Fteproxy для обхода проверки трафика. *Tor* даже умеет безопасно соединяться с сайтами в средах с ограничениями, где, например, блокируется весь трафик, кроме HTTP. Безопасность *Tor* обеспечивается с помощью транспортных уровней obafs2/3/4 и ScrambleSuit, которые маскируют ваш трафик и маршруты *Tor*.

> «Tor послужит эффективным средством обхода цензуры.»

Новая версия 4.5 предлагает ряд приятных улучшений, в том числе регулятор уровня безопасности, исправленное меню Onion и изоляцию узлов цепи. Кроме того, *Tor* блокирует свойства видео в API браузера и реализует методы, затрудняющие отслеживание и идентификацию пользователя.

Отныне *Tor Browser* использует поисковую машину Disconnect — основанную на Google One — и не собирает никаких данных пользователя. Браузер распространяется точно так же, как и обычный продукт Mozilla, и вам предлагается внушительный список сборок для любой комбинации ОС, языка и 32-/64-битных пакетов. Вы просто скачиваете и распаковываете соответствующую версию и запускаете *Tor* в директории **Applications**. Доходчивый мастер поможет вам выполнить пару настроек и установить соединение через цепь узлов *Tor*.

Исследуем интерфейс Tor Browser



Никаких скриптов
Кнопка NoScript предлагает солидный список опций для тонкого контроля за сайтами.

Меню Onion
Отсюда вы можете получить доступ к настройкам конфиденциальности и сетевым настройкам браузера, а также посмотреть на свою текущую цепь маскировки.

Настраиваемый поиск
Поисковая машина Disconnect анонимизирует ваш поиск благодаря специализированному VPN и частной версии поисковой машины Google, не позволяя сайтам и вашему ИП увидеть ваши параметры поиска. Кроме того, она обходит CAPTCHAS и баны.

Окно браузера
Tor отображает страницы не хуже браузера *Firefox*, несмотря на значительно более низкую скорость соединения.

Выберите страну
Результаты поиска будут разными для разных стран, и здесь вы можете выбрать свое географическое положение.

Текстовый редактор

GNU Nano

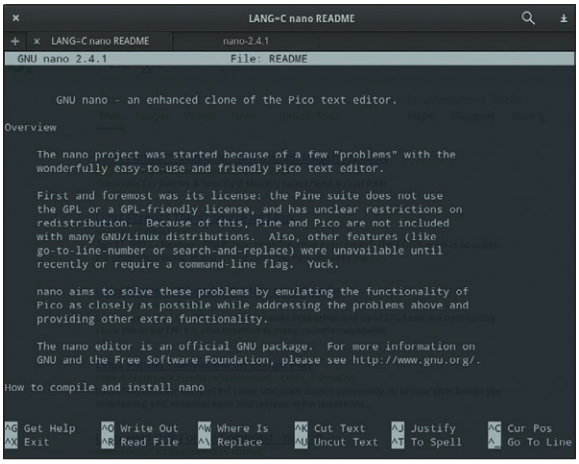
Версия: 2.4.1 **Сайт:** www.nano-editor.org

Среди новичков в Linux бытует шутка: у редактора *vi* есть два режима — в первом он попискивает, а во втором печатает бред. В этой шутке есть доля истины, поскольку *vi* держится старомодного подхода к UI (первый код был написан в 1976 г.!), и многие современные пользователи находят его двухрежимный принцип неуместным, особенно если надо всего лишь малость подправить текстовый файл. *GNU Nano* — более простая замена *vi*, и это изящный консольный редактор на *Ncurses*; он основан на редакторе *Pico*, но схож и с классическими.

На нижней панели размещен набор подсказок для быстрых действий, которые привязаны к сочетанию Ctrl и выбранной клавиши, но вы можете выучить и побольше удобных горячих клавиш. Используйте Alt+Space и Ctrl+Space, чтобы перейти на одно слово назад и вперед соответственно, Alt+L для переключения разбивки строк и Alt+G для перехода на строку с заданным номером. Имеются параметры для запуска Nano с индивидуальными настройками или функциями (**nano -h** огласит весь их список). Попробуйте **nano -t** для автосохранения изменений при выходе из программы, **nano -B** для тихого и незаметного автоматического сохранения всех редактируемых файлов, и **nano –m** для включения поддержки мыши (удобно для текстового режима с GPM).

Новая версия 2.4 — первый за долгие годы стабильный релиз, и он воплотил многие новые функции из серии 2.3, в т.ч. полноценную систему undo (отныне включенную по умолчанию), совместимую с *vim* блокировку файлов, поддержку linter, поддержку форматирования, гибкое выделение синтаксиса и множество отладок проблем, о которых сообщалось с 2.2.

> «Замена vi и изящный консольный редактор на Ncurses.»



› Освоение *Nano* не требует особых навыков, и во многих дистрибутивах он уже стал редактором по умолчанию.

Ваш дистрибутив Linux может по-прежнему предлагать старую версию 2.2.x, а на сайте *Nano* отсутствуют опции скачивания, кроме tar-архива исходника, но компилируется это приложение очень легко. В Ubuntu просто подготовьте почву с помощью

```
$ sudo apt-get install build-essential
```

и, предполагая, что в вашем *apt* подключены репозитории 'source', установите зависимости компиляции *Nano*:

```
$ sudo apt-get build-dep nano
```

и скомпилируйте *Nano* с помощью

```
$ ./configure && make && sudo make install
```

Файловая система

ZFSonLinux

Версия: 0.6.4 **Сайт:** http://zfsonlinux.org

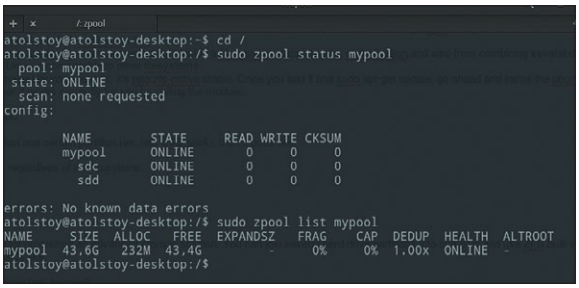
ZFS — это комбинация файловой системы и менеджера логических томов, изначально разработанная Sun Microsystems и реализованная в Solaris OS. Файловая система ZFS имеет множество расширенных и уникальных функций: например, она способна хранить до 256 зебибайтов (ЗиБ) данных, организованных в 264 пула данных максимум, тогда как один пул может хранить до 3×1,023 петабайт данных.

Поддержка таких больших объемов опирается на 128-битную архитектуру ZFS и широко используется в устройствах хранения данных уровня предприятия. ZFS не является технологией чисто для Solaris — встроенный порт ядра Linux существует уже не первый год и считается стабильным, начиная с 2013 г. ZFS удобна для домашнего пользования, поскольку в ней имеется функция моментальных снимков, клонов копирования при записи, постоянная проверка целостности и автоматическое восстановление. Функция моментального снимка означает, что резервное копирование файла производится без всяких сторонних программ, только благодаря встроенным функциям файловой системы.

Есть еще удобные PPA (ppa:zfsnative/stable) с готовыми к использованию пакетами для всех основных релизов Ubuntu. Добавив их и введя **sudo apt-get update**, продолжайте — установите пакет *ubuntu-zfs*, который найдет остальные зависимости и скомпилирует модуль ядра Linux для поддержки ZFS. Проверьте, правильно ли он работает, вызвав модуль:

```
$ sudo modprobe zfs
$ lsmod | grep zfs
```

> «ZFS удобна для дома: в ней есть функция моментального снимка.»



› Менеджер логического тома ZFS — интересная альтернатива томам LVM в Linux.

Давайте опробуем несколько примеров. Сначала создадим пул из двух USB-брелков на **/dev/sdb** и **/dev/sdc**:

```
$ zpool create mypool sdb sdc
```

Чтобы более точно определять устройства, используйте ID устройства вместо sdX. Выясните его через команду **sudo ls /dev/disk/by-id**. Она молча создаст пул и монтирует его как директорию **/mypool**. Чтобы проверить ее статус, введите **sudo zpool status mypool**, а чтобы увидеть его том, воспользуйтесь **sudo zpool list mypool**. Теперь создадим Snapshot1 текущего состояния файловой системы:

```
$ sudo zfs snapshot mypool@snapshot1
```

Вы можете создать сколько угодно моментальных снимков, создать их списки, разрушить или откатить свою систему к определенному моментальному снимку — все это описано в **zfs --help output**.

Док

Plank

Версия: 0.9.0 **Сайт:** https://launchpad.net/plank

Apple OS X впечатлила многих отчасти и благодаря своим чудесным и гладким докам, которые породили множество клонов в Linux. Самым потрясающим визуальным эффектом в Mac OS было увеличение значков: его было трудно реализовать, удержав низкое потребление ресурсов доком.

Одним из самых интересных и удобных доков в Linux является *Docky*, который на самом деле основан на более простом доке под названием *Plank*. Цель *Plank* — обеспечить лишь то, что необходимо в доке, и ничего больше, однако это библиотека, которую можно расширить для создания программ других доков с более продвинутыми функциями. Таким образом, *Plank* — это лежащая в основе *Docky* технология (начиная с версии 3.0.0), и ее цель — обеспечить все основные функции, тогда как *Docky* расширяет ее, добавляя всякие интересные штучки типа docklet, редакторов, диалоговых окон настройки и т. п.

Почти весь код написан на Vala, и настройки *Plank* доступны в контекстном меню правой кнопки обязательного значка «якоря» в большинстве дистрибутивов Linux (или на системной панели Settings, если вы используете elementary OS). При всей своей простоте, он предлагает солидный объем индивидуальной настройки, включая размер значков, опции скрытия, программу выбора тем (новые темы лежат в **~/.config/plank/theme**), положение экрана, размещение дисплея (если используется более одного дисплея). В *Plank* заметно не хватает опций анимации, но этот док по своему дизайну не предусматривает их предоставления, поскольку должен оставаться простым. Однако, вместе с новыми функциями и отладками, *Plank 0.9.0* все же добавил анимацию для добавления и удаления объектов, новый хитроумный режим скрытия (док скрывается, если он заслоняет окно), новую опцию настройки отложенного скрытия, отладки для недавних релизов *GTK3* (после 3.14) и многое другое. *Plank* по умолчанию поставляется с elementary OS, но доступен также в отдельном PPA (ppa:docky-core/stable) для Ubuntu и его производных, а также в стандартных репозиториях многих других дистрибутивов Linux (Arch, Fedora и CentOS — лишь некоторые из них).

> «Цель Plank — обеспечить лишь необходимое в доке, и не более.»

› **Док *Plank* быстр, прост и не привязан ни к какому рабочему столу — то есть универсален.**

Ядро операционной системы

GNU Hurd

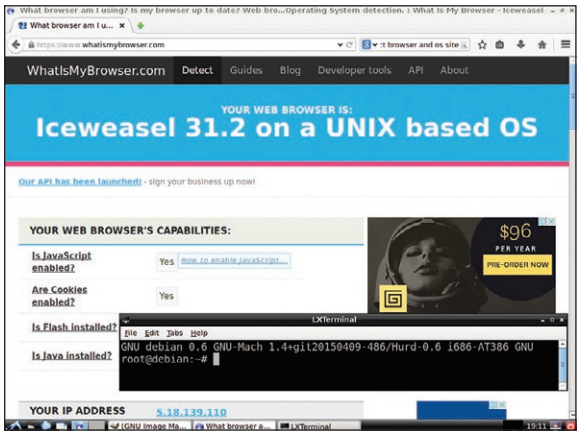
Версия: 0.6 **Сайт:** www.gnu.org/software/hurd

GNU/Hurd — долгосрочный проект сообщества, целью которого является воссоздание ядра Unix. Hurd — это Hird of Unix-Replacing Daemons, a Hird — это Hurd of Interfaces Representing Depth; и это первая программа, которая названа парой взаимно рекурсивных акронимов. GNU Hurd реализует альтернативу ядра Linux посредством сбора ряда серверов, работающих поверх микроядра GNU Mach. Эти серверы реализуют файловые системы, сетевые протоколы, управление доступом к файлам и прочие функции, обычно присущие ядру ОС.

Пусть это звучит слегка патетически, но Hurd — вечная мечта об истинно свободном ядре, объединяющая философию Unix с духом свободных технологий. GNU/Hurd 0.6 — результат более 18 месяцев разработки, предлагающий обновление GNU Mach 1.5, основную поддержку ACPI, механизм инициализации в стиле SysV и гораздо лучший стек fakeroot для создания еще большего количества сторонних пакетов, плюс множество других обновлений.

Поскольку GNU Hurd — это все-таки нечто вроде ядра, вам нужен дистрибутив, чтобы с ним поработать. И у нас такой есть: команда Debian обновила свой релиз 8.0 респином Debian/Hurd, который использует респин Debian/FreeBSD для создания неофициальной версии Debian. Помните, что GNU Hurd обеспечивает лишь весьма скромный набор «настоящих» драйверов оборудования, а значит, работать-то он, вероятно, будет, но без аудио, USB и поддержки OpenGL. Вот почему разумнее будет запустить виртуальную машину и протестировать Debian-Hurd на ней.

> «GNU/Hurd — это вечная мечта об истинно свободном ядре.»



› **Система на базе Hurd определяется в Интернете как «операционная система на базе Unix».**

Очень скоро вам станет ясно, что этот дистрибутив не готов для производственного использования, однако Debian GNU/Hurd 8.0 хорошо подходит для его тестирования с минимальными усилиями. Во время установки надо будет выбрать английскую (US) раскладку клавиатуры, чтобы заработал сервер *Xorg* (это известное ограничение), и после первой загрузки введите **startx**, чтобы загрузить рабочий стол *LXDE*. Debian GNU/Hurd предлагает около 80% официальных пакетов Debian, так что вы сможете играть с системой и исследовать ее, как вам заблагорассудится.

Инструмент карт GPS

QMapShack

Версия: 1.2.2 **Сайт:** http://bit.ly/QMapShack

Для нишевого ПО всегда найдется место, и несмотря на отсутствие ажиотажа вокруг него, мы рады сообщить, что *QMapShack* — первоклассное приложение с открытым кодом. Оно является потомком *QLandkarteGT*, мощного приложения GIS, способного отображать ваши данные GPS на разных картах, планировать маршрут, исследовать ландшафт, отслеживать перемещение объектов и делать еще много чего хорошего.

Возможно, внешне *QMapShack* похож на научное приложение, но на самом деле нацелен на туристов, любителей отдыха в палатках и вообще на свежем воздухе с устройством GPS при себе, будь то отдельный Garmin, TwoNav или просто смартфон с GPS.

QMapShack умеет импортировать, просматривать и редактировать данные GPS в разных форматах файлов, таких, как GPX, и поддерживает карты в формате GEOTIFF, а также векторный формат карт Garmin IMG.

При первом запуске *QmapShack* его интерфейс будет практически пуст, однако это мгновенно починяется кнопкой 'instant fix' для быстрого доступа к набору стандартных карт. Стоит на нее нажать, как загрузятся 6 карт в форматах TMS и WMTS.

Карты отличаются высоким качеством (OSM Topo, OpenStreetMap, Worldsat и т.д.) и показывают множество дополнительной информации, которую не найти на Google или Bing Maps. Простейший способ наложить ваш маршрут на выбранную карту — использовать меню File > Load GIS data и выбрать файл GPX (телефоны на Android могут генерировать GPX в приложении My Tracks, см. http://bit.ly/1GTBQi9).

Заставить *QMapShack* работать в Linux непросто, так как прекомпилированных бинарников немного — по сути, их обеспечивают только Arch AUR и Debian Sid. Для Ubuntu и его производных придется компилировать *QMapShack* из исходника, так что позаботьтесь о нужной среде компиляции, с помощью

```
$ sudo apt-get install build-essential g++ mercurial cmake cmake-curses-gui gdal-bin libgdal-dev libgdal1h pythongdal libproj-dev qttools5-dev-tools qttools5-dev qttools5-private-dev qt5-default libqt5webkit5-dev qtscript5-dev checkinstall
```

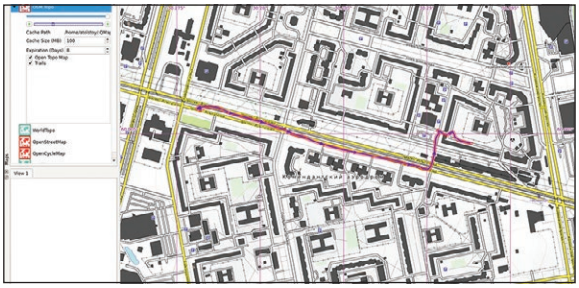
Затем найдите исходник приложения:

```
$ hg clone https://bitbucket.org/maproom/qmapshack QMapShack
```

и, наконец, скомпилируйте и установите его:

```
$ cd QMapShack && mkdir build && cd build && cmake ../ && make && sudo make install
```

> «Отлично сделанное приложение с открытым кодом.»



› Самый распространенный способ применения *QMapShack* — поиск маршрута от дома до ближайшей пивной.

Терминал

Guake

Версия: 0.7 **Сайт:** https://github.com/Guake/guake

Количество эмуляторов терминала в Linux уступает разве что количеству текстовых редакторов. Хотя большинство этих приложений — похожие друг на друга консоли, есть и более элегантные решения. В KDE можно нажать F4 в *Dolphin* и получить мгновенный доступ к командной строке. Еще один подход — перетащить терминал вниз с верха экрана, как вы видите во многих компьютерных играх; и *Guake* — отличный пример подобного вдохновленного играми UI, получившего свое название в честь легендарной игры *Quake* и написанного на Python (по большей части). *Guake* будет отлично сочетаться с выбранным вами рабочим столом (каким бы он ни был), и отлично поддается настройке.

Guake предлагает множество функций и опций, доступных через меню Preferences. Меню доступно и отдельно — чтобы им воспользоваться, приложение открывать не обязательно. Опции разделены по вкладкам, где можно выбрать любой интерпретатор, визуальные параметры (например, прозрачность) и даже привязку терминала (вкладка General).

Новая версия 0.7 улучшила поддержку нескольких мониторов; добавила больше цветных палитр; возможность запускать определенный пользователем скрипт при запуске *Guake*; опцию открытия ссылок (контекстное меню ссылки в окне терминала) и исправленную функцию автоскрытия, которая теперь корректно работает с современными средами рабочего стола (типа Unity).

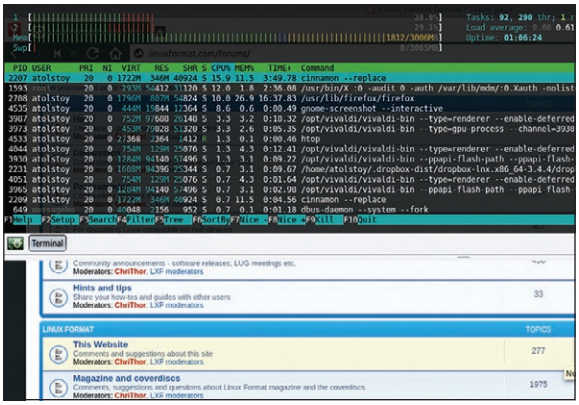
Если раньше вы не использовали *Guake*, то первый запуск данной версии может вызвать сообщение об ошибке:

```
# Guake can not init! Gconf Error. Have you installed guake.schemas properly?
[Guake не инициализируется! Вы правильно установили guake.schemas?]
```

Чтобы исправить ошибку, просто запустите *Guake* снова, и всё будет отлично. Вызывайте *Guake* простым нажатием F12, а скрывайте повторным нажатием на ту же клавишу.

Пакеты *Guake* доступны в нескольких дистрибутивах Linux, включая Ubuntu и Arch, хотя 15.04 и даже AUR предлагают устаревшую версию 0.5 (а предыдущие версии Ubuntu вообще предлагают 0.4.x). Чтобы найти и установить *Guake 0.7*, воспользуйтесь ppa:webupd8team/unstable.

> «Отличный пример вдохновленного играми расширения UI.»



› Немедленный доступ к командной строке, без лишних значков или пунктов меню: всё, что вам нужно — F12.

HotGames Развлекательные приложения

Головоломка

Minilens

Версия: 1b2 **Сайт:** http://alketii.github.io/minilens

Minilens — имя робота, которым вы управляете в игре, где вам приходится очищать пост-апокалиптическую Землю от радиоактивных бочек и собирать цветочки. Как можно догадаться, создатели игры хотят донести до нас необходимость заботиться о нашей планете, пока не станет слишком поздно.

Но хотя этот посыл весьма серьезен, игра очень жизнерадостная и яркая, что будет очень привлекательно для детей. Вы можете перемещать своего маленького робота по экрану, направо и налево, но не можете прыгать. Здесь множество уровней, которые идут в пакетах с разнообразными картами и дополнительными элементами, например, бомбами и лестницами.

Игровой процесс *Minilens* напоминает *KGoldRunner*, головоломку *Om Nom* и кучу классических аркад. Это интересная головоломка с уровнями, которые потребуют некоторого времени и усилий на прохождение; к счастью, у вас неограниченное число жизней.

По умолчанию вы получаете восьмиуровневый пакет обучения и еще 18 уровней в еще шести пакетах. Установка *Minilens* очень проста, хотя на сайте игры нет пакетов для отдельных дистрибутивов. Взамен предлагается ZIP-файл со статической компиляцией. Есть и 32-, и 64-битные версии; и этого должно быть достаточно для любого пользователя. Игра отличается минимализмом во всем, не только в своей мультяшной графике, но и в доставке файлов: внутри ZIP-файла игры вы найдете только один консолидированный файл игры и еще один файл для уровней.

> «Уровни потребуют времени и усилий на прохождение.»

› *Minilens* напоминает *KGoldRunner*, головоломку *Om Nom* и кучу классических аркад.

Игра использует Godot Engine (www.godotengine.org), и поэтому принимает от пользователей самодельные уровни в обыкновенном формате XML — загляните на сайт *Minilens* в поисках инструкций по созданию собственных головоломок.

Стратегия

MegaGlest

Версия: 3.11.1 **Сайт:** https://megaglest.org

MegaGlest — фантастическая свободная стратегия, которая появилась из студенческого проекта под названием *Glest*, который был прекращен в 2009 г., но позднее ответвился. Фантастический мир игры населен семью расами: Магами, Технарями, Индейцами, Египтянами, Норманнами, Персами и Римлянами, и вместе они образуют 'Megapack'. У каждой расы свой набор юнитов, зданий и обновлений, а также преимуществ и недостатков, что позволяет разнообразить стратегию, притом поддерживая сбалансированный сюжет: например, раса Технарей использует традиционных человеческих воинов, имеет в своем арсенале средневековые механические устройства и сильна в рукопашной, а раса Магов создана для более продвинутых игроков, и большинство их юнитов превращаются или собираются другими. Карты игры варьируются по размеру и ландшафту: есть, например, большие равнины и поля с реками, горами, морями или утесами. Игроки должны создавать поселения, чтобы накапливать ресурсы, защищаться от других игроков и обучать юниты исследовать карту и атаковать врагов.

Легко догадаться, что *MegaGlest* — это игра в стиле *Age of Empires*, со множеством дополнительных штрихов у листьев, элементов, деталей ландшафта... и, конечно, с превосходной воинственной музыкой. Мы не можем также не сравнить *MegaGlest* с *0 A.D.*, еще одной стратегией в реальном времени с открытым кодом [см. HotPicks **LXF193**]. Хотя *MegaGlest* явно отстает по качеству графики, он может похвастаться более зрелым, и, следовательно, лучшим игровым процессом. Игра за определенную расу в *MegaGlest* может также занять больше времени и потребовать больших навыков, независимо от того, играете ли вы с ИИ или с игроками LAN. В *MegaGlest* также есть удобный редактор карт и программа просмотра моделей, поэтому вы можете сами создавать собственные режимы игры, с индивидуальными юнитами и способностями.

> «У каждой расы свой набор юнитов и строений.»

› Добыча золота — ключ к обладанию множеством работников и воинов, а также и к прокачке их умений.

Игра широко доступна во многих дистрибутивах Linux, плюс можно скачать статическую сборку прямо с сайта проекта. Например, в Ubuntu игра находится в репозитории Playdeb (www.playdeb.net).

Среда рабочего стола

Vera

Версия: 1.0.0 **Сайт:** https://github.com/vera-desktop

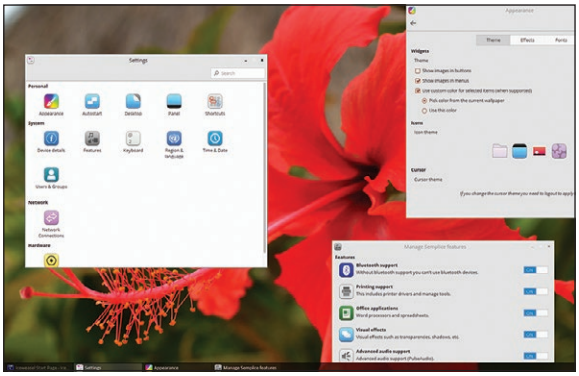
Среды рабочего стола — это программы, где богатство выбора проявляется особенно ярко. Помимо полнофункциональных рабочих столов, существует ряд простых легковесных и мало нагружающих ресурсы разновидностей. [см. более подробно о *Budgie* в Hotpicks **LXF194**].

Vera — еще один представитель данного подвида, с собственными свойствами. Для начала, это новая ориентированная на плагины среда рабочего стола на базе *GTK3* на ранней стадии развития (вследствие чего немного шероховата). Она состоит из ряда основных компонентов, написанных на Python 3 и Vala, таких, как диспетчер электропитания и плагин соответствующей панели, центр управления, оболочка рабочего стола, программа автозапуска приложений и некоторые части интеграции.

Интеграция необходима из-за разных сторонних элементов, от которых зависит *Vera*. Перечислим лишь некоторые: менеджер файлов *PCManFM* (скомпилированный на *GTK3*), менеджер окон *OpenBox*, апплет Blueman, *Screenshot* из *XFCE 4*... и т.д. Vera Desktop объединяет их все, добавляет собственную панель (Tint2), генератор меню (Alan2) и предлагает отличное решение с изящной темой на основе Zukitwo и крутыми значками Faba.

Одной из уникальных функций является дружелюбный мастер, который появляется при первом запуске рабочего стола. Мастер основан исключительно на скриптах и выглядит очень глубоко интегрированным в рабочий стол; он проведет вас по основным сценариям использования и основным функциям, и это — руководство, лучшее в своем классе, конечно, с точки зрения новичка.

> «Среда состоит из базовых компонентов на Python 3 и Vala.»



› **Control Center сейчас является самым богатым на функции компонентом *Vera*.**

Основная оговорка, которую стоит сделать по поводу рабочего стола *Vera* — насчет доступности. Хотя можно создать рабочий стол вручную, скомпилировав код *Vera* и настроив под него свой дистрибутив Linu, все же самый лучший опыт работы по умолчанию предлагает Semplice Linux. Это быстрый и легкий современный дистрибутив Linux, который использует Vera Desktop по умолчанию.

Технически Semplice является основанным на Debian Unstable дистрибутивом со множеством небольших модификаций и несколькими разными приложениями, такими, как *Pragha* для воспроизведения музыки и *Iceweasel* в качестве web-браузера. Дайте ему шанс, особенно на старом оборудовании.

DNS-прокси

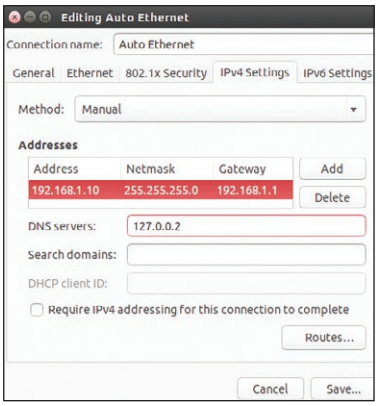
DNSCrypt

Версия: 1.4.3 **Сайт:** http://bit.ly/DNSCryptProxy

Идеальной сквозной безопасности между вами и сайтом можно добиться не всегда, но и маскирование — неплохое решение. Имя ему *DNSCrypt*, и это — локальный сервис для обеспечения безопасности соединений между клиентом и интерпретатором DNS. Причины, по которым вы, возможно, решите им воспользоваться — предотвращение шпионажа, атак через посредника или спуфинга, а чтобы использовать *DNSCrypt*, понадобится инструмент под названием *dnscrypt-proxy*, который, по утверждению авторов, «может применяться напрямую в качестве вашего локального интерпретатора или в качестве механизма продвижения DNS, аутентифицирующего запросы с помощью протокола DNSCrypt и передающего их предыдущему серверу». *Dnscrypt-proxy* поставляется в пакетах для Arch, Mageia, Debian Sid и Slackware.

Для полноты картины, вот несколько подсказок, как заставить его работать в Ubuntu и его многочисленных производных. Скачайте пакет *dnscrypt-proxy* из ppa:anton+/dnscrypt (надо признать, это самая простая часть). По умолчанию пакет настроен на использование 127.0.0.2 в качестве локального IP-адреса, так что он не вмешивается в имеющиеся настройки по умолчанию (127.0.0.1 в Ubuntu занимает *dnsmasq*). Также для дополнительной безопасности пакеты используют отдельного системного пользователя без привилегий — *DNSCrypt* войдет с помощью chroot в директорию **home** пользователя и немедля сбросит привилегии root для UID данного пользователя. Интерпретатором по умолчанию, работающим на *DNSCrypt*, используемым в пакете Ubuntu, является

> «Предотвращение шпионажа, атак через посредника, спуфинга.»



› **Для большей безопасности можно использовать локальный прокси своего интерпретатора DNS.**

DNSCrypt.eu Resolver #1 @ The Hague, Holland. Но это, как и другие настройки, можно изменить, отредактировав файл настройки **/etc/default/dnscryptproxy** (после чего запустите **sudo service dnscrypt-proxy restart**). Затем перейдите в настройки своего интернет-соединения в *NetworkManager*, укажите во вкладке IPv4 Settings DNS-адрес 127.0.0.2 и перезапустите сеть (**sudo service networking restart**).

Проверка работы *dnscrypt-proxy*—

```
$ sudo tcpdump -i eth0 dst host 176.56.237.171
```

Замените eth0 на имя соединения. IP-адресом будет интерпретатор DNS по умолчанию, используемый *dnscrypt-proxy*. LXF

На диске

Дистрибутивы, приложения, игры, подкасты и всякое-разное...

Лучшее из Интернета, упакованное в 9 ГБ качественного DVD.

Дистрибутивы

В этом месяце у нас самый свежий релиз Debian, а в прошлом месяце был Ubuntu 15.04. Ubuntu произошел от Debian, но эти два дистрибутива совершенно разные. Ubuntu выпускает релизы каждые полгода; релизы Debian выходят «по мере готовности» и только после тщательного тестирования на всех поддерживаемых платформах. Ubuntu охватывает только платформы x86, 32- и 64-битные, для настольных ПК, ноутбуков и серверов; Debian поддерживает еще ряд архитектур, таких, как ARM и PowerPC. Debian управляется демократическим советом с общественным договором, а Ubuntu — коммерческой организацией.

Различия этим не исчерпываются, но вы видите, что у них очень разный подход. И при этом в основе лежит один и тот же код. Это иллюстрирует как минимум гибкость экосистемы GNU/Linux.

Одно из различий в том, что релизы Debian используют ветвь Stable, а Ubuntu рассчитывает на ветвь Debian Testing. Использование здесь слова 'Stable' вызывает некую путаницу. Оно не означает, что другие ветви нестабильны или имеют склонность к сбоям: здесь имеется в виду отсутствие резких изменений, что весьма важно для производственных систем.

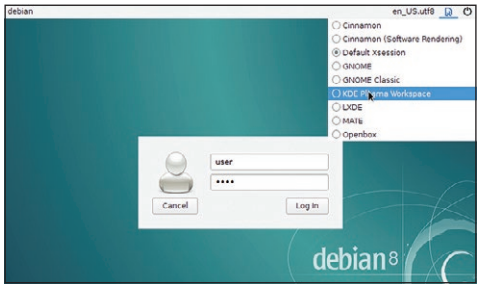
Neil

Универсальный дистрибутив

64-битный

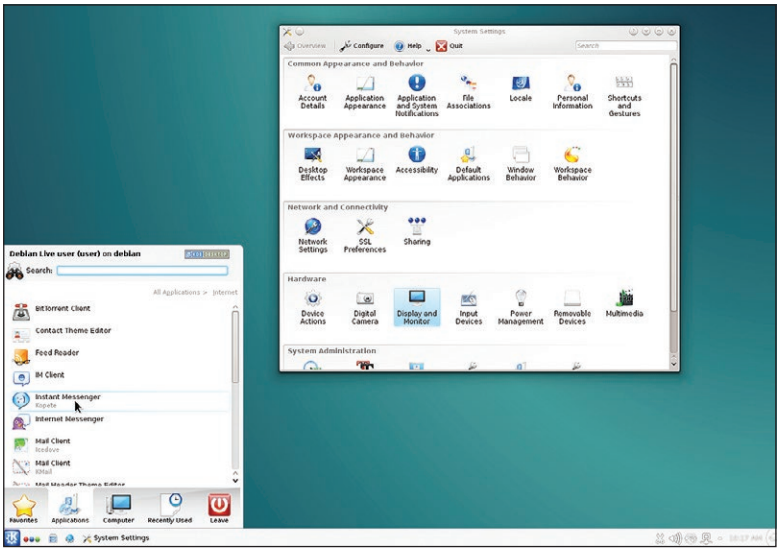
Debian 8

Основные релизы Debian случаются не слишком часто, так что можете быть уверены, что мы отметим это событие, включив его в **LXFDVD**. Debian считается одним из самых стабильных и солидных дистрибутивов. Возможно, в нем нет самых новых блестящих программ (правда, в этой версии Debian переходит на *Systemd*), но зато он совершенно надежен. Debian 8, под кодовым названием Jessie, дистрибутив массивный: в нем более 43 000 программных пакетов. Весь набор установочных дисков целиком заполняет три DVD. Диапазон предлагаемых ими программ весьма обширен — по самым скромным оценкам. Соедините все эти программы с их политикой тщательного тестирования на всех поддерживаемых платформах, и станет понятно: то, что у них вообще остается время на релизы — это уже настоящее чудо. На нашем DVD нет места на все 14 ГБ, поэтому мы включили версию live (с программой установки). Диски live, доступные для скачивания, ограничены одним рабочим столом, а мы знаем, что наши читатели, как и остальной мир Linux, не могут прийти к соглашению о том, какой рабочий стол самый лучший. Поэтому мы использовали инструменты компиляции live Debian, чтобы обновить live DVD и включить в него пять разных сред рабочего стола, а именно: Cinnamon, Gnome, KDE, LXDE и Mate. Нам очень жаль, поклонники *Xfce*, но попытки добавить и его привели к конфликту с рабочим столом Gnome — но когда дистрибутив будет на вашем жестком диске, ничто не мешает вам установить *Xfce* самим.



› На экране приглашения можно выбрать желаемый рабочий стол.

DVD загружает рабочий стол Gnome. Чтобы попробовать другой рабочий стол, выйдите и снова войдите как пользователь **user** с паролем **live**, после выбора рабочего стола из меню в верхней правой части экрана приглашения. DVD всегда будет сначала загружать Gnome, но после установки Debian и выбора рабочего стола на экране приглашения тот станет вашим рабочим столом по умолчанию. Можно загрузить Debian live с **LXFDVD** и попробовать разные рабочие столы, но нельзя установить его в таком же виде из-за способа работы установщика Debian. Надо прожечь ISO-образ из директории Debian на DVD или записать на USB-брелок, загрузиться с этого носителя и выбрать опцию установки из меню загрузки. При затруднении с записью ISO-образа на диск или карту читайте FAQ по DVD, где есть и другая полезная информация, чтобы вы извлекли максимум пользы из своего **LXFDVD**.

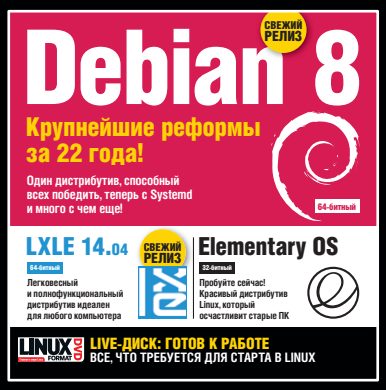


› Debian Jessie на базе KDE. Как и большинство программ Debian, он придерживается настроек по умолчанию.

» Важно ВНИМАНИЕ!

Порченые диски

В маловероятном случае какого-то дефекта вашего **LXFDVD** обращайтесь, пожалуйста, по адресу disks@linuxformat.ru или телефону +7 (812) 309-0686.



Новичок в Linux? Начните отсюда!

» Что такое Linux? Как его установить?
» Есть ли в нем эквивалент *MS Office*?
» Зачем нужна командная строка?
» Как устанавливать программы?

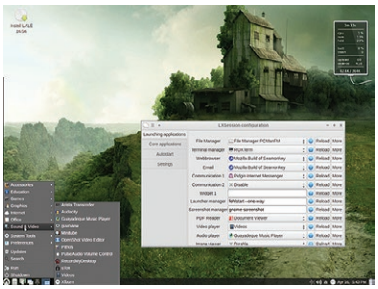
Ответы приводятся в Index.html на диске.

Индивидуальный LXDE

64-битный

LXLE 14.04.02

LXLE — производное Ubuntu на базе переделанного рабочего стола *LXDE*. Это отличает его от Lubuntu, который более плотно придерживается программ по умолчанию *LXDE*/Ubuntu. LXLE создан, чтобы вдохнуть жизнь в старые компьютеры, но хорошо ладит и с современным оборудованием, высвобождая его возможности для более полезной работы, чем обеспечение рабочего стола. Программы по умолчанию тоже иные: так, вместо обычных *Firefox* и *Thunderbird* взят пакет для работы в сети и электронной почты *Mozilla SeaMonkey*, плюс прекомпилированы добавочные PPA для установки большего количества программ.



Элегантный дистрибутив

32-битный

Elementary OS 0.3

Debian может считаться дистрибутивом для матерых линуксоидов. Не все попадают в эту категорию, да не все и стремятся туда попасть. Есть и новички в Linux, и те, кто им пользуется недавно, и им просто нужно то, что для них заработает — а очки в рейтинге умников им пока набирать неинтересно. У нас есть дистрибутив, идеальный для таких пользователей, elementary OS. Это не значит, что он годится исключительно для начинающих пользователей: как и большинство дистрибутивов, он будет расти вместе с вами. Интерфейс отличается привлекательностью и простотой — здесь нет никаких загадочных названий программ: простое и понятное меню приложений перечисляет большинство программ по их функциям, а не по имени. В отличие от остальных двух дистрибутивов на этом DVD, elementary OS является 32-битным, и это означает, что вы можете его установить почти на любую машину. Благодаря своей легковесности он особенно подходит для маломощных устройств, таких, как нетбуки.

И еще!

Системные инструменты

Главное

Checkinstall Установка tar-архива с помощью менеджера пакетов.

GNU Core Utils Основные утилиты, обязанные быть в каждой операционной системе.

Hardinfo Инструмент тестирования системы.

Kernel Исходный код самого последнего стабильного релиза ядра.

Memtest86+ Проверьте ОЗУ на предмет сбоев.

Plop Простой менеджер загрузки для запуска ОС с CD, DVD и USB.

RaWrite Создавайте загрузочные дискеты в MS-DOS в Windows.

SBM Независимый от ОС менеджер загрузки с несложным интерфейсом.

WvDial Соединяйтесь с Интернетом через телефонный модем.

Чтение

Книжная полка

Руководство по созданию скриптов Bourne Shell Начинайте осваивать скрипты оболочки.

Расширенное руководство по скриптам Bash Изучите написание скриптов еще лучше.

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.

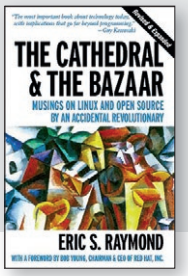
Введение в Linux Удобное руководство со множеством подсказок для начинающих пользователей Linux.

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.



LINUX FORMAT
Главное в мире Linux

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:

250 руб.

LXF196
Май 2015

» **100 программ-жемчужин** Без них вам не жить
» **Аноним Фигаро** Дистрибутивы заметают следы
» **Торификация** Упрячем свой трафик
» **Ядро Linux** Позови меня, позови...

LXFDVD: Mint 17.1, Kodi 14, Scientific Linux, Tiny Core, руководство по Android, 10 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_196/
PDF-версия: shop.linuxformat.ru/elxf_196/

250 руб.

LXF197
Июнь 2015

» **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/

250 руб.

LXF198
Июль 2015

» **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, Chromixium, 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/

А чтобы не упустить ничего, оформите подписку! Все, кто подписался на печатную версию журнала через www.linuxformat.ru/subscribe или www.linuxcenter.ru, получают электронную версию в подарок! На сайте shop.linuxformat.ru вы также сможете приобрести предыдущие выпуски **LXF**.

Подписывайтесь на сайте www.linuxformat.ru/subscribe

Телефоны отдела подписки:
» Санкт-Петербург (812) 309-0686
» Москва (499) 271-4954

Linux Format ВКонтакте

Вступайте в нашу группу **vk.com/linuxform**

На странице LXF ВКонтакте вы найдете:
» Новости о Linux
» Статьи из архива **LXF**
» Анонс свежего выпуска **LXF** и часть статей из него
» Живое общение и онлайн-консультацию по подписке на наше издание

DVD-приложение к журналу Linux Format № 8 (199). Август 2015

» Содержание

LINUX FORMAT

Сторона 1

ДИСТРИБУТИВЫ*
Debian 8 (64-битный)
Elementary OS (32-битный)
LXLE 14.04.2 (64-битный)

НОТРICKS
DNSCrypt DNS-прокси
GNU Nano Текстовый редактор
Guake Терминал
Minilens Головоломка
Plank Док
QmapShack Инструмент карт GPS
Tor Browser Анонимный web-браузер
Vera Среда рабочего стола
ZFSonLinux Файловая система

ПОМОЩЬ
Руководство новичка
Справочники
Ответы

СРАВНЕНИЕ: КЛИЕНТЫ БЫСТРОГО ОБМЕНА СООБЩЕНИЯМИ
Empathy
Jitsi
Kopete
Pidgin
Psi

УЧЕБНИКИ
Python Dice Simulator

ГЛАВНОЕ
CheckInstall
Coreutils
Hardinfo
Kernel
Memtest86+
Plop
SBM
WvDial

* Запускаются прямо с DVD. Перезагрузитесь с DVD и выберите имя дистрибутива в загрузочном меню.

Окончание на обороте »

Информация о диске

Внимательно прочтите это перед тем, как использовать LXFDVD!

Что-то потеряли?

Часто случается, что новые программы зависят от других программных продуктов, которые могут не входить в текущую версию вашего дистрибутива Linux.

Мы стараемся предоставить вам как можно больше важных вспомогательных файлов. В большинстве случаев, последние версии библиотек и другие пакеты мы включаем в каталог «Essentials [Главное]» на прилагаемом диске. Поэтому, если в вашей системе возникли проблемы с зависимостями, первым делом следует заглянуть именно туда.

Форматы пакетов

Мы стараемся включать как можно больше различных типов установочных пакетов: RPM, Deb или любых других. Просим вас принять во внимание, что мы ограничены свободным пространством и доступными двоичными выпусками программ. По возможности, мы будем включать исходные тексты для любого пакета, чтобы вы могли собрать его самостоятельно.

Документация

На диске вы сможете найти всю необходимую информацию о том, как устанавливать и использовать некоторые программы. Пожалуйста, не забывайте, что большинство программ поставляются вместе со своей документацией, поэтому дополнительные материалы и файлы находятся в соответствующих директориях.

Что это за файлы?

Если вы новичок в Linux, вас может смутить изобилие различных файлов и расширений. Так как мы стараемся собрать как можно больше вариантов пакетов для обеспечения совместимости, в одном каталоге часто находятся два или три файла для различных версий Linux и различных архитектур, исходные тексты и откомпилированные пакеты. Чтобы определить, какой именно файл вам нужен, необходимо обратить внимание на его имя или расширение:

» **имя_программы-1.0.1.i386.rpm** — вероятно, это двоичный пакет RPM, предназначенный для работы на системах x86;
» **имя_программы-1.0.1.i386.deb** — такой же пакет, но уже для Debian;
» **имя_программы-1.0.1.tar.gz** — обычно это исходный код;
» **имя_программы-1.0.1.tgz** — тот же файл, что и выше этажом по списку: “tgz” — это сокращение от “tar.gz”;
» **имя_программы-1.0.1.tar.bz2** — тот же файл, но сжатый bzip2 вместо обычного gzip;
» **имя_программы-1.0.1.src.rpm** — также исходный код, но поставляемый как RPM-пакет для упрощения процесса установки;
» **имя_программы-1.0.1.i386.FC4.RPM** — двоичный пакет RPM для x86, предназначенный специально для операционной системы Fedora Core 4;
» **имя_программы-1.0.1.ppc.Suse9.rpm** — двоичный пакет RPM, предназначенный специально для операционной системы SUSE 9.x PPC;
» **имя_программы-devel-1.0.1.i386.rpm** — версия для разработчиков.

Если диск не читается...

Это маловероятно, но если все же прилагаемый к журналу диск поврежден, пожалуйста, свяжитесь с нашей службой поддержки по электронной почте: disks@linuxformat.ru

Создание установочных дисков при помощи cdrecord

Самый быстрый способ записать ISO-образ на чистую матрицу — это обратиться к программе *cdrecord*. Для всех перечисленных ниже действий потребуются права суперпользователя-root. Сначала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdrecord -scanbus
```

После этого на экране терминала должен отобразиться список устройств, подключенных к вашей системе. SCSI-адрес каждого устройства представляет собой три числа в левой колонке — например, 0,3,0. Теперь вы можете с легкостью записать образ на диск:

```
cdrecord dev=0,3,0 -v /путь к образу/image.iso
```

Чтобы упростить дальнейшее использование *cdrecord*, сохраните некоторые настройки в файле **/etc/default/cdrecord**. Добавьте по одной строке для каждого устройства записи (скорее всего, в вашей системе присутствует только одно такое устройство):

```
Plextor= 0,3,0 12 16M
```

Первое слово в этой строке — метка; затем после адреса SCSI-устройства вы должны указать скорость и размер буфера. Теперь можете заменить SCSI-адрес в командной строке на выбранную вами метку. Все будет еще проще, если вы добавите следующее:

```
CDR_DEVICE=Plextor
```

Для записи ISO-образа вам осталось набрать команду

```
cdrecord -v /path/to/image.iso
```

Если вы не принадлежите к любителям командной строки, в таком случае вам придет на помощь утилита *gcombust*. Запустите ее от имени root и выберите вкладку Burn и ISO 9660 Image в верхней части окна. Введите путь к образу, который вы хотите записать на диск, и смело нажимайте на Combust! Пока ваш образ пишется на диск, можете выпить чашечку кофе.

Другая ОС?

Использовать Linux для записи компакт-диска не обязательно. Программы вроде *cdrecord* просто переносят двоичные данные на чистую матрицу. Все необходимые файлы уже включены в ISO-образ, который распознается любой операционной системой, будь то Linux, Windows, Mac OS X или AmigaOS.

Нет устройства для записи дисков?

Если у вас нет устройства, с помощью которого можно было бы записать образ на диск, можно найти какого-нибудь друга или организацию, у кого есть компьютер с дисководом, и прожечь диск у них. Опять-таки, вам подойдет любая операционная система, способная распознать пишущий привод (см. выше).

Некоторые дистрибутивы умеют монтировать образы дисков и выполнять сетевую установку или даже установку с раздела жесткого диска. Конкретные методы, конечно, зависят от дистрибутива. За дополнительной информацией обращайтесь на web-сайт разработчика дистрибутива.

DVD-приложение к журналу Linux Format №8 (199). Август 2015

» Содержание

LINUX FORMAT

Сторона 2

ДОКУМЕНТАЦИЯ: 10 КНИГ О LINUX (НА АНГЛИЙСКОМ ЯЗЫКЕ)

Bourne Shell Scripting Начальное руководство по программированию на Bash
Advanced Bash Scripting Guide Подробное руководство по программированию на Bash
The Cathedral and the Bazaar Классический текст Эрика Реймонда [Eric S Raymond] «Собор и базар»
System Administrators Guide Руководство по базовому администрированию Linux
GNU Tools Summary Руководство по работе в командной строке и обзор основных утилит GNU
The Debian Book Руководство администратора, написанное разработчиками Debian
Dive Into Python Учебник по программированию на Python
Intro to Linux Начальное руководство по Linux
Linux Dictionary Словарь Linux, объясняющий специфическую терминологию
Linux Kernel in a Nutshell Описание ядра Linux, созданное одним из его выдающихся разработчиков — Грегом Кроа-Хартманом [Greg Kroah-Hartman]
System Administrators Guide Руководство по базовому администрированию Linux

ДИСТРИБУТИВЫ

4MLinux 13.0 (32-битный, ISO-образ)
Antergos 2015.06.03 (32- и 64-битный, ISO-образ)
Chromixium OS 1.5 (32-битный, ISO-образ)
Linux Mint 17.2 Mate (64-битный, ISO-образ)

Пожалуйста, перед использованием данного диска ознакомьтесь с инструкцией, опубликованной в журнале на стр. 109!

КОММЕНТАРИИ? Присылайте ваши пожелания и предложения по электронной почте: info@linuxformat.ru

ДЕФЕКТНЫЕ ДИСКИ В маловероятном случае обнаружения дефектов на данном диске, обращайтесь, пожалуйста, по адресу disks@linuxformat.ru

Настоящий диск тщательно тестировался и проверялся на всех стадиях производства; однако, как и в случае с любым новым ПО, мы рекомендуем вам использовать антивирусный сканер. Мы также рекомендуем всегда иметь под рукой актуальную резервную копию данных вашего жесткого диска.
К сожалению, редакция Linux Format не в состоянии принимать на себя ответственность за любые повреждения, разрушения или иные убытки, которые может повлечь за собой использование этого DVD, представленных на нем программ или данных. Прежде чем устанавливать какое-либо ПО на компьютер, подключенный к сети, проконсультируйтесь с сетевым администратором.

Тираж изготовлен ООО «Маркон», 188652, Россия, Ленинградская область, Всеволожский р-н, дер. Юкки, Школьная ул., 7-а. Лицензия МПТР ВАФ N 77-103.

Есть вопросы?

Задавайте!

http://unixforum.org/

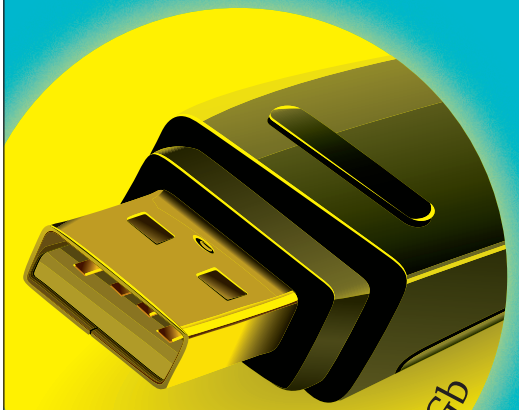
Отдел дистрибьюции ГНУ/Линуксцентра

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
HA USB FLASH



Linux Mint 16

Ubuntu 14.04

USB Flash 8 Gb

А ТАКЖЕ версии для юридических лиц — с лицензионным договором присоединения

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 сентябрьском номере

200 раз / 10 лет: 200 советов

А вы что будете делать, когда вам стукнет 200?
Мы — праздновать!

Linux тогда и теперь

Обозреваем 15 и 10 лет Linux Format и разработки ядра Linux. Включайте ностальгию!

Ядро Linux 4.0

Отставим прошлое и заглянем в будущее! Фантастические функции грядущих релизов ядра.

Открытые ОС

GNU/Linux — не единственная в мире; рассматриваем альтернативы и поясняем, зачем вообще брать их в голову.

Содержание будущих выпусков может меняться — вдруг нас вытурят на пенсию.

LINUX FORMAT
Главное в мире Linux

Журнал зарегистрирован Федеральной службой по надзору за соблюдением законодательства в сфере массовых коммуникаций и охране культурного наследия ПИ № ФС77-21973 от 14 сентября 2005 года.
Выходит ежемесячно. Тираж печатной версии 2000 экз., распространение электронной версии 30 000 экз.

РЕДАКЦИЯ РУССКОЯЗЫЧНОЙ ВЕРСИИ
Главный редактор
Кирилл Степанов info@linuxformat.ru
Литературный и выпускающий редактор
Елена Толстякова
Переводчики
Елена Ессяк, Даниил Кривошеин, Светлана Кривошеина, Валентин Развозжаев, Елена Толстякова
Редактор диска
Александр Баракин
Верстка, допечатная подготовка
Сергей Рогожников
Технический директор
Андрей Смирнов
Директор по рекламе
Владимир Савельев advert@linuxformat.ru
Генеральный директор
Павел Фролов
Учредители
Частные лица
Издатель
ООО «Линукс Формат»

Отпечатано в типографии ООО «ЛД-ПРИНТ»
196644, Санкт-Петербург, Колпинский р-н, пос. Саперный, территория предприятия «Балтика», д. б/н, лит. Ф
Тел. (812) 462-8383, e-mail: office@ldprint.ru
Заказ 12842

РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ
Редактор Нейл Мор [Neil Mohr] neil.mohr@futurenet.com
Научный редактор Джонни Бидвелл [Jonni Bidwell] jonni.bidwell@futurenet.com
Выпускающий редактор Крис Торнетт [Chris Thornett] chris.thornett@futurenet.com
Художественный редактор Эфраин Эрнандес-Мендоса [Efrain Hernandez-Mendoza] efrain.hernandez-mendoza@futurenet.com

ПОДГОТОВКА МАТЕРИАЛОВ
Джонни Бидвелл [Jonni Bidwell], Нейл Ботвик [Neil Bothwick], Джолион Браун [Jolyon Brown], Крис Браун [Chris Brown], Шон Конвэй [Sean Conway], Марко Фиоретти [Marco Fioretti], Мэтью Хэнсон [Matthew Hanson], Нейл Мор [Neil Mohr], Чуонг Нгуен [Chuong Nguyen], Кэрри-Энн Филбин [Carrie-Anne Philbin], Лес Паундер [Les Pounder], Алан Стивенс [Alan Stevens], Александр Толстой [Alexander Tolstoy], Михалис Цукалос [Mihalis Tsoukalos], Евгений Балдин, Андрей Гондаренков, Алексей Федорчук, Михаил Чеботарёв, Игорь Штомпель

Иллюстрации Шейн Коллинджа [Shane Collinge]

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 неэксклюзивное право на публикацию и лицензирование всех присланных материалов, если не было оговорено иное. Linux Format стремится оставлять уведомление об авторских правах всюду, где это возможно. Свяжитесь с нами, если мы не упомянули вас как автора предложенных вами материалов, и мы постараемся исправить эту ошибку. Редакция Linux Format не несет ответственности за опечатки.

Ответственность за содержание статьи несет ее автор. Мнение авторов может не совпадать с мнением редакции.

Все присланные материалы могут быть помещены на диски — CD или DVD, поставляемые вместе с журналом, если не было оговорено иное.

Ограничение ответственности: используйте все советы на свой страх и риск. Ни при каких условиях редакция Linux Format не несет ответственность за повреждения или ущерб, нанесенные вашему компьютеру и периферии вследствие использования тех или иных советов.

Linux — зарегистрированный товарный знак Линуса Торвальдса [Linus Torvalds]. "GNU/Linux" заменяется на "Linux" в целях сокращения. Все остальные товарные знаки являются собственностью их законных владельцев. Весь код, опубликованный в журнале, лицензирован на условиях GPL v3. См. www.gnu.org/copyleft/gpl.html

За информацией о журналах, издаваемых Future plc group company, обращайтесь на сайт www.futureplc.com.

© Linux Format 2005
© Future Publishing Ltd 2005

Future MEDIA WITH PASSION
BATH • LONDON • MILAN • NEW YORK • PARIS • SAN DIEGO • SAN FRANCISCO

16+

Добро пожаловать в робототехнику!

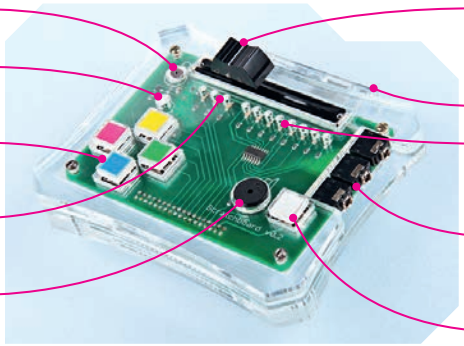
ScratchDuino

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.



* РАБОТАЮТ В РЕЖИМЕ ЭМУЛЯЦИИ PICOBOARD, НЕ ТРЕБУЯ ДОПОЛНИТЕЛЬНОГО ПРОГРАММИРОВАНИЯ.

ЦЕНА **9500 руб.**

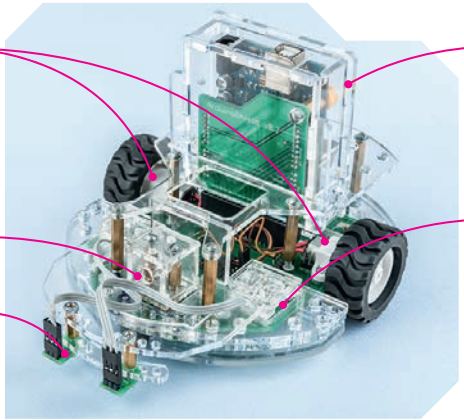
ГАРАНТИЯ 3 ГОДА, ВКЛЮЧАЯ ТЕХНИЧЕСКУЮ И МЕТОДИЧЕСКУЮ ПОДДЕРЖКУ

ScratchDuino.Робоплатформа

Внешний робот-исполнитель, управляемый из среды программирования Scratch. Не требует навыков программирования на языках высокого уровня и может применяться для обучения, начиная с младших классов.

ScratchDuino.Робоплатформа обеспечивает возможность:

- подключения внешних датчиков без механических соединений и пайки;
- расширения через последовательную шину RoboBus;
- установки деталей Lego Technics.



* ВСЕГО НА РОБОПЛАТФОРМЕ 5 РАЗЪЕМОВ. ВОЗМОЖНО ПОДКЛЮЧЕНИЕ ДАТЧИКОВ КАСАНИЙ, ИК-ДАТЧИКОВ РАССТОЯНИЙ/ПРЕПЯТСТВИЙ И ДР.

ЦЕНА **19 500 руб.**

ГАРАНТИЯ 3 ГОДА, ВКЛЮЧАЯ ТЕХНИЧЕСКУЮ И МЕТОДИЧЕСКУЮ ПОДДЕРЖКУ

Назначение ScratchDuino

- Изучение взаимодействия компьютера с внешней средой
- Изучение изменения параметров внешней среды
- Изучение процессов передачи информации и принципов ее построения
- Изучение внешних устройств управления
- Моделирование устройств

Среда разработки

Lazarus (язык Pascal) ■ Scratch ■ Arduino IDE

ScratchDuino адаптирован для учебных заведений. Поставляется с комплектами учебно-методических материалов.

Продукт разработан при финансовой поддержке Фонда содействия развитию малых форм предприятий в научно-технической сфере

WWW.SCRATCHDUINO.RU

(812) 309-0686 (Санкт-Петербург) ■ **(499) 271-4954** (Москва)

HETZNER DEDICATED SERVERS

HETZNER ONLINE

В ЦЕЛОСТИ И СОХРАННОСТИ!

ХРАНЕНИЕ ДАННЫХ В ГЕРМАНИИ

ВЫДЕЛЕННЫЙ СЕРВЕР PX60

- Intel®Xeon® E3-1270 v3 Quadcore Haswell
- 32 ГБ ECC RAM
- 2 x 2 ТБ 6 Гбит/с SATA 7200 об/м Enterprise класс Software-RAID 1
- 30 ТБ Трафик*
- Операционная система Linux
- Подключение 1 Гбит порт
- Гарантировано 200 Мбит/с
- IPv6 подсеть (/64)
- Без минимального контракта

2600 рублей в месяц
+ Установка 3700 рублей

intel inside XEON

ВЫДЕЛЕННЫЙ СЕРВЕР PX70-SSD

- Intel®Xeon® E3-1270 v3 Quadcore Haswell
- 32 ГБ ECC RAM
- 2 x 480 ГБ 6 Гбит/с SATA SSD Data Center Series
- 50 ТБ Трафик*
- Операционная система Linux
- Подключение 1 Гбит порт
- Гарантировано 200 Мбит/с
- IPv6 подсеть (/64)
- Без минимального контракта

3700 рублей в месяц
+ Установка 3700 рублей

intel inside XEON

* Нет платы за превышение. При превышении 30 ТБ/месяц (PX60), 50 ТБ/месяц (PX70-SSD) скорость соединения ограничивается (подсчёт ведётся по исходящему трафику, входящий и внутренний трафик не учитывается). Опционально можно снять ограничение, подтвердив оплату 80 рублей за каждый дополнительный ТБ.

Безопасное защищённое хранение ваших данных в собственных дата-центрах немецкой компании Hetzner Online, соответствующее высоким требованиям к стандартам защиты данных в Германии.

RU.HETZNER.COM

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!

Все цены указаны без учёта НДС и регулируются условиями предоставления услуг компанией Hetzner Online AG. Цены могут быть изменены. Все права защищены соответствующими производителями.