



LINUX FORMAT

Главное в мире Linux

Июнь 2013 № 6 (171)

MEGA-DVD С ИГРАМИ
Эксклюзивный набор игр для Linux
плюс OpenSUSE, Trisquel, Kali и более того!



Грузим новые ПК
» Навигация по коварным
водам Secure Boot **с. 40**



СИСТЕМ

ПРОВЕРЕНО!

Открываем лучшие дистрибутивы 2013 года в нашем супер-сравнении

Inkscape Рисуем векторные картинки
Turnkey Linux Сервер на халюву
Резервное копирование Сделайте сейчас!

ПЛЮС!
Шагаем в светлое
будущее
со стандартом OpenCL
с. 86



OpenSUSE 12.3
» Истинно немецкий
сверхнастраиваемый дистрибутив **с. 14**



Linux предустановлен
« Тут нет доброжелательности. Нет убеждения в правоте. Только бизнес »
Кэти Малмроуз — про мир оборудования с. 36

Также в номере...

Внутри Ardour 3.0
Записывайте, редактируйте и микшируйте музыку в Linux **с. 44**

Редакторы изображений
Не один GIMP в поле воин... есть и другие неплохие варианты **с. 20**

Без Dropbox, но в Unison
Сами синхронизируем свои компьютеры **с. 78**



Управление медиа-файлами
Спишем музыку
» Копируйте CD на жесткий диск, не нарушая закон

Контроль качества
Улучшим приложения
» Не код, а чистое золото; но надо отшлифовать

Облачные вычисления
Изучаем AWS
» Ваш Linux-ПК и Amazon: вместе вы — легион

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



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

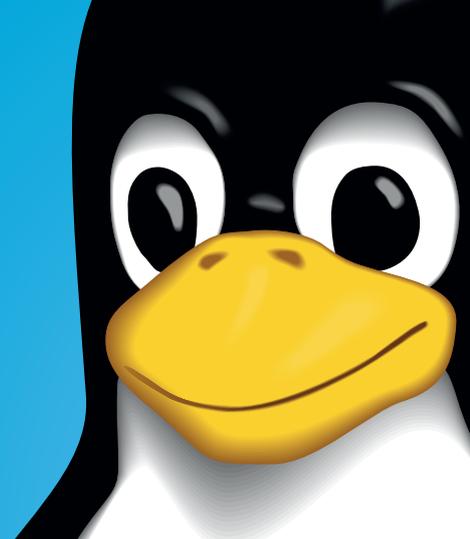
Санкт-Петербург
+7 (812)

309-06-86

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

Linux  center

Используйте свободное ПО — сэкономьте годовой бюджет!



Операционная система GNU/Linux поможет вам с **наименьшими затратами** решить проблему лицензирования программного обеспечения, навсегда избавиться от компьютерных вирусов и повысить надежность вашей компьютерной сети.



С нашей помощью вы сможете

Сконцентрироваться на своем бизнесе, не отвлекаясь на вопросы поддержки своей ИТ-инфраструктуры



Забывать о вирусах, угрозах безопасности и необходимости лицензирования программного обеспечения



Оптимизировать затраты на лицензирование ПО за счет максимально возможного использования свободного ПО

ГНУ/Линуксцентр предлагает:

- внедрение наиболее дружелюбных вариантов ОС GNU/Linux и прикладных решений на базе свободного ПО;
- абонентскую поддержку вашей сети;
- обучение сотрудников вашей компании.

Наш опыт внедрения свободного программного обеспечения в организациях различного профиля поможет выбрать **оптимальное сочетание свободного и коммерческого программного обеспечения**, подходящее именно для вашей компании, а также поможет избежать технических и организационных проблем при внедрении свободного ПО.

Решите проблемы лицензирования ПО и поддержки компьютерной сети с помощью профессионалов!

Москва
+7 (499)

271-49-54

Санкт-Петербург
+7 (812)

309-06-86

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

Linux  center

НАЙДИ РАБОТУ ЛЕГКО!

на www.hh.ru



САЙТ РАЗРЕШЁН ДЛЯ ПОСЕЩЕНИЯ БЕЗ ОГРАНИЧЕНИЯ ВОЗРАСТА

hh **ru**
HeadHunter

Выбирай из более чем 200 000 вакансий

Что мы делаем

» Мы поддерживаем открытое сообщество, предоставляя источник информации и площадку для обмена мнениями.

» Мы помогаем всем читателям получить от Linux максимум пользы, публикуя статьи в разделе «Учебники»: здесь каждый найдет что-то по своему вкусу.

» Мы выпускаем весь код, появляющийся на страницах раздела «Учебники», по лицензии GNU GPLv3.

» Мы стремимся предоставлять точные, актуальные и непредвзятые сведения обо всем, что касается Linux и свободного ПО.



Кто мы

В данном номере мы восхищаемся неизменным разнообразием Ubuntu. Но чего все-таки не хватает у нас в экосистеме? Какой дистрибутив создали бы вы?



Гэри Уокер
Коли и Китай завел свой госдистрибутив, графство Ратленд тоже такое может. Сисадмины из Оуксэма, за дело!



Эндрю Грегори
Мир взывает к очередной легковесной разновидности Ubuntu, на базе LXDE. Ох, нет...



Эфраин Эрнандес-Мендоса
Ни один дистрибутив не поддерживает ацтекский календарь. Внедрим его и принесем жертвы Солнцу.



Бен Эверард
Велпих: дистрибутив мешковатых твидовых пиджаков, махорки и интернационал-социализма.



Маянк Шарма
Есть у меня диск Ubuntu 8.10, и я им премного доволен, спасибо. Эх, денечки были...



Джонатан Робертс
Kernovix: это Fedora, но с поддержкой корнваллийского языка. Пойдет на-ура в Марационе, древней столице Корнуолла.



Майк Сондерс
С чего это все вокруг жужжат про какой-то Linux? Будущее всего мира — MikeOS, ха-ха.



Валентин Синицын
Я соскучился по дистрибутивам, имитирующим внешний вид Windows XP. Так что даешь Win8buntu!



Ник Вейч
Мне нравится Ubuntu для сатанистов, но в нем маловато Зла. Добавить бы туда адский KDE 4 для полноты картины.



Сюзан Линтон
Live-дистрибутив с предустановленными виртуальными машинами еще на 49 других. Милости просим в кроличью нору...



Шашанк Шарма
Такой, чтобы уметь блокировать ненавидимый голос диктора при прямой потоковой трансляции Radio 2.



Нейл Ботвик
Как элитный пользователь Gentoo, я и так уже имею идеальный вариант. И на это у меня ушло всего три года!



Технология, изменившая мир

» На страницах этого номера вы найдете сравнение 50 наиболее популярных дистрибутивов по версии LXF. Каждый из них по-своему уникален и имеет свою область применения, но опирается на общий набор базовых технологий, без которых Linux не появился бы на свет.

Если задуматься, то свободное ПО было бы невозможно без открытых стандартов, таких как POSIX или X11, открытого стека сетевых протоколов поверх TCP/IP и открытых форматов файлов. Вспомним и о «железных» стандартах, которые обеспечивают совместимость аппаратуры различных изготовителей.

Одному из таких стандартов — Ethernet — только что исполнилось 40 лет. 22 мая 1973 года аспирант Роберт Мэткаф [Robert Metcalfe] предложил использовать «общую полностью пассивную среду передачи электромагнитных волн» для связи между компьютерами и протокол CSMA/CD для доступа к ней. Международный стандарт ISO/IEEE 802.3 приняли лишь через десять лет, но за последующие 30 открытый Ethernet одержал полную и безоговорочную победу над стандартами проприетарными: некогда популярные TokenRing от IBM или TokenBus от General Motors ныне почти забыты. За эти годы скорость передачи данных по Ethernet возросла на три порядка, с 10 мегабит до 10 гигабит в секунду. Идут работы по стандартизации 100-гигабитного протокола, и активно обсуждается терабитный. Отличный пример технологии, продолжающей изменять мир!

Кирилл Степанов
Главный редактор
» info@linuxformat.ru

Как с нами связаться

Письма для публикации: letters@linuxformat.ru
Подписка и предыдущие номера: subscribe@linuxformat.ru
Техническая поддержка: answers@linuxformat.ru
Общие вопросы: info@linuxformat.ru
Вопросы распространения: sales@linuxformat.ru
Web-сайт: www.linuxformat.ru
Группа «ВКонтакте»: vk.com/club53383882

» **Адрес редакции:** Россия, Санкт-Петербург, Лиговский пр., 50, корп. 15
» **Телефон редакции:** (812) 309-06-86. Дополнительная информация на с. 112

Содержание

Пятьдесят оттенков Linux.

Обзоры

OpenSUSE 12.3 14

Приглядный и крепко сбитый релиз — в нем вас не ждут неприятные сюрпризы.



➤ Нам понравился новый темный экран OpenSUSE.

Crusader Kings II 16

Напряженная игра-стратегия, полная семейных интриг и европейских завоеваний.



➤ Строим империю гигантов! Еще три шага — и враг опрокинется в воду.

TeamViewer 17

Проприетарное кросс-платформенное приложение, идеальное и для новичков, и для зрелых профи.

Какой дистрибутив вам больше подходит? Здесь полсотни лучших, по рангам и разрядам с. 26

Сравнение: Редакторы изображений с. 20

Работа над ошибками

В майский номер (LXF170) вкралась досадная ошибка: на стр. 84 под фотографией во врезке «Наш эксперт» следует читать:

Андрей Ушаков

Активно приближает тот день, когда функциональные языки станут мейнстримом.

Редакция *Linux Format* приносит свои извинения автору материала.

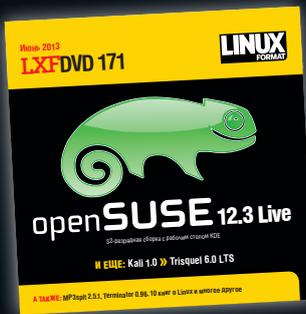
Люди говорят



«Это так непохоже на сообщество свободного кода — совершенно иной мир»

Кэти Малпроуз — о методах Microsoft с. 36

На вашем бесплатном DVD



OpenSUSE 12.3

» RPM-дистрибутив номер 1!

Trisquel

» Linux высшей степени свободы

Плюс: HotPicks и коды учебников **с. 104**



Ищите в этом номере



Secure Boot 40

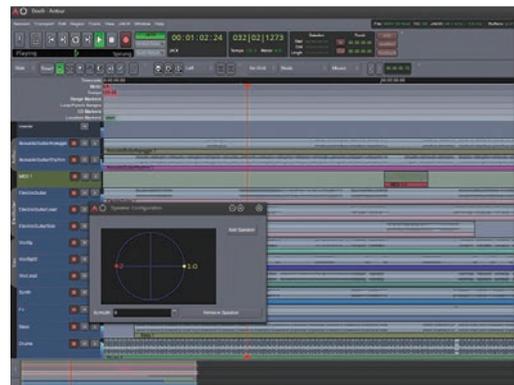
День грядущий готовит нам подлянку?

Ardour 44

Лучшему аудиоприложению нужна помощь.

Достижения Ubuntu 48

Этого достиг не кто-нибудь, а лично вы.



Пропустили номер?

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



Постоянные рубрики

Новости 6

Ubuntu упрощает формат пакетов, NASA переходит на Debian, продаются телефоны с Firefox OS, а Россия готовится к электронной демократии и рвется в Интернет.

Новости Android 18

Android развезжает на авто и лидирует в России, а Google предоставил для него среду разработки.

Сравнение 20

Долой блузы, кисти и мольберты — обойдемся достойным редактором изображений.

Интервью LXF 36

Сыновья Кэти Малпроуз достали ее «голубыми экранами смерти» — пришлось ей основать компанию по компьютерам с Linux.

Что за штука 48

Набрав Ubuntu Accomplishments, получаем призы сообщества.

Рубрика сисадмина 50

Пустим порезвиться Tomcat (бойся, мышь!), немножко поизучаем Java и напишем сервлет.

Ответы 94

ВАШИ ПРОБЛЕМЫ РЕШЕНЫ!

Нейл Ботвик — про потерянные терминалы, телевидение на Raspberry Pi, беспроводные сети, непознаваемые файлы и прочее.

Hotpicks 98

Отведайте горяченького: лучшие в мире новинки свободного ПО. Tiled, Mosh, Mp3split, i7z, Terminator, JSesh, The Silver Searcher, Irlflamb, QNetWalk, Halttimer, Torsocks.

Диск Linux Format 104

Содержимое двустороннего DVD этого месяца.

Пропустили номер? ... 108

Коллекция неполна? Если на вашей полке с журналами образовалась ниша размером с LXF, вы знаете, куда обратиться.

Через месяц 112

Что утратил Ubuntu? Разберемся, почему все пошло наперекосяк.

Учебники

Raspberry Pi Командная строка 56

Руководство для самых маленьких.

Работа с медиа-данными Обдерем свои CD 60

Запишем любимые мелодии на жесткий диск.

Режимы ПК Спи, усни 64

Дадим компьютеру отдых, не выключая его.

Облачные вычисления Web-сервисы Amazon 66

Спихнем расчеты на удаленный дата-центр.

Управление программами Тестирование 70

Пусть-ка программы работают как полагается.

Редактирование графики Рисуем в Inkscape 74

SVG-изображения по силам и не художникам.

Облачное хранение Файлы в Unison 78

Обойдемся без проблем с собственностью.

Виртуализация Turnkey Linux 82

Развернем web-приложения парой щелчков.

Параллельные вычисления OpenCL решает все 86

Накроем все устройства одним стандартом.

Резервное копирование Силач Bacula 90

Запустим сервер резервного копирования.



ГЛАВНОЕ Новый игрок » Пакеты Canopical » Космические ноутбуки
» Демократия и СПО » Россия в Интернете » Дан старт продажам

ПРЕЗЕНТАЦИЯ СМАРТФОНА

Ария китайского гостя

Смартфоны OPPO теснят в России известных флагманов.



» Рубрику готовил
АРТЕМ ЗОРИН

Рынок смартфонов радует тем, что никак не желает становиться «застойным». В компьютерной сфере сильнейшие игроки определены давно, но в стане телефонов все не так просто: ежегодно дают знать о себе новые перспективные компании, в короткие сроки (вспомним ZTE и Huawei) взлетающие к вершинам рейтингов. Так и планы руководства OPPO по вхождению в первую пятерку «строителей» умных телефонов кажутся не столько амбициозными, сколько вполне реалистичными.

Особенность положения бренда в том, что по техническим возможностям он не уступает гигантам «смартфоностроения», но чтобы добиться узнаваемости, еще предстоит вложить немало усилий и денег. В OPPO не скупятся на брендинг — в 2012 году на эту статью ушло \$300 млн, и начинают серьезно заявлять о себе за пределами азиатского региона. Приятно, что Россия вошла в число стран, где смартфоны OPPO будут продаваться официально.

Присутствие в нашей стране OPPO ознаменовала крупной пресс-конференцией в Москве 24 апреля — в культовом (как утверждалось в приглашении) клубе ICON Club, с присутствием порядка сотни журналистов. Лейтмотивом встречи считалась презентация флагманского смартфона

OPPO Find 5. Собравшиеся вдобавок получили информацию о бренде OPPO и особенностях его работы в России.

Новинка интересна уже тем, что по характеристикам ничуть не уступает флагманам начала 2013 года от компаний вроде Samsung. Соблюдены все «приличия»: 5-дюймовый экран формата Full HD, нержавеющая сталь в отделке корпуса, камера на 13 Мп. Аппаратная платформа та же, что и в HTC One и Sony Xperia Z.

Конечно, бесполезно стремиться в сегмент премиум-смартфонов без ярких особенностей. Поэтому дисплей в OPPO Find 5 выполнен с отсутствием воздушного слоя между матрицей и сенсорной панелью. Вдобавок поддерживается опция Wi-Fi Display для вывода картинки «по воздуху» на телевизор. Примечательна реализация технологии Dirac HD Audio, представленной связкой программной и аппаратной обработки звука — та же технология применяется в аудиосистемах автомобильных премиум-марок Rolls-Royce и Bentley. Впечатления от использования OPPO Find 5 в качестве плеера также весьма приятные. Завершим рассказ о модели указанием ее цены. 19990 руб. за 16-гигабайтный и 22990 руб. за 32-гигабайтный варианты говорят сами за себя — в пику «30-тысячникам» Samsung, Sony и HTC.



» **Стильный дизайн и впечатляющая производительность — вот главные козыри OPPO.**

В поддержке пользователей OPPO постаралась максимально избавиться от телефонной «бюрократии». В российском контакт-центре бренда нет голосового меню и, что обычно раздражает гораздо больше, перенаправления от одного специалиста к другому: поднявший трубку занимается проблемой пользователя «от и до». Существует еще и отдельный телефон для желающих добыть информацию о смартфонах OPPO. В общем, колл-центр OPPO неспроста называется не иначе как «COOL Center». Техподдержка бренда дополнительно трудится в соцсетях Facebook и ВКонтакте, а также на форумах крупных сайтов вроде 4PDA.

OPPO на удивление открыт в информационном плане — журналистам в конкретных цифрах рассказали и о глобальной работе бренда, и о планах по России, где в 2013 году намерены продать 200 тыс. смартфонов, а в 2015 довести показатель до 1,5 млн. С июня–мая в продажу ежемесячно будет поступать новая модель, и к концу года ассортимент дойдет до 8 смартфонов. На продвижение бренда в России на 2013 год выделено до \$5 млн. Не исключено, что к рекламе смартфонов OPPO привлекут российских «селебрити». К слову, в прошлом году лицом бренда в Китае был Леонардо Ди Каприо.

OPPO как производитель

Хорошее представление об уровне OPPO дает информация о наличии собственного завода с 1400 инженерами и 6 тысячами рабочих на сборочных линиях. Оборот компании за 2012 год достиг \$1,8 млрд, а по итогам того же года доля OPPO на рынке смартфонов Китая установилась на отметке в 6%. Делать прогнозы о вероятности успеха OPPO в России пока рано; хотя

объективно не видно особых препятствий завоеванию брендом «места под солнцем». У OPPO есть привлекательные смартфоны, ответственная техподдержка и, что заметно, желание завоевать доверие потребителей. Причем представители OPPO прямо говорят, что в первые годы бренд будет работать в стране на репутацию, а не на скорейшее завоевание некой доли рынка.

1С:ПРЕДПРИЯТИЕ 8

Новое поколение систем автоматизации управления и учета

- 1С:Управление производственным предприятием
- 1С:Бухгалтерия
- 1С:Зарплата и управление персоналом
- 1С:Управление торговлей
- Отраслевые и специализированные решения

Теперь работает в Linux!

Для запуска «1С:Предприятия» в Linux необходимо наличие WINE@Etersoft, который входит в состав дистрибутива Mandriva Linux 2011 Powerpack



Москва
+7 (499)

271-49-54

Санкт-Петербург
+7 (812)

309-06-86

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



НА ПУТИ К УНИФИКАЦИИ

Ubuntu — универсальная ОС

Canonical развивает свой формат пакетов для установки сторонних приложений.

Колин Ватсон [Colin Watson], глава команды разработчиков инсталлятора Ubuntu, представил проект создания нового формата пакетов и связанного с ним инструментария. Проект ориентирован на предоставление разработчикам сторонних приложений возможности быстрой подготовки пакетов, устанавливаемых пользователем одним щелчком без манипуляций с подключением репозитория. Новый формат в первую очередь предназначен для упаковки приложений, создаваемых при помощи средств разработки Ubuntu SDK для мобильных версий Ubuntu Touch.

Поддержка менеджера пакетов *dpkg*, инструментария *apt* и текущей организации работы репозитория пакетов сохранится в полном объеме — новый проект нацелен только на вспомогательные задачи упрощения распространения программ, не входящих в базовые репозитории. Новый формат не требует от разработчиков изучения особенностей сборки и распространения Deb-пакетов: все операции по формированию пакетов будут полностью автоматизированы



➤ **Инициатор создания универсального формата пакетов Колин Ватсон.**

и интегрированы в SDK Ubuntu. Уже подготовлен и готовится к публикации прототип нового инструментария на Python, использующий JSON как формат для описания метаданных. Контейнер для поставки приложения — максимально упрощенный вариант пакетов Deb, без скриптов и расширенных метаданных. Затем утилиту для управления новым форматом перепишут на языке C, повысив производительность.

Особенность нового формата — отсутствие необходимости учета зависимостей между приложениями: упакованная программа будет поставляться в известном базовом окружении Ubuntu и после установки размещаться в отдельной директории внутри домашней директории пользователя. Следствие подобного подхода — высокая скорость установки (не надо обновлять индексы, проверять зависимости и т.п.) и установка программ под идентификатором пользователя, без необходимости получения привилегий *root* и без пересечений с системой и приложениями других пользователей. Более того, планируется предоставить средства для запуска приложений внутри изолированных контейнеров.

Canonical намерена до конца мая довести состояние сборки Ubuntu для смартфонов до рабочего. Сборка позволит выполнять все типичные для телефонов действия, включая автоматизированное обновление прошивки, поддержку выхода в Сеть через 3G и Wi-Fi, применение датчиков положения в пространстве для портретного и ландшафтного режимов.

ОС ДЛЯ КОСМОСА

Linux выходит на орбиту

NASA переводит используемые на МКС ноутбуки на Linux.

Космическое агентство NASA приступило к переводу ноутбуков, используемых на Международной космической станции, с платформы Windows на Linux. Намеченные для миграции ноутбуки входят в сеть OpsLAN, применяемую для организации типовой ежедневной работы космонавтов, от обмена сообщениями и выполнения задач по инвентаризации до захвата изображений и видео с имеющихся на станции камер.

В процессе выбора дистрибутива инженеры NASA остановились на использовании уже проверенной временем ветки Debian 6. Переход на Debian 7 не был утвержден из-за возможных проблем с надежностью в связи с недостаточно полным тестированием свежей версии. Также отмечается, что кроме планов по переводу рабочих станций на Linux, на МКС уже

давно используются бортовые компьютеры, оснащенные Scientific Linux.

Причиной миграции с Windows на Linux называют необходимость в стабильной и надежной ОС, начинку которой можно полностью контролировать собственными силами. В качестве одного из важных критериев называется возможность при необходимости самостоятельно внести исправления или изменения в любые части ОС,



➤ **«Робонавт» — так назвали создатели R2 своего нового «сотрудника».**

адаптировать систему для специфичных применений и выполнить тонкую настройку параметров работы. Для ускорения миграции на Linux агентство NASA заключило контракт на проведение обучения с организацией Linux Foundation, которая разработала специальный сводный курс, охватывающий сразу несколько используемых в NASA дистрибутивов (Debian и RHEL).

Одновременно сообщается о начале испытаний R2 — первого человекоподобного робота на базе Linux для проведения работ в открытом космосе. Робот уже доставлен на МКС и задействован для проведения экспериментов. В дальнейшем он будет использоваться для выполнения задач, слишком опасных или утомительных для космонавтов.

Подробности о новом чудо-роботе см. на сайте <http://robonaut.jsc.nasa.gov/>.

softline®



Services Software Cloud

ИТ-архитектура вашего бизнеса



СПО И ОБЩЕСТВО

За электронную демократию

Продукты СПО — логичный выбор инструмента для ее построения.

14 и 15 мая 2013 года в Москве прошел Второй федеральный конгресс по электронной демократии. Руководители органов государственной власти, общественные деятели, российские и зарубежные эксперты обсуждали вопросы государственной политики в сфере обеспечения открытости деятельности органов власти и местного самоуправления.

25 министерств и федеральных структур России уже используют решения на основе ПО с открытым кодом, 21 регион принимает и обрабатывает федеральные сведения в среде СПО. На базе СПО было реализовано web-наблюдение во время выборов Президента РФ, функционирует Единый портал электронных госуслуг и портал Российской общественной инициативы.

СПО в государственных структурах активно используется во всем мире: США,

Франция, Великобритания, Италия, Исландия, Бразилия высоко оценили удобство и экономичность открытого кода. Из 178 членов ООН 84 страны используют на серверах свободные операционные системы. Россия планирует вступить в международное партнерство «Открытое правительство [Open Government Partnership]».

По мнению экспертов «ПингВин Софт-вер», озвученном на конгрессе, для развития электронной демократии государству необходимо обеспечить свободу действий российским производителям СПО. Это стимулирует рост отечественного ИКТ-рынка и создаст солидную технологическую основу. Открытые технологии позволяют свободно применять, тиражировать и модифицировать системы для реализации электронных услуг, обеспечивают цифровой суверенитет России и независимость от иностранных вендоров, а также существенную эконо-

мическую выгоду, особенно ощутимую при масштабных внедрениях.

Распоряжение Правительства РФ №2299-р «О плане перехода федеральных органов исполнительной власти и федеральных бюджетных учреждений на использование свободного программного обеспечения 2011–2015 гг.» вселяет уверенность, что в России открытый код продолжит набирать популярность и станет приоритетом не только для государственных структур, но и для большинства коммерческих организаций.

Презентации от «ПингВин Софт-вер»:
 » Д. Фишелев «Открытые технологии для развития информационного общества» <http://www.slideshare.net/pingwinsoftware/open-technologies-forinformationsocietypresentation15052013>
 » Е. Василенко «Электронное правительство. Социальный аспект» <http://www.slideshare.net/pingwinsoftware/social-effect-fromegovernment>

ИНТЕРНЕТ — НАШЕ ВСЕ

СПИК-2013

Состоялось основное мероприятие Северо-Запада по бизнесу в Интернете.

Санкт-Петербургская интернет-конференция, организованная Российской ассоциацией электронных коммуникаций и компанией TRINET, прошла 20 и 21 мая в конгресс-центре гостиницы «Прибалтийская Park Inn». На открывающей сессии были оглашены трендообразующие факты о Рунете:

» Ежемесячная аудитория Интернета в России (2012) составляет 55 % населения страны (64,4 млн). Наибольший вклад вносит сегмент 25–44 лет. 89 % подростков пользуются Интернетом ежедневно, а в возрастной группе 18–24 — 96,6 %.

» 18,4 млн пользуются мобильным интернетом (1,2 млн — только им), и за 2012 год это количество увеличилось на треть. Более 80 % пользователей выходят в Интернет из дома с мобильных устройств. 25,1 % просмотров сайтов в Рунете — с мобильных устройств, это 3-е место в Европе.

» В Госдуме РФ за 2012 год внесен 31 законопроект по интернет-сфере. 49 % из них отрицательно оцениваются экспертами отрасли, 23 % — положительно.

» За 2012 год стоимость широкополосного доступа снизилась в 4 раза, доля ШПД среди пользователей составляет 73 %.

» В 2012 году продано порядка 3,3 млн планшетов, рост продаж — 444 %.

» В 2012–2013 году ожидается рост экономики Рунета на 30 %. Рынок интернет-рекламы составил 56,3 млрд руб. (2012 год).

» \$792 млн — объем венчурных инвестиций в 2012 году (вдвое выше прошлогодних показателей), это 4-е место в Европе.

» Всего просмотров видео в Рунете — 10,3 млрд, около 55,6 млн человек.

Ключевые онлайн-активности — соцсети, поиск, почта, видео, новости (в порядке уменьшения среднесуточной аудитории). Удвоилось число пользователей smartTV, составив около 4 % населения России. У большинства пользователей несколько точек доступа в Сеть. По проникновению лидирует Москва, но в Петербурге этот показатель также растет (77 % и 73 %).

Около 200 докладов российских, немецких, финских и итальянских специалистов ИТ-отрасли практически офици-

» СПИК-2013 собрал около 2500 участников.



ально придали СПИК международный статус. Участники обсудили кибербезопасность, э-коммерцию, гражданские проекты, онлайн-видео, юзабилити, перспективы новых СМИ, монетизацию проектов, развитие стартапов и интернет-законодательства и многое другое.

Параллельно прошла выставка 26 интернет-компаний, представивших актуальные решения и разработки рынка. Стендисты предоставляли открытые консультации и мини-мастер-классы, а также проводили развлекательные мероприятия.

Фотоотчет о конференции можно посмотреть на <http://2013.sp-ic.ru/photo/>.

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

НАЧАЛО ПОЛОЖЕНО

Огненные телефоны

Хакеры наконец дождались продаж.

23 апреля этого года стартовала поставка двух первых моделей телефонов — недорогой Keon (€91) и «продвинутой» Peak (€149), на базе Firefox OS, подготовленных проектом Geeksphone совместно с компаний Telefonica и сообществом Mozilla. Пока это ознакомительные модели [Developer Preview], только для разработчиков приложений для Firefox OS — чтобы те оценили работу платформы и приложений на реальном оборудовании, а не в симуляторе (например, акселерометр и камеру в нем проверить трудно):

Текущие производственные мощности позволяют выпускать около 5000 аппаратов в день; заявки от разработчиков будут удовлетворяться в порядке очереди. Заказы принимаются из любых стран. Начало продаж первых потребительских устройств на основе Firefox OS ожидается в июне. Телефоны не содержат средств привязки к вендору и поддерживают автоматическую установку обновлений по сети. На устройства предустановлена последняя тестовая сборка платформы Firefox OS, которая интенсивно развивается и еще не содержит все запланированные функции, которые будут доступны в первых выпусках для конечных потребителей. Обновление прошивки осуществляется автоматически при появлении новой сборки на серверах Mozilla.

Платформа Firefox OS ориентирована прежде всего на мобильные устройства и предоставляет расширенный Web API для создания специализированных мобильных web-приложений, в полной мере использующих все возможности современных телефонов. За основу принято ядро Linux и низкоуровневые компоненты из платформы Android. Вместо виртуальной машины Dalvik для запуска приложений задействован web-стек Mozilla.

Познакомиться с возможностями платформ Firefox OS без покупки нового устройства можно, установив симулятор, оформленный в виде дополнения для обычного браузера Firefox. Протестировать приложения для Firefox OS можно на телефоне с платформой Android, установив на нее мобильную версию Firefox и приложение Marketplace for Android.

Приложения распространяются через каталог-магазин Mozilla Marketplace, который поддерживает распространение как бесплатных, так и платных при-

ложений. Для организации идентификации пользователей задействован сервис Mozilla Persona, основанный на технологии BrowserID.

Приложения для Firefox OS формируются с использованием стека HTML5 и расширенного программного интерфейса Web API, который позволяет организовать доступ приложений к аппаратному обеспечению, телефонии, адресной книге и дру-

«Платформа Firefox OS ориентирована на мобильные устройства.»

гим системным функциям. Вместо предоставления доступа к реальной файловой системе, программы ограничены внутри виртуальной ФС, построенной с использованием IndexedDB API и изолированной от основной системы. В конечном счете, планируется сформировать набор стандартов для создания универсальных мобильных web-приложений, которые могут обеспечить функциональность, свойственную обособленным мобильным стекам — как правило, контролируемым отдельными производителями (Android, iOS, Apple iOS и Windows Phone).

В начале мая проект Mozilla представил инициативу «Телефоны для разработчиков программ для Firefox OS», в рамках которой любой разработчик интересной программы для платформы Firefox OS может бесплатно получить для тестирования своего приложения телефон Geeksphone Keon. В заявке могут фигурировать и идеи планируемых приложений, и намерения портировать программы с других платформ, использующих web-технологии, таких как Chrome, webOS, BlackBerry WebWorks или PhoneGap. Число телефонов ограничено. Заявки принимались до конца мая. **LXF**

➤ **Дешевые и стильные оранжевые телефоны скоро появятся и в России.**



Новости короткой строкой

➤ 20–22 сентября 2013 г. в г. Калуге состоится юбилейная X Конференция разработчиков свободных программ. Ее тематика — научные проекты в различных областях знаний, особенности свободного лицензирования, СПО для органов госвласти. Приглашаются разработчики из России и других стран. Источник: <http://www.altlinux.ru/news/item/695/>

➤ В Конгресс США подан новый законопроект “Unlocking Technology Act of 2013”, легализующий право на разблокировку своего мобильного телефона. Источник: <http://mashable.com>

➤ Суд признал недействительным контракт на приобретение муниципалитетом г. Алмада (Португалия) программных продуктов от Microsoft на сумму €550 тыс. Источник: <http://fsfe.org>

➤ Apache OpenOffice 3.4 преодолел отметку в 50 миллионов загрузок с момента выпуска (учтены загрузки 3.4.0 и загрузки корректирующего выпуска 3.4.1). Источник: <https://blogs.apache.org>

➤ Эрик Шмидт [Eric Schmidt], председатель Совета директоров компании Google, объявил на конференции Dive Into Mobile 2013 о взятии нового рубежа — 1,5 млн активаций новых Android-устройств в день. Планка в 1 млн в день была преодолена в июне прошлого года. Источник: <http://androidcommunity.com>

➤ Администрация Эстремадуры, автономной области Испании, начала процесс перевода с Windows на Linux и свободное ПО около 40 тыс. компьютеров в госучреждениях. Источник: <https://joinup.ec.europa.eu>

➤ Подготовлена сборка дистрибутива openSUSE 12.3 для телефонов и планшетов на Android. Источник: <http://lizards.opensuse.org>

➤ 5 мая, после более чем двух лет разработки, увидел свет релиз Debian GNU/Linux 7.0 (Wheezy). Источник: www.debian.org

➤ Согласно опросу компании SafeNet, Inc. и Ассоциации информационной и программной индустрии (SIIA), в США две трети издателей ПО намерены уже к концу 2013 года использовать облачную модель доставки. Источник: www.siiainet



Новинки программного и аппаратного обеспечения в описании наших экспертов



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

Монотеизм или дуализм? А может...

В прошлой колонке (LXF170) был дан эскиз апокалиптической картины — засилья одного-единственного истинно правильного Linux'a, под гребенку которого постригут все дистрибутивы.

Такой сюжет ныне вполне реален. Но, к счастью, пока не реализован. Ибо существует альтернатива — Ubuntu, недавно отпраздновавший выход своей очередной версии. О нем самом по себе написано столько, что повторяться неинтересно. Отметим только несколько моментов, важных в данном контексте.

Первое — это дистрибутив с самой развитой инфраструктурой. Второе — число «настольных» его пользователей превышает количество всех остальных «desktopных линуксоидов», вместе взятых. Третье — это дистрибутив, породивший неисчислимое множество ремиксов, респинов и прочих клонов. И, наконец, четвертое и главное — разработчики Ubuntu не собираются отказываться от его своеобразия в угоду новомодным тенденциям.

Однако не Ubuntu единым живое многообразие Linux'ов. Не забудем и о прародителе-Debian, который в силу своей «многоядерной» ориентации не может замкнуться на унифицированном Linux — и дал необозримый океан производных дистрибутивов. В том числе таких любопытных, как Aptosid и Siduction, основанных на «нестабильной» ветке. Очередная версия Aptosid только что вышла. По опыту прежних лет, днями можно ожидать и появления свежего Siduction. И тогда им будет посвящена следующая колонка.

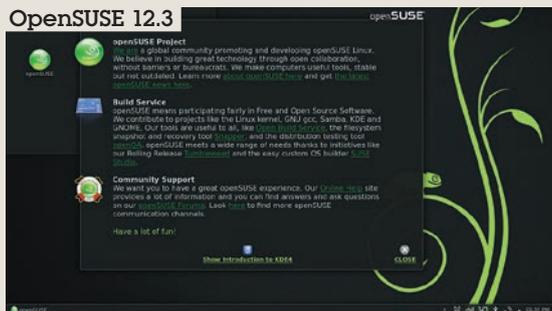
alv@posix.ru

Сегодня мы рассматриваем:

OpenSUSE 12.3 14

Шесть месяцев после 12.2 пролетели быстро, и зеленый хамелеон возвращается с лучшим из дистрибутивов — по крайней мере, для KDE-поклонников. Пользователи Gnome будут менее довольны, но в любом случае ясно: команде Fedora придется попотеть, прежде чем они догонят текущего чемпиона среди RPM-систем.

OpenSUSE 12.3



» Судьбоносный хамелеон... красно-золотой и зеленый. Но зеленый, безусловно, перевешивает.

Crusader Kings II 16

Пускай норманны сравняли с землей северную Англию, сделали язык администрации иностранным и отказали женщинам в правах на 900 лет. Зато без них у нас не было бы чудесных замков Уэльса — и удобной стартовой площадки для этой игры!

Crusader Kings II



» Где же отважный сэр Робин, когда он вам позарез нужен? Как где — крушит крестьян в долине Рейна.

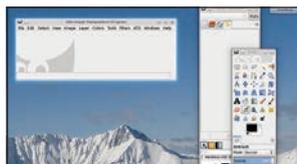
TeamViewer 8 17

Мы рекомендуем *TeamViewer*. А почему бы и нет? Найдите на стр. 17 всю подноготную этого решения удаленного рабочего стола, который способен даже создать для вас более счастливые и здоровые отношения с удаленными родственниками.

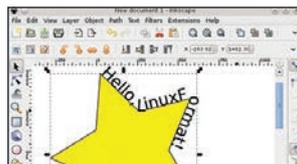
Сравнение: Редакторы изображений

с. 20

GIMP



Inkscape



Krita



MyPaint



Pinta



Мы знаем: картинка стоит тысячи слов, а стало быть, и редакторы изображений стоят своего веса в электронных. Пиксельный Ренессанс начинается!



OpenSUSE 12.3

Разочарованный последними версиями основных дистрибутивов, Маяк Шарма приступает к данной версии с осторожностью.

Вкратце

» Основной дистрибутив Linux, поддерживающий несколько рабочих столов. Альтернативные RPM-дистрибутивы — Fedora и Mageia.

Кто-то не придает этому большого значения, но постоянная неразбериха со сроками выхода версий не радует фактических пользователей. Поэтому мы немного настороженно отнеслись к этой запоздалой версии, появившейся через каких-то шесть коротких месяцев с момента выхода последней, также припоздавшей версии. Тем не менее, рады сообщить, что, несмотря на сжатые сроки выпуска, OpenSUSE 12.3 — довольно целостная версия, в которой можно не опасаться неприятных сюрпризов.

Движок этой версии работает на базе ядра Linux 3.7. Основная изюминка этого ядра — поддержка обновленных драйверов для всех типов «железа», особенно новейших карт Nvidia и Intel.

Также в дистрибутиве OpenSUSE 12.3 полностью завершен переход на менеджер системных сервисов *systemd*. Это преобразование началось еще в версии 12.1

«Рабочий стол KDE выглядит цельным, и ощущается интеграция.»

при участии разработчиков проекта Fedora. Также усовершенствован стек виртуализации, загрузки OpenSUSE в виртуальной среде и поддержка других операционных систем.

Версия доступна как устанавливаемые Live-образы и образы DVD только для



» Особая похвала команде дизайнеров OpenSUSE за такую неожиданно темную и более органичную заставку. Мы находим ее довольно милой.

установки, и для 32-, и для 64-битных машин. Кроме них, имеются также образы для архитектуры ARMv7. Live-образ для KDE и Gnome весит порядка 1 Гб, его придется записывать на DVD или USB-брелок. Разработчики употребили дополнительное место, чтобы втиснуть сюда больше программ, чем в предыдущих версиях. Однако если вы хотите установить OpenSUSE на машине, не имеющей стабильного интернет-подключения, лучше воспользоваться образом DVD.

В установщике дистрибутива нет каких-либо примечательных изменений. Установщик поддерживает UEFI на 64-битных машинах, а также обладает экспериментальной поддержкой для Secure Boot. При установке дистрибутива на машину

с поддержкой Secure Boot ознакомьтесь с примечаниями к версии.

Разработчики OpenSUSE не отдают приоритета какому-то одному рабочему столу, их рабочий стол KDE выглядит очень цельным, и ощущается его тесная интеграция. Здесь эlegantная темно-зеленая заставка и приятные эффекты при уменьшении и увеличении окна. При установке дистрибутива с DVD рабочий стол KDE устанавливается по умолчанию.

Проворный работник

OpenSUSE 12.3 поставляется с последней версией KDE 4.10. Пользователи Internet единодушно называют эту версию «быстрейшей из всех KDE», что OpenSUSE только на руку. Одним из важнейших факторов, приведших к улучшению характеристик KDE 4.10, является переработанная платформа Nepotik, являющаяся собой новый индексатор, который не сжигает все доступные ресурсы.

Улучшен также файловый менеджер Dolphin — он научился группировать файлы по метаданным. Венцом является инструмент Nepotik Cleaner, который вы могли видеть в версии 4.9. С помощью этого инструмента можно удалять старые и неверные метаданные, к тому же он приятно ускоряет некоторые простые вещи, например, поиск файлов.

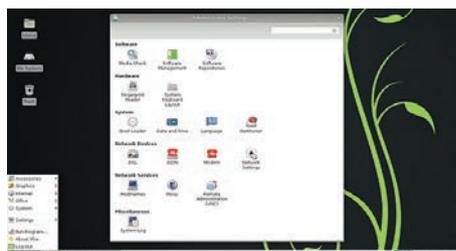
Для поддержки согласованного рабочего процесса между Gnome и KDE OpenSUSE 12.3 заменил некоторые привычные для рабочих столов приложения: так, приложение KDE Calligra Suite заменено на LibreOffice (хотя и на более старую версию 3.6).

Свойства навскидку



Более увесистые Live-образы

Инсталлируемые образы и для KDE, и для Gnome больше не помещаются на CD. Благодаря дополнительному пространству DVD, разработчикам удалось разместить больше приложений, однако все еще нет приложения GIMP.



Спасательный диск

В придачу к обычным инсталлируемым версиям в OpenSUSE 12.3 есть 570-МБ неинсталлируемый образ, предназначенный для спасения и восстановления системы. Также есть ряд соответствующих модулей YaST.

OpenSUSE против Fedora

С момента выхода двух последних версий разработчики этих основных RPM-дистрибутивов прилагают много усилий, чтобы не ограничиваться рамками одного рабочего стола. Благодаря их трудам стало неверным рекомендовать Fedora только из-за рабочего стола Gnome, а OpenSUSE — только из-за рабочего стола KDE. Поэтому приходится их сравнивать.

Fedora 18 не очень впечатляет из-за изменений в установщике *Anaconda*. Расстроил также менеджер пакетов *PackageKit*.

Зато разработчики OpenSUSE 12.3 улучшили свою систему менеджеров пакетов. Их системой инсталляции One-Click приятно пользоваться. А еще в OpenSUSE 12.3 включена последняя вер-

сия KDE. Рабочий стол Gnome 3.6 ничуть не хуже, чем в Fedora 18, но ему нечем переманить к себе существующих пользователей Fedora. Тем не менее, даже пользователям Gnome мы рекомендовали бы осваивать OpenSUSE 12.3 — по крайней мере, до тех пор, пока Fedora не приведет в порядок свой установщик *Anaconda*.

Уточним, браузером по умолчанию выбран популярный *Firefox*.

В отличие от Fedora, OpenSUSE воспользовалась задержкой выпуска версии для настройки работы приложений, включая *LibreOffice*, и они гармонично сочетаются со всем, что есть на рабочем столе. Это совсем не обязательное, но приятное внимание к деталям существенно добавляет очков OpenSUSE при ее позиционировании в качестве настольной системы для корпоративного использования.

Другой хорошо работающей в OpenSUSE 12.3 новинкой KDE стало разворачивание меню приложений в кнопке заголовка окна, либо сворачивание в глобальном меню на самом верху экрана. Управлять этой опцией можно из-под вкладки тонкой настройки модуля представления приложений системных настроек инструмента конфигурации KDE.

Изменения в Gnome носят менее революционный характер. Это Gnome 3.6, который есть также в Fedora 18, вышедшей за пару месяцев до OpenSUSE 12.3. На самом деле, к моменту прочтения вами данной статьи, в последней стабильной версии они уже разместят Gnome 3.8.

Единственная заметная черта Gnome 3.6 — пользовательские настройки ПО, например, новое меню в *Nautilus*, отображающее последние открытые документы. Еще есть новое приложение для управления виртуальными машинами *Voxes*. Но в нем отсутствуют опции настройки, по-

» **Увы, ничего выдающегося в Gnome 3.6 на OpenSUSE 12.3 не обнаружено.**

этому в отличие от других решений по виртуализации, включая *VirtualBox*, нельзя сконфигурировать виртуальную сеть.

Ай да YaSTe

Gnome 3.6 было посвящено довольно много места в обзоре Fedora 18, тогда как в OpenSUSE 12.3 в этом смысле нет ничего особенного. Поэтому мы вдоволь поразвлеклись с KDE 4.10.

В этой версии есть все выдающиеся приложения KDE. Здесь есть *Amarok 2.7*, который щеголяет прекрасными новыми опциями, например, плагином *Perotnik*, и самый последний *DigiKam*, 3.0, с новыми опциями. К изумлению, в версии KDE в OpenSUSE отсутствует просмотрщик PDF. Также почти 1-ГБ образ не включил *GIMP*. Пришлось нам самим выудить их из онлайн-репозитория.

Здесь следует упомянуть, что OpenSUSE не особенно изящно обрабатывает обновления после установки. Если вы решите установить сначала пакет, а потом уж обновления, эти обновления притянутся автоматически при установке приложения. Тем, кто не знаком с обработкой приложений в OpenSUSE, придется поскрести в затылке, недоумевая, почему установка *LibreOffice 4* заодно вовлекает *GIMP* или *telnet*.

В остальном установка пакетов на OpenSUSE приносит сплошное удовольствие, а система установки одним щелчком определенно впечатляет. Кроме того,

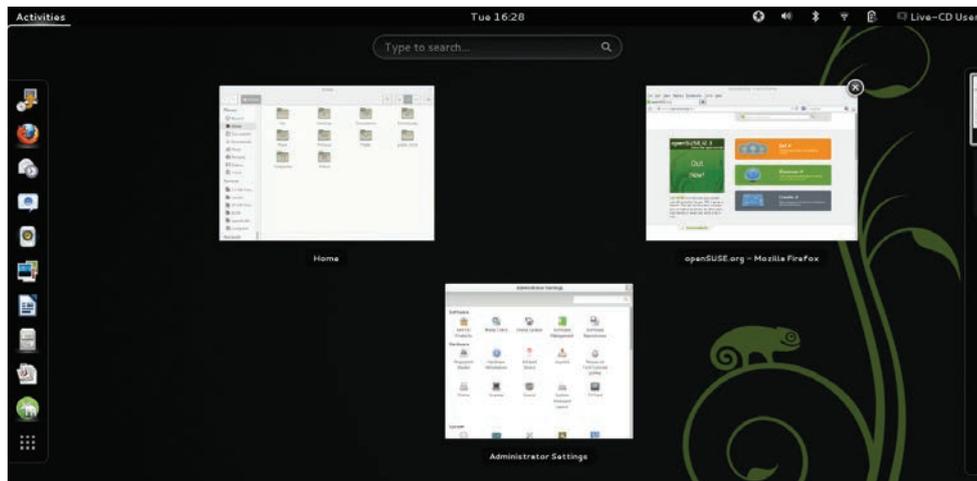
при установке приложений можно пользоваться *Apper* или встроенным компонентом управления пакетами *YaST*, но они не сравнятся с *Software Center* в Ubuntu или *Mint*.

YaST — это инструмент настройки OpenSUSE, который преобразует множественные задачи настройки в инструментарий, не зависящий от рабочего стола. Аналогичные инструменты можно встретить и в других дистрибутивах — например, в *Magia*; но в OpenSUSE это решение более комплексное.

И хотя *YaST* отлично справляется с конфигурацией большинства аспектов системы, его можно бы чуть-чуть доработать. Здесь могли бы быть новые компоненты для настройки расширенных возможностей, например, виртуализации, а также для общих задач предприятия — управление через удаленные рабочие столы.

В целом на работу KDE 4.10 на OpenSUSE 12.3 жаловаться нечего. Все работает, как обещано. Однако мы бы не стали рекомендовать его новичкам в Linux, из-за его малоизящных сообщений о неисправности: вывод имени и идентификатора процесса, который предотвращает текущее действие — это не то, что полезно неопытным пользователем.

Все-таки OpenSUSE 12.3 заслуживает внимания и особенно актуально для пользователей Fedora, подыскивающих RPM-дистрибутив, который можно было бы реформировать по своему вкусу. **LXF**



LINUX **Вердикт**
FORMAT

OpenSUSE 12.3

Разработчик: OpenSUSE Project
Сайт: www.opensuse.org
Лицензия: GPL

Функциональность	8/10
Производительность	8/10
Простота использования	7/10
Документация	8/10

» Очевидно, что это лучший RPM-дистрибутив. И точка.

Рейтинг 8/10

Crusader Kings II

Алекс Кокс побыл графом Эссексом и королем Франции, и развалил Священную Римскую империю. Вот и доверяй ему после этого ответственную работу...

Вкратце

» Стратегия о семейных интригах и Европейском завоевании. См. также: *Freeciv*, *Battle for Wesnoth*.

В моей семье проблемы. Мне 40, и смерть может застичь меня в любой момент, так что я должен найти подходящего преемника. В руках моего старшего сына мой род будет в опасности — этого обжору и шута ненавидит почти все королевство. Даже моя нынешняя жена плетет заговоры с целью его отречения. Моя более чем способная дочь сосватана важному магнату на другом краю континента, и сможет унаследовать мое королевство только через сложнейшие политические маневры — убийство тут, заключение там — и с одобрения моих неумытых вассалов, которые отнюдь не приняли на ура мою прошлую поправку к законам. Да еще меня раздражает, что Франция на нас смотрит косо. Вот они, трудовые будни короля... но эту жизнь я не променяю ни на что на свете.

Итак, *Crusader Kings II*. Окажитесь в Европе между 1066 и 1453 годами, выберите себе любого властителя в этом периоде — от всемогущего императора до захолустного графа — и проживите его жизнь и династию. Понадобится давать потомство, учить его и оставлять наследие, достойное вашего доброго имени: ведь все идет к тому, что вам придется играть за преемника каждого персонажа до его неминуемой смерти. Помня об этом, сплотите вокруг себя счастливых и толковых людей, чтобы справляться со всякими мелочами в вашем государстве: подданные-ненавистники могут счесть, что вам более к лицу нож в спине. Также следует установить добрые отношения с соседями — заодно обдумывая притязания на их территорию, дабы расширить свои границы.

Вас, разумеется, ждет множество битв, хотя их элементы крайне просты: соберите

» Битвы и интриги выдающихся, ярких гигантов, подавляющих рост городов, за которые они сами же и сражаются.



» Политическая карта четко покажет, как люто соседи вас ненавидят.

армию, натравите ее на другую армию, немного понаблюдайте. Нечто вроде упрощенной *Civilisation*. Сражения выступают скорее в роли фона для политических интриг, ибо даже в качестве короля при любой серьезной заварухе вам придется вербовать войска у ваших вассалов. Вассалов это злит и даже может сподвигнуть их на бунт или неподчинение вашим непогрешимым приказам. То же вам светит и при чрезмерном разрастании ваших владений, а значит, вассалы — и необходимость хороших с ними отношений — являются неизбежным злом. Одно слово, феодализм.

Нужно сбалансировать свой золотой запас, репутацию (которая за это придаст вам весу в политических переговорах) и набожность, помогающую как-то терпеть вездесущую и довольно назойливую церковь. По умолчанию вы играете только за христианские нации, но по ходу придется иметь дело с фракциями и народами других конфессий, от язычников до мусульман и всех прочих. И остерегайтесь епископов, возглашающих радикальные речи. Одержимые религией плохо относятся ко всем проявлениям ереси.

Подобно многим стратегиям, *Crusader Kings II* в основном сводится к генератору случайных чисел, прикрепленному к большой электронной таблице, поверх которой помещена карта; но пусть вас это не смущает. Игра не менее увлекательна, чем любая другая в этом жанре, в которую

нам случалось играть, и ваш образ действий в ней целиком зависит от того, чего вы хотите добиться. Единственно верная цель — продолжение своего рода; если судить и сравнивать вас с успешными реальными династиями на момент вашей неизбежной кончины, итог в лучшем случае выразится как "Game Over".

Да, интерфейс неуклюж и труден для освоения, и хотя имеется справочник, придется немало прочесть онлайн, чтобы добиться успеха. Да, это не назовешь игрой для всех, даже с натяжкой. Не-фанатам стратегий *Crusader Kings II* может показаться слишком сложной, чтобы в нее втянуться. Но в Paradox создали глубокую и крепкую игру, даже с учетом ее цены. **LXF**

LINUX FORMAT Вердикт

Crusader Kings II

Разработчик: Paradox Interactive
Сайт: www.paradoxplaza.com
Цена: £34,99

Игровой процесс	9/10
Графика	8/10
Долговечность	10/10
Оправданность цены	7/10

» Увлекательная смесь семейных интриг и долгосрочной стратегии с погружением в былые эпохи.

Рейтинг **8/10**

TeamViewer 8

Вправду ли *TeamViewer 8* — лучший инструмент удаленного рабочего стола для предприятия и домашних пользователей? **Шашанк Шарма** это подтвердит.

Вкратце

» Проприетарный межплатформенный инструмент удаленного рабочего стола со множеством полезных функций; идеален в равной степени для новичков и продвинутых пользователей. См. также: *TightVNC*.

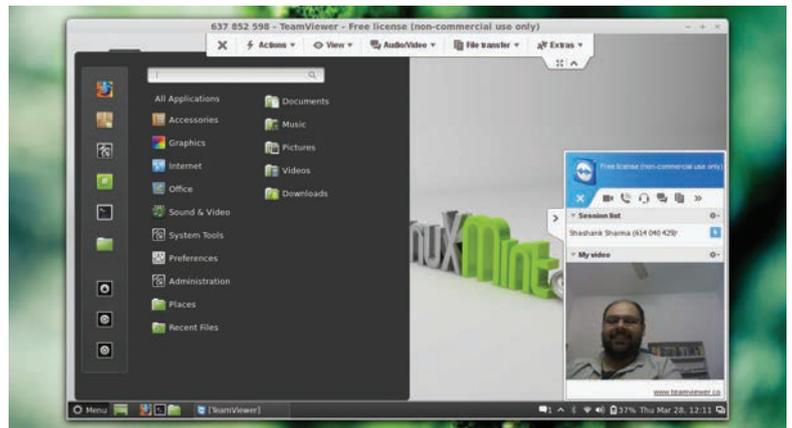
Спросите пользователей Linux об инструментах удаленного рабочего стола, и в качестве практически универсального ответа вы получите *VNC*. Но если для ваших потребностей *VNC* маловато, *TeamViewer* — одна из богатых функциями альтернатив. Он доступен для Windows, OS X и Linux.

Лучшее в нем — простота установки. Вместо разных пакетов RPM и Deb для разных дистрибутивов, единственный Deb-пакет имеется для Debian, Ubuntu и Mint. Пакет .rpm подобным же образом работает для дистрибутивов на базе RPM, типа Fedora, OpenSUSE и CentOS. К сожалению, последний релиз недоступен в виде исходного пакета для других дистрибутивов.

TeamViewer идет в связке с *Wine*, устанавливаемым автоматически. Хотя это не родное приложение Linux, по сравнению с другими программами, которые требуют наличия *Wine*, при работе с *TeamViewer* нет проблем с производительностью. Домашние пользователи могут получить полную версию бесплатно, но коммерческие пользователи должны выбрать один из трех вариантов лицензий, по цене от £343.

Удаленный рабочий стол

Для инструмента, позволяющего управлять удаленными рабочими столами, записывать сессии, делиться файлами и проводить встречи, интерфейс весьма прост. При запуске *TeamViewer* в главном интерфейсе будет две вкладки, Remote Control и Meeting. Надо сообщить свой идентификатор пользователя (ID) и пароль, указанные в Allow Remote Control, тем, кому вы даете доступ к своей машине. Анало-



» Пусть вас не одурячит список функций на сайте. Интерфейс может быть одним и тем же на разных платформах, но набор функций отличается.

гично вы должны ввести ID, предоставленный справа на экране, и затем пароль для доступа к удаленной машине.

Есть еще второе окно, скрытое; доступ к нему дает щелчок по кнопке Computers & Contacts внизу справа. Там перечислены все машины, с которыми можно соединиться при должной настройке. Традиционно требовалось, чтобы кто-то физически вошел на удаленную машину, и только тогда к ней был возможен доступ; в *TeamViewer 8*, настроив персональный пароль на удаленной машине, можно получать к ней доступ, даже если никто не зашел на нее. Эта функция, именуемая «несопровожаемый доступ», идеальна, если надо влезть на свой домашний ПК с работы.

Другая важная функция, запись сессии, серьезно улучшена. Последний релиз позволяет записывать онлайн-встречи, а также сессии удаленного управления, со звуком и видео. Это может стать солидным подспорьем в создании документации и обучающих руководств. Во время удаленной сессии рабочего стола щелкните по Extras в панели инструментов наверху, и выберите Record > Video. Чтобы воспроизвести записанное видео, перейдите в главный интерфейс *TeamViewer* и щелкните Extras > Play или конвертируйте записанную сессию. Поддержки конвертирования видео в какой-либо из популярных форматов дистрибутивов Linux пока нет.

В последнем релизе появилась новая функция — консоль управления на базе браузера. Создав себе бесплатную учет-

ную запись *TeamViewer*, можно управлять всеми своими контактами и машинами через интерфейс браузера, и даже начать сессию удаленного управления одним щелчком внутри интерфейса браузера.

Консоль управления

Бесплатным пользователям консоль предлагает не слишком много, но коммерческие получают больше функций, таких, как определение разрешений для пользователей и групп и создание подробных журналов для ваших сессий.

Среди других новых функций — поддержка web-камеры для трансляции видео во время встреч. *TeamViewer* невероятно прост в установке, работает на нескольких платформах и даже на устройствах Android. В *VNC* ничего этого нет. **LXF**

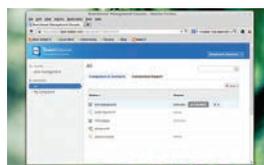


Свойства навскидку



Запись сессий

Легко записать звук и видео во время удаленных сессий рабочего стола или встреч онлайн.



Консоль управления

Консоль управления позволяет запускать и завершать удаленные сессии из браузера.

LINUX FORMAT Вердикт

TeamViewer 8

Разработчик: TeamViewer GmbH

Сайт: www.teamviewer.com

Цена: От £343 (для некоммерческих пользователей бесплатно)

Функциональность 9/10

Производительность 9/10

Удобство использования 9/10

Оправданность цены 7/10

» Коммерческая лицензия неприлично дорога; зато невероятно простая установка и впечатляющий набор функций.

Рейтинг **9/10**



НОВАЯ АВТОПЛАТФОРМА

Машина для Android

Представлен прототип автомобильной информационно-развлекательной системы, развиваемой Linux Foundation.

На прошедшей в середине мая в Сан-Франциско конференции Linux Collaboration Summit представители компании Jaguar Land Rover объявили о намерении вывести в 2014 году на рынок новую информационно-развлекательную автомобильную систему на базе Linux и открытой автомобильной платформы, развиваемой альянсом GENIVI и группой Automotive Grade Linux (AGL). Суть платформы — предоставление готовой ОС и программных интерфейсов, с которыми автопроизводители будут думать не о низкоуровневой инфраструктуре, а только о конечных приложениях.

Там же был представлен первый демонстрационный прототип новой открытой автомобильной ОС на базе Linux — AGL Demonstrator, и сопутствующего пакета для разработки приложений. Платформа поддерживает методы голосового управ-

» 2014 год
станет годом
Linux и Android
в автомобилях.



ления и взаимодействие с CAN-сетью для доступа к датчикам и передаче данных между узлами автомобиля. Для разработки приложений предлагается фреймворк на базе технологий HTML5. Также предлагается базовый интерфейс пользователя, который производители смогут переработать на свой вкус. За основу представленной системы взяты наработки мобильной платформы Tizen и Android.

Опрос клиентов Jaguar Land Rover показал, что большинству пользователей

в автомобиле достаточно функций обычной домашней мультимедиа-системы, плюс навигация и отслеживание информации о трафике. Тем, кому базовых функций мало, новая платформа предоставит возможность установки расширяющих функциональность приложений, написанных с применением технологий HTML5.

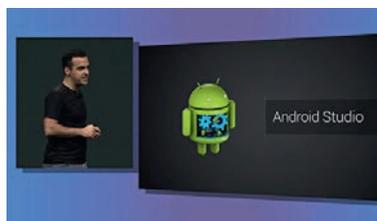
Для стимулирования разработки таких приложений и для привлечения энтузиастов, компания Jaguar Land Rover и Linux Foundation объявили конкурс “2013 AGL User Experience Contest”, предложив до 17 мая оформить концепт интерфейса информационно-развлекательной системы. Победители получат интернет-планшеты и шанс подключиться к работе с AGL и Jaguar Land Rover. Номинаций три: наиболее удобный интерфейс, лучшее визуальное оформление и лучшая новая или добавочная функциональность.

В ПОМОЩЬ РАЗРАБОТЧИКАМ

Унификация создания приложений

Компания Google представила среду разработки Android Studio.

Компания Google представила на майской конференции Google I/O новую интегрированную среду разработки приложений для платформы Android — Android Studio, на базе исходных текстов системы IntelliJ IDEA, развиваемой компанией JetBrains. Это открытый проект, и код IDE уже доступен для загрузки через репозиторий исходников Android, под лицензией Apache 2.0.



» Самое ценное в Android Studio — это удобный универсальный интерфейс разработки приложений.

Новая среда оптимизирует выполнение типовых задач разработки и включает средства упрощения тестирования программ на совместимость с разными версиями платформы и инструменты для проектирования приложений для устройств с различными разрешениями экрана (планшеты, смартфоны и ноутбуки). Google планирует применять данное приложение как первичную среду разработки взамен плагина ADT для Eclipse.

Сверх функций IntelliJ IDEA в Android Studio реализовано несколько новых, таких как новая подсистема сборки, тестирования и развертывания приложений, основанная на сборочном инструментарии Gradle и поддерживающая средства непрерывной интеграции. Разработку приложений ускорит коллекция типовых элементов

интерфейса и визуальный редактор для их компоновки, с удобным предпросмотром состояний интерфейса приложения (скажем, как интерфейс будет выглядеть для различных версий Android и различных размеров экрана). Для проектировки нестандартных интерфейсов присутствует мастер создания собственных элементов оформления, поддерживающий использование шаблонов. Расширенные инструменты рефакторинга, выявление проблем с производительностью и оценки удобства использования учли особенности API Android. В редактор добавлен режим быстрых правок, поддержка оптимизатора кода ProGuard и средства генерации цифровых подписей. Для приложений, разработанных в Eclipse и ADT Plugin, есть инструмент автоматического импорта.

ПАТЕНТЫ И ОТЧИСЛЕНИЯ

Мзда с производителя чипов

Microsoft и Foxconn заключили связанное с Android и ChromeOS патентное соглашение.

Компания Microsoft вернулась к практике сбора отчислений с производителей оборудования на базе открытых платформ Android и ChromeOS, манипулируя заявлениями о нарушении патентов в продуктах на базе данных ОС. На этот раз соглашение о выплате отчислений заключено с компанией Hon Hai, известной широкой публике под брендом Foxconn. Foxconn занимается производством по контракту со многими известными производителями, в том числе производит приставки Xbox для Microsoft, iPhone/iPad для Apple и телефоны для Motorola. Приблизительно 40 % потребительской электроники в мире произведено на мощностях Foxconn.

В рамках подписанного соглашения Foxconn признал наличие в платформах Android и Chrome OS технологий, нарушающих патенты Microsoft, и согласился выплачивать лицензионные отчисления за каждый произведенный смартфон, планшет и телевизор, укомплектован-

ный Android или Chrome OS. Сумма отчислений держится в тайне. До сих пор Foxconn являлся одним из крупных производителей, отказывавшихся заключать соглашение и не принимающих претензии компании Microsoft. Из крупных производителей, не заключивших соглашения с Microsoft, остались только Motorola, Huawei и Sony.

В настоящее время около 75 % из продаваемых в США устройств на базе платформы Android выпускаются компаниями, выплачивающими отчисления Microsoft. Сделки за закрытыми дверями, подразумевающие выплату отчислений за нарушение патентов в платформе Android, ранее были подписаны с компаниями Samsung, Acer, ViewSonic, HTC, LG, Quanta Computer, Velocity Micro, General Dynamics, Onkyo, Wistron Corp, Coby, Aluratek, Compal Electronics и Barnes and Noble. Во всех случаях список фигурирующих в соглашениях патентов и финансовые детали сделок не разглашаются.

Причем вектор нападения Microsoft уже не направлен только на Android, но и затрагивает всю экосистему Linux. Так как Chrome OS технически мало отличается от других Linux-дистрибутивов, и с учетом того, что компания Casio ранее заключила соглашение о выплате отчислений за использование Linux, Microsoft пытается сформировать ничем не подкрепленное мнение о наличии в Linux интеллектуальной собственности Microsoft, за которую необходимо платить роялти.

Еще в декабре 2003 года Microsoft заявила, что Linux нарушает принадлежащие ей 235 патентов (Linux-ядро — 42 патента, элементы графических интерфейсов пользователя (меню, панели инструментов) — 65 патентов, пакет *OpenOffice.org* — 45 патентов, почтовые клиенты — 15 патентов, другие открытые программы — 68 патентов). Указанные патенты так и не были раскрыты, а все связанные с ними сделки проводились с обязательной подпиской о неразглашении.

ИССЛЕДОВАНИЯ РЫНКА

Лидер — он и в России лидер

Android держит уверенное лидерство на российском рынке.

Интересные исследования провел аналитический отдел компании IDC. Согласно исследованию, доля продаж устройств с OS Android на российском рынке выросла с 39,7 до 70,9 % — почти вдвое по сравнению с прошлым годом. Данное исследование проводилось по запросу компании Mail.ru.

Аналитики IDC заявили, что самая большая доля по итогам первого квартала 2013 года принадлежит версии Android 2.3 и составляет 45,9 %; в прошлом году она лидировала с цифрой 66,6 %. Последняя версия Android 4.1 на данный момент установлена лишь на 15 % устройств.

Специалисты из Mail.ru также добавили, что более 50 % покупателей в России больше всего интересуются смартфонами на OS Android, а не на iOS и Windows Mobile. Итоги первого квартала 2013 года показали, что 70,9 % ввозимых в Россию мобильных устройств составляют уст-

ройства, работающие на Android, они и держат уверенное лидерство.

Если Google улучшит автономность работы своей операционки, то в будущем наметится еще более резкий рост продаж, тем более что в России у Android нет серьезных конкурентов.

«Доля продаж устройств с Android выросла почти вдвое.»

Специалисты компании NordFX считают, что вскоре ОС Android станет еще популярнее, если объединится с Chrome.

IDC проанализировала и положение на мировом рынке мобильных устройств в первом квартале 2013 года. Результаты по-прежнему в пользу Android и iOS: они занимают 92,3 % от всех поставок

смартфонов за этот период. Однако доля iOS сократилась. Доля Android повысилась на 59,1 %, всего было поставлено 199,5 млн смартфонов. Еще год назад эта цифра составляла 125,4 млн.

Что же до доли Apple, то в целом продажи в первом квартале 2013 года очень сильно возросли — это был лучший первый квартал Apple за все время — но в процентах доля мобильных устройств на iOS сократилась. Если поставки «андроидных» устройств составили 75 %, то устройств на iOS — только 17,3 %. Конечно, тут речь идет о поставках, а не о проданном оборудовании — известно, что Apple продает практически все, что выпускает, и доля трафика, генерируемого iOS, до сих пор выше, чем у Android.

Интересно, что Windows Phone благодаря Nokia наконец обогнала BlackBerry — ее доля выросла на 133,3 % и составила 3 % от всех поставок смартфонов. **LXF**

Сравнение

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

Редакторы изображений

Нужно приправить вашу работу графикой, или вы просто любите рисовать? **Марко Фиоретти** представляет пять редакторов, способных помочь вам в этом деле.



Про наш тест...

Пять описанных на этих страницах редакторов тестировались на 64-битном компьютере с Fedora 17, со всеми текущими заплатками.

По причинам, объясненным во введении, мы устанавливали и тестировали версии, имеющиеся в стандартных онлайн-репозиториях для Fedora. Нам надо было выяснить основные функции каждой программы и проверить, просто ли будет среднему пользователю использовать ее для работы начального уровня. Именно по этой причине, в отличие от большинства Сравнений, некоторые экранные снимки были получены не из наших тестов, а прямо с сайта программы.

Наша подборка

- » GIMP
- » Inkscape
- » Krita
- » MyPaint
- » Pinta

Одна картинка стоит тысячи слов. А вам известно, что, по данным некоторых источников, этому высказыванию почти сто лет? Впрочем, оно верно не всегда: это знают все, кто хоть раз бегло просматривал перенасыщенные графикой сайты при очень медленном соединении. И все же хорошо сделанные логотипы, постеры, оргсхемы, фоновые и многие другие изображения могут намного эффективнее донести информацию, чем их чисто текстовые соперники. А еще важнее тот факт, что рисование может доставлять огромное удовольствие — и для многих

из нас является такой же потребностью, как чтение или занятия спортом.

Но как сделать эту работу в Linux, или хотя бы в свободном ПО? Наше Сравнение постарается помочь максимуму среднестатистических пользователей найти свой ответ на этот вопрос. Мы сознательно не пытались выяснить, какая программа лучше для художников и про-

фессионалов, занимающихся графикой. Вместо этого мы искали программу, простую как в установке, так и в использовании, для создания простых рисунков, постеров, диаграмм и брошюр, или просто чтобы рисовать на компьютере ради забавы или дохода. Это привело нас к пяти растровым и векторным приложениям, которые вы и найдете на этих страницах.

**«Картинка стоит тысячи слов.
А рисование может доставлять
огромное удовольствие.»**

Совместимость и межплатформенная поддержка

Нет ли риска угодить в западню?

Эти две проблемы всегда актуальны для программ, однако для графических редакторов они еще более взаимосвязаны и важны, чем обычно. В марте 2013 krita.org только лишь упомянул «очень экспериментальную программу установки для Windows». А вот офисный пакет *Calligra*, включающий *Krita*, предложил на calligra.org/get-calligra «предварительную поддержку Mac OS X и Windows». Возможно, к тому времени, когда вы приступите к чтению нашего Сравнения, эта ситуация изменится.

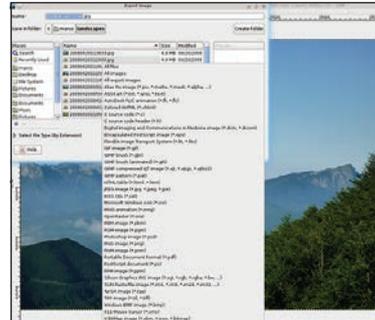
Все остальные редакторы отлично работают на Windows и Mac OS. *GIMP* даже обзавелся сторонними программами установки для обеих платформ на Gimp-win.sourceforge.net и Gimp.lisanet.de.

Совместимость тесно связана с форматами файлов. В принципе, при любом конвертировании графики из одного формата в другой вы можете что-то потерять, будь то разрешение, глубина цвета, слои или метаданные. Теоретически вы должны создавать и сохранять изображения только в родном или предпочтительном фор-

мате файлов используемого редактора, пока не закончите работу в нем. В противном случае определенные метаданные или даже слои могут быть потеряны.

Эти потери могут быть значительными, особенно при переходе с растровых форматов на векторные и наоборот. Причина кроется в разнице между ними. Растровые изображения — матричные или пиксельные, а векторные соединяют линии и другие объекты, создаваемые более или менее сложными инструкциями. Это важно, поскольку *Inkscape* разработан специально для векторной графики, а остальные — для растровой. *Inkscape* умеет делать импорт в PostScript и записывать и читать несколько растровых форматов. Но если *Inkscape* нужно применять совместно с растровыми инструментами, возможно, лучше будет поставить его только в самое начало или самый конец рабочего процесса.

На практике, кроме JPG и PNG, *Pinta* также принимает TGA, ICO, TIFF и BMP. *Krita* может импортировать и экспортировать изображения PNG и JPEG, читать XFC,



» Список форматов, в которых *GIMP* умеет сохранять изображения, настолько длинен, что не умещается на экране...

PDF и некоторые форматы фотокамер или выполнять импорт в BMP. *GIMP* распознает практически все существующие графические форматы.

Что касается совместимости, четыре растровых программы также поддерживают OpenRaster (ORA, freedesktop.org/wiki/Specifications/OpenRaster), хотя в *GIMP* для этого нужен плагин. ORA — это замена PSD для обмена многослойными растровыми изображениями, дополненными метаданными слоев. ORA — хороший выбор, если не необходимость, если вы используете более одного растрового редактора. Например, *MyPaint* сохраняет слои по отдельности только в формате ORA!

Вердикт

GIMP ★★★★★

Inkscape ★★★★★

Krita ★★★★★

MyPaint ★★★★★

Pinta ★★★★★

» Если межплатформенная поддержка и форматы для вас важны, берите *GIMP*.

Поддержка текста

Картинка стоит 1000 слов — но и без слов не обойтись.

Инфографика — контекст нашего Сравнения — частенько сочетает текст и рисунки. И в этой категории наши редакторы показали себя очень по-разному. В стандартном интерфейсе *MyPaint* вообще нет функции Add Text, а в *Pinta* можно написать что-то поверх

изображения и настроить базовое форматирование (шрифт, размер и т. д.). Нелзя, однако, сказать, что это весело и легко: если, например, вам нужно изменить надпись, единственным решением будет стереть ее и вписать заново.

В этом отношении *GIMP*, *Krita* и *Inkscape* значительно лучше. Text Tool в *GIMP* 2.8 дает доступ ко всем параметрам форматирования (начиная с таких элементов, как размер шрифта (кегель), начертание и тип, и заканчивая цветом, смещением от опорной линии и кернингом) прямо на главном холсте, для тех слов, которые вы печатаете. Это намного удобнее, чем в преды-

дущих версиях, где все эти настройки вынесены в отдельное окно.

И совсем хорошая новость в том, что *Krita* и *Inkscape* работают со словами даже лучше, чем *GIMP*. Они сильно упрощают не только добавление текста к графике, но и возможность поиграть с ним, превратив его в элемент рисунка, а не в наложенную поверх рисунка нащлепку.

В *Inkscape* можно установить шрифт (включая шрифты, которые пишутся слева направо), настроить интервалы и кернинг по умолчанию и вставить одну или несколько строк текста. Более того, *Inkscape* позволяет легко написать текст на выбранной строке. И, конечно, и сам текст, и строка, на которой вы велели *Inkscape* разместить его, поддаются редактированию. В *Krita* есть похожая функция привязки текста, а также кисть Text Brush, которой можно воспользоваться для раскраски текста, в нескольких режимах.



» Панель форматирования текста в *Krita*: шрифт, цвет, начертание и возможность привязки текста к контуру или фрейму.

Вердикт

Inkscape ★★★★★

GIMP ★★★★★

Krita ★★★★★

Pinta ★★★★★

MyPaint ★★★★★

» Если в вашу графику включен текст, лучше всего вам подойдут *Krita* или *Inkscape*.

Интерфейс пользователя

Какая программа продуктивнее в использовании? Вот в чем вопрос...

Редактирование графики — обширная область, и в ней столько же граней, сколько возможностей эффективно создавать и редактировать изображения. Редакторы изображений, рассмотренные на этих страницах, сразу же подтверждают эту точку зрения, и у каждого есть своя сфера особого внимания и свои методы выполнения работы.

GIMP пытается стать своего рода мастером на все руки, и ему это более или менее удается. *MyPaint* почти исключительно концентрируется на ручном рисовании, а *Krita* и *Pinta*, каждый по-своему, осели где-то на полпути между этими крайностями.

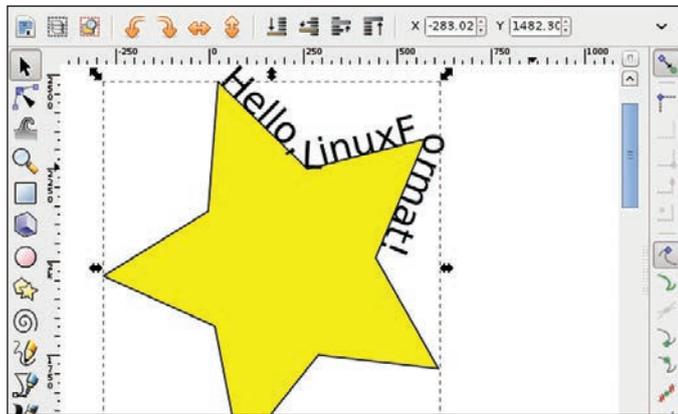
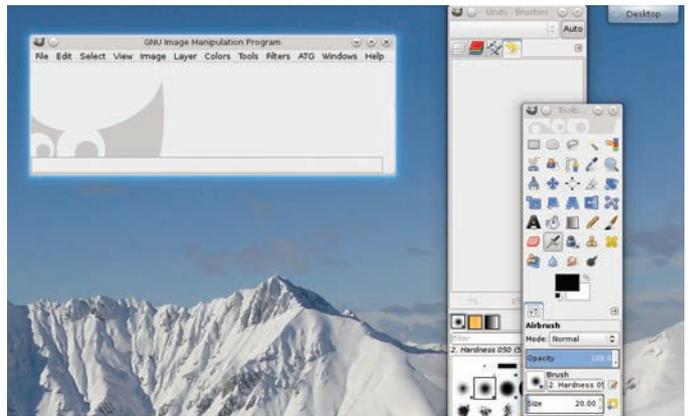
А вот *Inkscape* решил помочь вам освоить рисунки, которые по существу являются всего лишь

серией уравнений или последовательностью команд.

Сравнивать интерфейсы инструментов со столь разными целями — это все равно, что сравнивать яблоки даже не с апельсинами, а с ветряными мельницами. Поэтому мы просто попытались в предоставленных нам сжатых рамках упомянуть характерные свойства каждой программы.

GIMP ★★★★★

GIMP — растровый редактор общецелевого назначения, уже широко используемый для коллажей, сложного редактирования фото и всего прочего между этими задачами. Чтобы справляться со всем этим, *GIMP* завел солидный пользовательский интерфейс. По умолчанию эта программа умудряется втиснуть на ваш экран (как минимум) три окна, да еще кучу всего за пару щелчков. Если вам это неудобно, переключитесь в режим Single Window Mode [Однооконный Режим]. У вас есть одна вкладка на каждое открытое изображение в центре окна, плюс множество панелей по бокам. Более подробную информацию вы найдете в окне UI customizability [Настройка интерфейса]. А сейчас просто запомните, что *GIMP* понимает Save [Сохранить] только как Save as XCF [Сохранить в формате XCF], поскольку это родной формат *GIMP*. Чтобы выбрать любой другой формат, надо перейти в Export — но делайте это только по окончании редактирования!



Inkscape ★★★★★

Inkscape был рожден для векторной графики, что становится очевидно после двухсекундного обозрения его кнопок. В *Inkscape* абсолютно все является объектом, соединяемым с другими объектами. Это отлично подходит для создания оргсхем или блок-схем.

Объекты *Inkscape* легко поворачивать, обрезать и маскировать разными способами, или размещать в разных слоях. Инструментом Shape Tools создаются скругления, звезды, многоугольники и другие фигуры. *Inkscape* также поддерживает сложные градиенты и клоны live — т.е. копии объекта, изменяющиеся вместе с ним при всяком редактировании оригинала. Линии изображаются инструментом ручного рисования или с помощью инструмента Bezier — просто щелчками в точках, где должна пройти линия. Все рисунки сохраняются в виде Standard Vector Graphic (SVG) — это основанный на тексте формат, его можно редактировать в XML-редакторе *Inkscape*.

Мультимедиа и анимация

Что-о?.. Нет видео?!

В основном, у героев нашего Сравнения поддержки анимации нет, ну разве что самые примитивные действия. В *Pinta* нет вообще ничего, а в других — весьма немного.

Первыми идут *GIMP* и *Inkscape*. Опциональный пакет *GIMP* Animation Package (gimp.org/tutorials/Using_GAP) позволяет объединять изображения в разного вида циклы, например, Direct [Прямой], Reverse [Обратный] или Pingpong [Пингпонг], т.е. бесконечный цикл, когда

переходы делаются сначала в одном порядке, потом в противоположном. В спецификациях стандарта SVG, на котором построен *Inkscape*, указано, как вставить в файл несколько анимаций на базе временной шкалы. Каждая анимация может иметь свой, фиксированный или управляемый пользователем триггер. Однако *Inkscape* не поддерживает всех возможностей данного стандарта. Можно также создать анимацию, поместив единичный кадр в отдельный слой: функция Paste In Place

[Вставить на место] явно и предназначена для такой работы, а расширение JettyInk умеет экспортировать каждый слой как отдельный файл PNG. Анимация в *Krita* или *MyPaint* годится скорее для пользователей, уже имеющих определенные навыки и время для возни с программой. В *Krita* должен быть плагин для анимации, но его разработка засиделась в 2011-м. У *MyPaint* есть две экспериментальных ветви с поддержкой анимации по слоям, а-ля *Inkscape*. Детали см. в wiki.mypaint.info/Animation.

Вердикт

- GIMP ★★★★★
- Inkscape ★★★★★
- Krita ★★★★★
- MyPaint ★★★★★
- Pinta ★★★★★

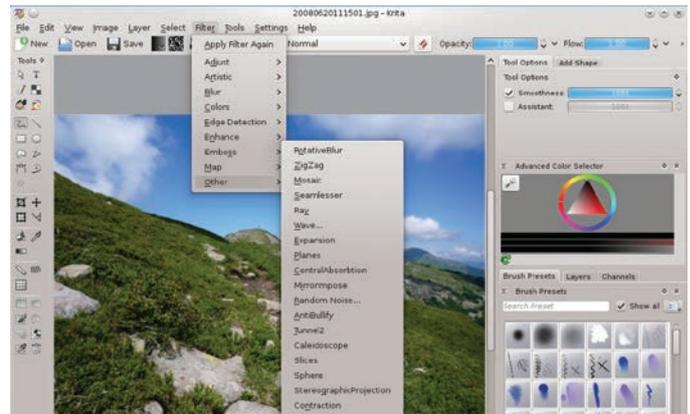
» Эти редакторы не предназначены для анимации; не ждите от них чудес.

Krita ★★★★★

Возможно, вы использовали продуманный и упорядоченный интерфейс *Krita* для обработки существующих изображений, однако он был создан не для этого. *Krita* по-шведски значит «мелок». Это приложение было создано для создания рисунков с нуля на компьютере с целью добиться максимального сходства с рисованием нецифровым способом. Именно поэтому его разработчики заявляют, что «это не самое большое удовольствие без графического планшета». Но если вы спросите наше мнение, мы скажем, что это не совсем справедливо.

В *Krita* также можно рисовать с помощью векторных и растровых инструментов и десятков фильтров, сохраняя каждую часть изображения в отдельном слое, с глубиной цвета до 32 бит с плавающей точкой на канал.

Как часть *KOffice* и *Calligra*, *Krita* может стать лучшим выбором для тех, кто создает графику для вставки в текст или в презентацию.



Pinta ★★★★★

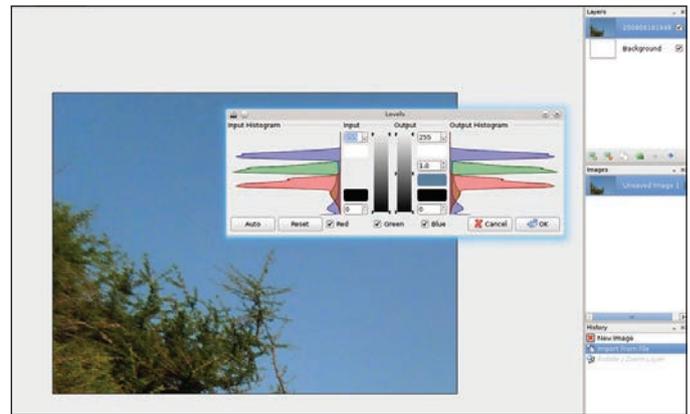
Создание *Pinta* было вдохновлено Paint.NET. Вам предлагаются все необходимые базовые инструменты для настройки яркости, контраста и других фотографических деталей, и дается неограниченное количество слоев, функция скриншота и журнал, так что вы всегда можете отменить любое действие текущей сессии.

Два особых инструмента *Pinta* — это Magic Wand [Волшебная палочка] и опция Rescolor [Переокрасить]. Magic Wand применяется для быстрого выбора области с цветом, подобным цвету в текущем выделении. Rescolor изменяет цвет только внутри уже окрашенных областей. Вы можете рисовать вручную, двигая курсор, как вам заблагорассудится, включая замкнутые фигуры и автоматически замыкаемые выделенные области. Последняя функция именуется Lasso Selection Tool. *Pinta* также имеет массу графических настроек и эффектов для мелких — или крупных — уточнений в текущем изображении.

MyPaint ★★★★★

MyPaint в основном создан для работы с графическим планшетом, во всем многообразии стилей. Ваши прикосновения к планшету (движения кисти) будут переведены в последовательность крошечных эллипсов или кружков, именуемых «мазками [dab]». Если вы попытаетесь выйти без сохранения, *MyPaint* сообщит вам, сколько минут работы вы можете потерять. Интерфейс, в основе которого лежит холст неограниченного размера с настраиваемым фоном, старается не мешать вам как можно дольше. Очень удобно, что функция Save Next Scar [Сохранить следующий вариант] (которая привязана к клавише F2) сохранит для вас сколько угодно копий рисунка.

И венчает все это отличная поддержка слов. Кисти *MyPaint* — их размер вы можете изменять даже во время работы с ними — вероятно, самые мощные во всей группе. Кроме того, есть широкий выбор опций настройки существующих кистей и даже создания собственных.



Настраиваемость интерфейса

Вас не устраивает положение кнопки? Ну так переместите ее...

Иногда любовь или ненависть к приложению определяется (не)возможностью разместить элементы графического интерфейса там, где нам хочется. С этой точки зрения *MyPaint* и *Inkscape* наименее гибки из всех. *Krita* и *Pinta* позволяют создавать любые комбинации закрепленных и плавающих окон; ну, а *GIMP* отличается наибольшей настраиваемостью. Быть может, в том-то и его беда: здесь столько всего, что волей-неволей надо это как-то разме-

щать и перемещать. Но ведь главное — это выбор пользователя, правда? При желании можно изменить во внешнем виде *GIMP* практически все, от размера значков до цветов и тем виджетов. Используя *GIMP* на планшете, вы также найдете специальные слайдеры и диалоговые окна для настройки силы давления и ручки. Однако это не самые важные опции настройки, которые он предлагает.

GIMP объединяет сотни функций в панели, именуемые также доками. Вы мо-

жете расположить эти панели на свой вкус — по всему рабочему столу или рядами в одном окне. Нажатие на кнопку Tab откроет либо закроет все доки. Однако для максимизации производительности вам придется найти такую раскладку, которая будет удобна именно вам, и научиться использовать ее. И это солидная работа. Настолько солидная, что ее хватило на руководство Managing Dockable Dialogs в registry.gimp.org/node/20335. Вы предупреждены!

Вердикт

GIMP	★★★★★
Pinta	★★★★★
Krita	★★★★★
Inkscape	★★★★★
MyPaint	★★★★★

» Хотите все кнопки в одном окне? Большинство инструментов даст вам выбор.

Автоматизация и настройка

А можно сделать так, чтобы он поработал за меня?

Создание графики — это удовольствие. И мы все знаем, что это правда, если скинуть самую скучную работу на компьютер. В наших редакторах то, что вы можете на самом деле настроить или делегировать программе, сильно отличается в зависимости от программы, поскольку у них разные цели и области применения. В этих рамках каждое приложение может вам что-то предложить.

Вы можете найти несколько расширений для *Pinta* на сайте проекта, или написать собственные, если вы изучили Visual Studio или MonoDevelop. В *MyPaint*, похоже, нет поддержки режима пакетной обработки и автоматизации. Но это

не лишено смысла: зачем автоматизация программе, которая создана для как можно более естественного рисования? Другое дело — индивидуальная настройка. Вы, например, можете создать собственные кисти, в соответствии с объяснением в mypaint.intilinux.com/?page_id=173, или в ramonmiranda.com/2011/04/mypaint-how-to-make-brushes-onmy-paint.html.

Krita умеет загружать и запускать скрипты, написанные на Ruby, JavaScript или Python, которые соответствуют общим положениям среды создания скриптов Cross scripting framework (kross.dipe.org/dox/krita.html). Дополнительные фильтры можно найти в проекте *Krita-plugins*. Теоретически должно быть еще что-то в разделе *Krita Scripts* на kde-files.org, но в последнее время там собран чистый спам. Зато в *Krita* можно сохранить последовательность действий в виде макроса!

В *GIMP* допускается использовать язык скриптов под названием Script-Fu для автоматизации повторяющихся задач и добавления функций. Script-Fu даже может делать такие вещи, для которых не надо устанавливать скрипт,

типа рисования стрелок. Отдельный раздел сайта *GIMP* (registry.gimp.org) перечисляет плагины, которые можно создать на Script-Fu или других языках. Например, Save for the Web [Сохранить для Сети] удаляет метаданные и выбирает оптимальное соотношение между размером изображения и качеством перед загрузкой изображения в ваш блог. Возможно, вам понравится National Geographic, который придаст вашим фотографиям такой же вид и стиль, если не качество, что и в этом журнале.

В *GIMP* также имеется режим пакетной обработки, и плагин file-glob — для работы из командной строки или в скрипте оболочки. Если в оболочке ввести `gimp -i -b`, откроется интерактивная консоль, в которой можно запускать все команды Script-Fu, перечисленные в GIMP Procedure Browser. Чтобы узнать об этом больше, взгляните на страницу Gimp.org/tutorials/Basic_Batch.

Inkscape может запускать скрипты Python и имеет режим командной строки, как и *GIMP*, но он еще интереснее в плане автоматизации — в основном благодаря тому, что формат SVG — это XML, то есть в основном простой текст. При желании можно создать графику для *Inkscape* с помощью оболочки или скриптов Perl, а *Inkscape* открыть только для финальных штрихов.



» Редактор XML в *Inkscape*: насколько крута возможность создавать или обрабатывать сложную графику в виде простого текста?

Вердикт

Krita	★★★★★
GIMP	★★★★★
Inkscape	★★★★★
MyPaint	★★★★★
Pinta	★★★★★

» *Krita* получает высший балл, за поддержку скриптов на нескольких языках.

Документация

Кто поможет мне узнать, что надо делать?

А ну, быстро! Как называются действия, совершаемые в графическом редакторе для получения штуковины, которая описывается как «куб с разными изображениями на каждой грани»? Этот хитрый вопрос показывает, почему в графических редакторах документация нужнее, чем в других сферах. К счастью, ее почти всегда хватает!

Pinta обладает хорошей справочной информацией по инструментам, но сверх того предлагается только простое руководство по установке да справочник, где объяснено, как подготовить и установить расширения. Так что пользы тут мало.

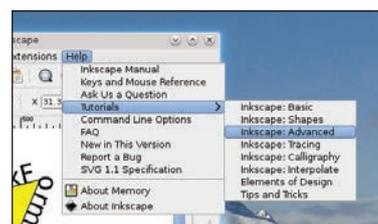
GIMP предлагает контекстную справку (жмите F1), руководство и три группы справочников на официальном сайте, плюс множество онлайн-сообществ. Есть

по крайней мере десять книг по *GIMP*, хотя часть из них изрядно устарела.

Даже в *MyPaint* неплохая справочная система. На домашней странице есть ссылка на руководство для начинающих, несколько сторонних статей и wiki. В последней вы найдете руководство по созданию рисунков, ресурсы по настройке кистей и список совместимости с оборудованием. Сайт code.google.com/p/mypaintatelier испаноязычный, но там есть несколько видеороликов — от рисования цветов до создания кистей — которыми вы сможете воспользоваться без всяких проблем. Посетите также чудесный канал *MyPaint* на mypaint.deviantart.com/!

У *Inkscape* отличный официальный справочник, также в виде печатной или цифровой книги на tvmjong.free.fr/

INKSCAPE. Там же есть файлы, помогающие разобраться с объяснениями, данными в книге. Самая полезная документация *Krita* — на krita.org/resources: кроме основных руководств, имеется серия видео о создании комиксов и два руководства Start to Finish [С начала до конца] по всему процессу рисования в *Krita*, *MyPaint* и *GIMP*.



» У *Krita* отличные справочники, а у *Inkscape* лучшее меню справки Help Menu.

Вердикт

Inkscape	★★★★★
Krita	★★★★★
GIMP	★★★★★
MyPaint	★★★★★
Pinta	★★★★★

» Ресурсы для *Krita* и *Inkscape* наиболее полны и просты в использовании.

Редакторы изображений

Вердикт

Если вы не ставите перед собой очень узких задач, а планируете регулярно работать с графикой, вам вряд ли хватит одного инструмента, какой бы вы ни выбрали. Нет такой вещи, как лучший графический редактор вообще. Но в нашем Сравнении мы все же рассматриваем только один класс, хоть и с широкой областью применения: «программы, простые как в установке, так и в использовании, для создания простейших рисунков, постеров, диаграмм и брошюр».

Благодаря решению ограничиться таким сценарием, мы полагаем, что можем почти уверенно выбрать одного победителя. Однако путь, который привел нас сюда, был нелегок.

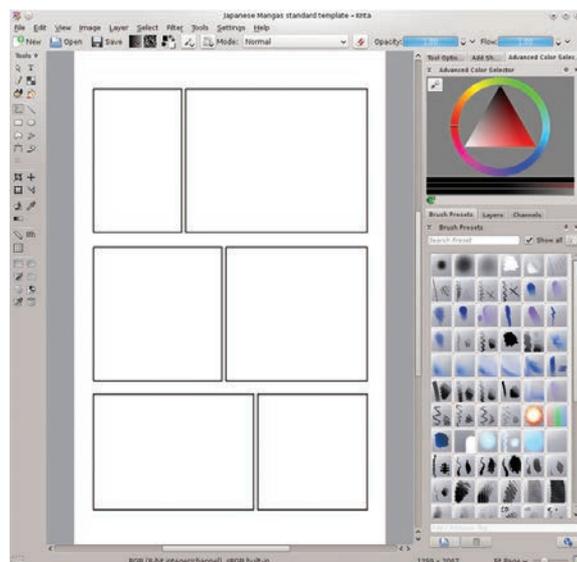
Pinta — хорошая программа, но в целом более ограничена, чем ее соперники, и ее легко было снять с дистанции. *MyPaint* кажется идеальным для избранной им миссии — ручного рисования; по той же причине мы полагаем, что

он не может считаться инструментом общего назначения.

Таковым является *GIMP* — крупный растровый графический редактор; быть может, даже слишком крупный и сложный для пользователя, которого мы имели в виду. Итак, остаются *Krita* и *Inkscape*, из них мы должны выбрать один. Мудрено.

Причина в том, что и *Krita*, и *Inkscape* очень хорошо делают одно и то же; и это — рисование и редактирование. Но делают они это в разных мирах: растровой и векторной графики. В них одинаково велики количество функций и возможностей индивидуальной настройки. И оба позволяют новичкам достаточно быстро научиться делать что-то интересное.

Это все равно, что решить, какая машина лучше — отличный городской седан или джип-внедорожник. Учитывая все вышесказанное, мы объявляем победителем *Krita*, однако *Inkscape* занимает второе место с очень незначительным



отрывом. Мы так решили по одной простой причине: наше внимание фокусировалось на несложных рисунках, постерах, диаграммах и брошюрах — то есть на том, что часто приходится включать в текст, буклеты и слайд-шоу. И тогда приложение из полного офисного пакета может оказаться более продуктивным. Но даже если вы, поверив нам, сделаете *Krita* своим основным графическим редактором, не забывайте обращаться и к *Inkscape*!

» Помимо всего прочего, в *Krita* есть готовые шаблоны для комиксов и иных категорий графической продукции.

I Krita ★★★★★
 Сайт: www.krita.org Лицензия: GPLv2+, LGPLv2+
 » Отличное сочетание функциональности и простоты, также и для не графической работы.

IV MyPaint ★★★★★
 Сайт: www.mypaint.intilinux.com Лицензия: GPLv2+, LGPLv2+, CC-BY
 » Менее многогранный, чем остальные, но со своей миссией справляется отлично.

II Inkscape ★★★★★
 Сайт: www.inkscape.org Лицензия: GPLv2+
 » Если бы мы могли обойтись только векторной графикой, *Inkscape* победил бы.

V Pinta ★★★★★
 Сайт: www.pinta-project.com Лицензия: MIT, CC-BY
 » Отлично сделан, но менее зрелый, полный и расширяемый, чем соперники.

III GIMP ★★★★★
 Сайт: www.gimp.org Лицензия: GPL v3+
 » Умеет практически все, но не дотягивает до самых высоких стандартов.

Обратная связь
 Вы нашли некую хитрость, еще не задокументированную? Шлите нам свои мнения об этом Сравнении и всем остальном на lxf.letters@futurenet.co.uk.

Рассмотрите также...

Если вы серьезно увлеклись графикой, одно можно сказать точно: рано или поздно вам придется использовать более одной программы. Будем надеяться, что информация из нашего Сравнения сэкономит ваше время при определении подходящего рабочего инструмента.

И в данном контексте стоит обратить внимание еще на одну вещь. В определенных видах работы

есть множество простых операций, которые просто позорно делать вручную.

Инструменты *ImageMagick* не вошли в наше Сравнение, потому что его сложно назвать графическим редактором, при любом определении графического редактора. Это не значит, что вы должны игнорировать *ImageMagick*. Если вам приходится много работать с графикой, не пожа-

лейте времени и узнайте, что это такое и как оно может вам пригодиться!

И потом, метаданные. На них тут места не было, и вообще-то они даже и не в тему. Но их нельзя игнорировать — особенно EXIF и IPTC — редактируя фотографии. Каким бы редактором вы ни пользовались, проверьте, чтобы он сохранял (или изменял) метаданные в ваших файлах! **LXF**



СИСТЕМ

ПРОВЕРЕНО!

Какая вам больше подходит? Здесь полсотни лучших, по рангам и категориям.

Выбор и гибкость — фирменные признаки экосистемы Linux. В Windows и OS X, если вам не по душе какие-либо аспекты операционной системы, с этим мало что поделаешь. В мире Linux вы, можно сказать, избалованы выбором благодаря наличию множества дистрибутивов.

Сердцевина каждого дистрибутива — ядро Linux, но на его основе дистрибутивы достраивают свои компоненты, которые подбираются в зависимости от целевой аудитории данного дистрибутива. Разные дистрибутивы предлагают разные опции индивидуальной настройки, так что вы можете поработать с дистрибутивом и настроить его по своему вкусу и усмотрению, пока не добьетесь желаемого результата. И независимо

от того, какой тип пользователя вы собой представляете, для вас обязательно найдется подходящий дистрибутив.

В этой статье мы в первую очередь уделяем внимание рабочему столу. Некоторые дистрибутивы рабочего стола ставят своей целью максимизировать простоту, другие предоставляют вам больше контроля. У них разные процедуры установки, разные требования к рабочему столу, разные схемы управления пакетами и разные

инструменты администрирования. Мы рассмотрим привычные дистрибутивы, применяемые для обычных задач, а также разработанные специально для привлечения пользователей, привыкших к другим операционным системам, таким, как Windows и Mac OS X. Мы также рассмотрим дистрибутивы, способные превратить старую рухлядь в современную машину, и те, что предоставляют опытным пользователям полный контроль над их рабочей средой.

На следующих страницах мы рассмотрим более четырех дюжин лучших дистрибутивов, протестируем их уникальные функции и сравним их сильные и слабые стороны, чтобы вы смогли с нашей помощью выбрать дистрибутив, лучший именно для вас!

«Мы рассмотрим более четырех дюжин лучших.»

мальную простоту, другие предоставляют вам больше контроля. У них разные процедуры установки, разные требования к рабочему столу, разные схемы управления пакетами и разные

Индекс

aLinux 33	Foresight Linux 35	Mageia 27	Porteus 32	Slitaz 31
Arch 35	Frugalware 35	Manjaro 30	Puppy 32	SolusOS 34
Bodhi Linux 33	Fuduntu 32	Netrunner 31	PureOS 27	Sparky Linux 32
CentOS 35	Gentoo 35	Oantix 33	Qubes OS 30	Tiny Core Linux 33
Chakra 29	Kanotix 30	Open Xange 30	Rosa 29	Ubuntu 27
Crunchbang 33	Knoppix 29	OpenSUSE 27	Sabayon Linux 28	Ultimate Edition 34
CruX 35	Kororaa 30	OS4 34	Salix OS 29	WattOS 30
Debian 35	Linux Mint 28	Parsix 34	Slackel 27	Zenwalk 31
DSL 33	LinuxLite 34	PCLinuxOS 28	Slackware 35	ZevenOS 31
Fedora 27	LXF Games 35	Pear Linux 34	Slax 32	Zorin 34

На каждый день

Дистрибутивы, разработанные на замену вашей нынешней ОС.

Ubuntu

Это не первый дистрибутив для новичков в Linux, но он, бесспорно, самый известный. В нем есть несколько инновационных функций, в том числе рабочий стол Unity, который всем так нравится ненавидеть. Затем имеется Ubuntu One, многоплатформенный сервис для облачного хранения и синхронизации файлов по множеству устройств, который предлагает 5 ГБ свободного места для хранения данных. А для добавления новых программ предусмотрено *Ubuntu Software Center*.

У этого дистрибутива один из самых простых механизмов установки. По умолчанию он не содержит проприетарных кодеков, но их можно включить во время установки, просто щелкнув по флажку. Дистрибутив выходит дважды в год, с постоянными релизами долгосрочной поддержки (Long Term Support, LTS) на пять лет.

Вердикт: Unity улучшается, и Ubuntu продолжит борьбу с проприетарными настольными ОС.



OpenSUSE

Еще один ветеран и один из ведущих пользователей (и участников разработки) рабочего стола KDE — хотя официально они не объявляли о своих предпочтениях. На самом деле, этот дистрибутив достойно ведет себя на обоих рабочих столах, и визуально весьма привлекателен.

Его инструмент управления *YaST* (Yet another Setup Tool), работающий по принципу все-в-одном, умеет устанавливать программы, а также настраивать и администрировать систему. Хотя иметь все инструменты

в одном месте довольно удобно, все же это довольно сложно и слегка пугает новых пользователей Linux. Да и установщик этого дистрибутива не так прост, как в Ubuntu. А также, в нарушение всех традиций, дистрибутив выпускает новый релиз каждые восемь месяцев.

Вердикт: Хотя этот дистрибутив и разработан для пользователей настольных ПК, он не столь дружелюбен, как Ubuntu или Mint, но станет отличной ОС для предприятия.



Fedora

Последний релиз этого дистрибутива, традиционно считающегося альтернативой Ubuntu и разрабатываемого сообществом при поддержке Red Hat, отдал предпочтение серверным функциям против настольных. И поэтому он скорее игровая площадка для опытных пользователей. Да и вообще, это тестовый полигон для будущих предложений Red Hat.

Кроме того, дистрибутив стал сложнее в использовании из-за введения рабоче-

го стола Gnome 3 (постепенно улучшающегося). Новички оторопеют при виде голого экрана, требующего особых навыков, чтобы разобраться в навигации. Отсутствие рабочего менеджера пакетов и мультимедиа-кодеков, а также антиинтуитивный инсталлятор тоже не радуют.

Вердикт: Весьма перспективный дистрибутив для опытных пользователей, миновавших кризис среднего возраста.



Slackel

Этот дистрибутив берет самые свежие программы из ветки Slackware Current и предлагает их в виде дружелюбного настольного пакета, взяв ряд инструментов из богатого на них дистрибутива Salix OS. Основанные на KDE образы Slackel имеются на носителях только для установки и для установки в режиме

live. Некоторые инструменты от Salix OS — программа установки кодеков, менеджер пакетов *Gslapt*, а также программа установки live CD.

Вердикт: Возобновляемый релиз, который предоставляет все самое современное из Slackware, используя инструменты из Salix OS.



PureOS

Дистрибутив имеется в двух видах; основан или на Gnome, или на легковесном рабочем столе *Openbox*. PureOS основан на тестовой ветке Debian и поддерживает многоязычные локализации. PureOS доступен только в виде образов для создания Live USB-дисков. Цель его — быть функциональным дистрибутивом,

который затем можно персонализировать, добавив модули с помощью включенных в него скриптов. Также содержит скрипт *smxi* для тонкой настройки системы и установки проприетарных драйверов.

Вердикт: Гладкий, готовый к работе дистрибутив для умельцев.



Mageia

Mandrake Linux был первым дистрибутивом Linux для обычных пользователей. Он прошел через ряд перевоплощений, и его последняя аватарка заставила членов его сообщества со стажем уйти в ответвление. Теперь дистрибутив на базе RPM именуется Mageia и поддерживается солидной инфраструктурой сообщества. Он предлагает рабочие столы KDE и Gnome, и в трех репозиториях дистрибутива найдутся все нужные вам про-

граммы. Его программа установки проста в навигации, и несколько экранов имеют кнопку *Advanced*, с опциями для опытных пользователей. Самая выдающаяся функция этого дистрибутива — *Mageia Control Center*, здесь можно настроить почти все аспекты системы.

Вердикт: Ответвление на базе сообщества на солидном фундаменте, и достойный дистрибутив для повседневных задач.



PCLinuxOS

Этот дистрибутив начинал как репозиторий для улучшения основного релиза Mandriva, а позднее стал отдельным дистрибутивом. PCLinuxOS официально является дистрибутивом KDE, однако имеет отвлечения от сообщества на основе рабочих столов LXDE и Xfce.

Дистрибутив умеет воспроизводить все виды мультимедиа. Для установки RPM-пакетов он применяет *apt-rpm* и *Syn-*

aptic Package Manager. Его инструмент настройки и программа установки явно отвергает родство с Mandriva. Дистрибутив содержит иллюстрированное руководство по установке, а также выпускает для своих пользователей ежемесячный журнал.

Вердикт: Считайте его чем-то наподобие Mageia, но с уже заготовленными мультимедиа-кодеками.



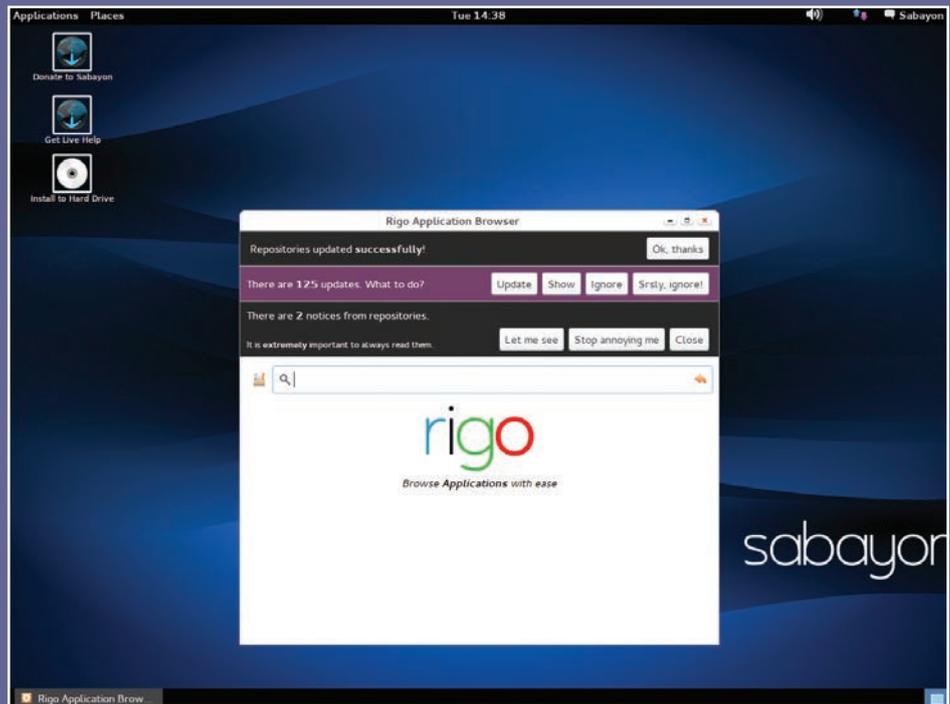
Sabayon Linux

Цель проекта Sabayon — предоставить постоянным пользователям настольных компьютеров возможность познакомиться с дистрибутивом Gentoo. Это богатый функциями дистрибутив, созданный на стабильной и зрелой основе.

В Sabayon есть все необходимое для пользователя настольной системы, в том числе все виды кодеков и модулей расширения (плагинов). Помимо обычного множества приложений, в нем также есть медиа-плеер *XVMS* и *Wine* для работы с приложениями Windows.

Для управления пакетами имеется настраиваемый браузер приложений *Rigo*. Это графическая оболочка для *Entropy*, системы управления пакетами Sabayon, и она имитирует внешний вид поисковой машины Google. Приложение просто в работе и очень разговорчиво. Вместо отображения загадочных сообщений оно объясняется с пользователями на человеческом (английском) языке.

У дистрибутива есть разновидности для всех популярных рабочих столов, в том числе для Gnome, KDE, Xfce и Mate. И этот дистрибутив поставляется не исходный релиз Gnome, а производит некоторые настройки — например, чтобы отображать уменьшенные кнопки — для поддержки единообразия разных релизов. Вы найдете в его официальных репозиториях и другие рабочие столы, например, Cinnamon и Razor-Qt. Помимо этого, есть несколько спецразновидностей для



особых целей: например, минималистский CoreCDX и HardenedServer на базе Gentoo Hardened kernel.

Хотя Sabayon — дистрибутив с обновляемым релизом [rolling release], разработчики настроили его так, чтобы работа с ним была удобна для начинающих пользователей.

Установка производится переработанной программой установки *Anacoda*, созданной дистрибутивом Fedora. Также

можно загрузить 64-битные образы более новых версий — в системы с включенным SecureBoot. На сайте дистрибутива вы найдете массу документации, удобной для начинающего пользователя и включающей пошаговое руководство по установке и подробный FAQ.

Вердикт: Дистрибутив предлагает всю мощь и гладкость Gentoo в смягченном и дружелюбном к пользователю настольном пакете.



Linux Mint

С момента своего дебюта в 2006 Linux Mint медленно, но верно перемещался к вершинам всех рейтингов любимых дистрибутивов. Одна из основных причин успеха дистрибутива в том, что, хотя он основан на Ubuntu, его рабочий стол по умолчанию гораздо более традиционен, чем спорный интерфейс Unity в Ubuntu. Linux Mint предлагает пользователям выбор из двух сред рабочего стола на основе Gnome, которые он назвал Mate и Cinnamon.

Mate создан для того, чтобы быть преданным продолжателем старого рабочего

стола Gnome 2. Cinnamon — поновее, с опрятным меню, дающим доступ ко всем настройкам системы и приложениям из одного центрального пункта.

Дистрибутив также идет с предварительно загруженным полным набором аудио- и видеокodeков и обладает впечатляющим *Software Manager*.

Вердикт: Простой в установке и отполированный рабочий стол, который с ходу начинает работать. Идеален для вас, если пресловутый Unity вам против шерсти.





Rosa

Дистрибутив создан московской лабораторией Rosa Labs, которая работала над последним релизом Mandriva. Потом Rosa ответилась, превратившись в отдельный дистрибутив. Основное внимание в нем сконцентрировано на рабочем столе KDE, но через пару месяцев после выхода релиза KDE дистрибутив выпускает разновидность на базе Gnome.

От прочих дистрибутивов на базе Mandriva Rosa отличается инструментами пользователя, которые делают их рабочий стол KDE непохожим на любой другой. Здесь своя программа запуска при-

ложений и свое стартовое меню, внешне похожее на Unity Dash и меню Activities в Gnome 3. Кроме переделанного рабочего стола, в дистрибутиве есть ряд усовершенствованных функций. Инструмент *TimeFrame* использует *Neptun* от KDE для отображения всех ваших файлов, видео и музыки в уникальной и приятной манере. Более новые версии этого инструмента также поддерживают социальные сети вроде Facebook.

Вердикт: Попробуйте его ради рабочего стола KDE с индивидуальной настройкой.



Рейтинг

Chakra

Это, бесспорно, самый привлекательный внешне дистрибутив рабочего стола KDE, и у большинства пользователей он работает прямо сразу. Изначально дистрибутив основывался на Arch Linux. Это наполнило скользкий релиз, который, по его собственному признанию, предназначен для пользователей, которые не боятся командной строки.

Его менеджер пакетов все еще в стадии разработки, но у него есть достойная замена в виде менеджера «пучков» [bundles

manager]. Пучки — это автономные комплекты популярных приложений, устанавливаемые одним щелчком. Установка делается программой, которая использует KDE Partition Manager для разбиения диска на разделы. Также здесь есть впечатляющее приложение персонализации первой загрузки [first-boot personaliser].

Вердикт: Очень симпатичный рабочий стол KDE, и он помогает вам индивидуально настроить свою рабочую среду.



Рейтинг



Knoppix

Пользователи Linux со стажем тепло вспоминают Knoppix как первый Linux live CD. Этот дистрибутив содержит все виды программ из стабильных, тестируемых и нестабильных репозиториях Debian.

Дистрибутив доступен в двух версиях — образ Live CD предоставляет более 2 Гб программ, а образ DVD умудрился втиснуть более 8 Гб. Фактически это единственный live-дистрибутив, содер-

жащий три среды рабочего стола — LXDE (по умолчанию), а также Gnome и KDE. А есть еще официально поддерживаемая разновидность для слабовидящих пользователей, которую также могут использовать начинающие пользователи компьютера.

Вердикт: Никто не справится лучше него с задачей демонстрации лучших программ с открытым кодом.



Рейтинг

Salix OS

Разработчики Salix OS рассматривали свой дистрибутив как некий бонсай — маленький, легкий и являющийся продуктом бесконечной заботы; и с этим нельзя не согласиться. Оформление дистрибутива ласкает глаз благодаря подборке обоев, значков и тем.

Основанный на Slackware дистрибутив имеется в шести разных версиях — с рабочими столами KDE, Xfce, Mate, LXDE, Fluxbox и Ratpoison. Salix быстр и прост в использовании. Он содержит 78-страничное руководство, и сайт дистрибутива изобилует дополнительной документацией по отдельным задачам.

Хотя кодаков Salix не содержит, в нем есть программа их установки за один щелчок. И вообще, в этом дистрибутиве мно-

жество пользовательских инструментов внутренней разработки. Среди наиболее интересных, помимо установщика кодаков: графическая утилита для клонирования live-системы, мастер создания постоянного хранилища и графический инструмент для выполнения простых административных задач, например, добавления новых пользователей.

Есть и графическая программа установки Salix, и она доставляет приятное разнообразие в ряду привычных инсталляторов. Для управления пакетами дистрибутив использует графический менеджер пакетов *Gslapt*, являющийся интерфейсом для *slap-get*. Поскольку дистрибутив полностью совместим со Slackware, в нем также есть графический инструмент для компиляции пакетов из репозитория SlackBuilds.



Рейтинг

Вердикт: Для пользователей, заинтересованных в стабильности Slackware и удобстве менеджера пакетов.

Qubes OS

Практикует подход к обеспечению безопасности, известный как безопасность через изоляцию, и по этой причине безопаснее, чем средний дистрибутив рабочего стола. Каждое приложение в Qubes работает внутри собственного домена безопасности. Новая установка автоматически создает такие домены, как Work [Работа], Personal [Личное] и Untrusted [Непроверен-

ное], и делит приложения между ними: вы найдете *LibreOffice* в домене Work, *Firefox* в домене Untrusted, и т.д. По большей части Qubes OS работает как любой другой дистрибутив рабочего стола, но вместо категорий приложений его меню KDE выдает список разных доменов.

Вердикт: Уникален, безопасен и легко осваивается.



Open Xange

Этот дистрибутив на базе Fedora использует слегка подправленный рабочий стол KDE, сделав его чуть понятнее для традиционного пользователя Windows. Кроме обычных приложений рабочего стола, этот дистрибутив также предоставляет солидную подборку плагинов и кодеков. На рабочем столе вы найдете кнопку для установки проприетарного клиента Dropbox.

Для управления другими программами с открытым кодом Open Xange применяет *Apper*, интерфейс к *PackageKit* от KDE. От Fedora он унаследовал программу установки *Anaconda*. Дистрибутив не поддерживает 32-битную архитектуру.

Вердикт: Основанный на Fedora рабочий стол, которому не удастся превзойти Kororaa.



WattOS

Целью этого дистрибутива было донести все прелести Ubuntu до слабых машин, используя легкие приложения, например, рабочий стол *LXDE*. Он поставляется с браузером *Chromium*, который включает плагин Flash; кроме того, в дистрибутиве предусмотрен плейер *VLC*, для работы с различными мультимедиа-файлами.

Загрузка и выключение WattOS идут значительно быстрее, чем у обычного дистрибутива Ubuntu. Он использует менеджер пакетов *Synaptic* и настроен с помощью собственного PPA, но можно также установить пакеты из репозитория Ubuntu.

Вердикт: Самый легковесный дистрибутив на основе *LXDE*.



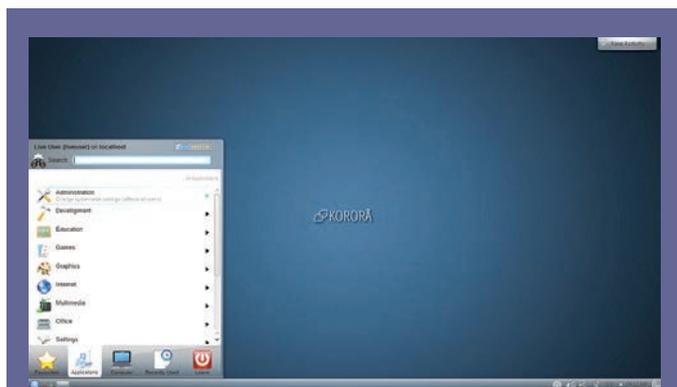
Manjaro

Этот дистрибутив с возобновляемыми релизами основан на Arch и предназначен для пользователей настольных ПК. В нем есть плагины, кодеки и приложения для работы с разными видами файлов.

Дистрибутив также поддерживает собственные репозитории, в которые входят и тщательно проверенные стабильные программы, и самые свежие, но нестабильные версии. Установка программ производится индивидуальным инструментом *Pamac* — это графический ин-

терфейс для *Pacman*, менеджера пакетов Arch. С помощью *Pacman* также можно установить в Manjaro программы из поддерживаемого сообществом Arch пользовательского репозитория. Manjaro официально поддерживает рабочие столы *Xfce*, *Cinnamon* и *Openbox* и имеет две текстовых программы установки; более новая поддерживает устройства Secure Boot и RAID.

Вердикт: Manjaro для Arch — то же, что Salix для Slackware.



Kororaa

Был создан на базе Gentoo; сейчас базируется на Fedora. Есть отдельные издания для Gnome (с инструментом расширений Gnome) и KDE (с подчищенным меню). Полностью поддерживает мультимедиа и активирует сторонние репозитории, например, RPMFusion, *Google Chrome* и *VirtualBox*. Новые рели-

зы дистрибутива идут с клиентом Steam. Есть пара инструментов пользователя, таких, как удобная утилита *undistract-me*, которая уведомляет, что команда терминала выполнена.

Вердикт: Самый дружелюбный к пользователю дистрибутив Fedora.



Kanotix

Основан на KDE и самом свежем стабильном релизе Debian, использует менеджер пакетов *Synaptic* и идет с мультимедиа-кодеками и плагинами. Кроме обычного набора стабильных (правда, старых) программ, Kanotix предоставляет *Wine* для работы с приложениями Windows. На момент написания данного материала разработчики выпустили специальную версию для выставки CeBIT; в отличие от постоянного релиза, она об-

новляет часть приложений до более свежих версий и включает несколько новых приложений. Здесь более новое ядро 3.8.2, более новые драйверы для видеокарт Nvidia, ATI и Intel, поддержка Secure Boot и многое другое. Предустановленный клиент Steam работает, даже если установить дистрибутив на USB-диск, и можно носить свои игры с собой.

Дистрибутив использует утилиты *acritox*, который передает задачу разбиения диска одному из трех соответствующих инструментов (*Gparted*, *fdisk* или *cfdisk*).

Вердикт: Стабильный дистрибутив для повседневной работы на базе KDE и Debian, с такими удобствами, как мультимедиа-кодеки и драйверы.



Netrunner

Главный сюрприз в том, что этот дистрибутив занимает такую низкую позицию в таблице популярности DistroWatch. Если вы о нем еще не слышали, считайте, что Netrunner — это Ubuntu без Unity.

Netrunner основан на Kubuntu, но это отнюдь не всего лишь очередной дистрибутив на базе Ubuntu, взявший KDE вместо Unity. Дистрибутив предлагает собственный облачный сервис под названием Runners-ID (созданный на сервере с открытым кодом ownCloud). Этот сервис интегрирован в дистрибутив и предлагает 5 ГБ хранилища, которое можно использовать для хранения данных, изображений, контактов, календарей и потоковой музыки через его приложение Android. Знакомо звучит, да?

И это еще не все. Дистрибутив также интегрировал в свой рабочий стол web-приложения — такие, как Cut The Rope,

Google Docs, Facebook и многие другие. Когда вы входите в свою учетную запись онлайн, ваши данные становятся доступными для таких локальных приложений, как *Contact* и *Dolphin*.

Одно из интересных web-приложений, имеющихся в дистрибутиве — JacknJoe. Это сетевой магазин приложений, где размещаются все популярные приложения и игры с открытым кодом, которые можно установить одним щелчком. Также в дистрибутиве имеются такие проприетарные приложения, как *Skype*, *Wine* для работы с программами Windows, и множество удобных аудио- и видеокодеков и плагинов.

Справочная информация о дистрибутиве рассредоточена по онлайн-форумам на его сайте. Здесь вы также найдете видео по уникальным приложениям дистрибутива и скринкасты по наиболее частым задачам. Разработчики дистрибутива



также публикуют онлайн-журнал под названием Netrunner-Mag. Дистрибутив поддерживается немецкой Blue Systems, в платежной ведомости которой упоминаются несколько разработчиков KDE.

Вердикт: Ubuntu-подобные функции в основанном на Kubuntu дистрибутиве.



ZevenOS

Проект ZevenOS выпустил два дистрибутива, и оба пытаются имитировать внешний вид и способ работы BeOS. Основной дистрибутив, именуемый ZevenOS, основан на Xubuntu и использует *Xfce*. Он идет со всеми мультимедиа-кодеками и плагинами и предназначен для систем, которым не хватает ресурсов для работы с самыми свежими дистрибутивами рабочего стола Linux. Далее идет дистрибутив ZevenOS Neptune, который главным образом осно-

ван на тестовой ветке Debian, но использует более новое ядро и содержит ряд дополнительных драйверов. Он использует рабочий стол KDE 4, опять же стилизованный под пользовательский интерфейс BeOS. Дистрибутив предназначен для установки на такие портативные носители, как USB-брелки. На его сайте немало видео по обоим дистрибутивам.

Вердикт: Ubuntu для поклонников BeOS.



Легковесы

Небольшие пакеты, идеальные для старого оборудования.

Zenwalk

Мини-дистрибутив на базе Slackware с массой разновидностей. В стандартном издании — рабочий стол *Xfce*, а другие предлагают Gnome и *Openbox*. Его цель — донести до старого оборудования

новые приложения. Совместим с двоичными пакетами Slackware, но применяет свой менеджер пакетов *Netpkg*, вводя возможность разрешения зависимостей в пакеты TGZ. Удивительно, но в нем есть полнофункциональные приложения вроде *Firefox* и *LibreOffice*. Zenwalk легко превратить в тонко настраиваемый LAMP или сервер распределенного доступа к файлам.

Вердикт: Быстр, с новыми приложениями и шармом старой школы.



Slitaz

Использует помесь *LXDE* с менеджером окон *OpenBox*, и создан для работы на оборудовании всего лишь с 256 МБ ОЗУ. Здесь много пользовательских инструментов, типа *SlitazPanel*. Это удобная па-

нель все-в-одном, позволяющая администрировать все аспекты работы. Весит Slitaz менее 30 МБ и занимает всего 80 МБ на жестком диске. Здесь нет офисного пакета и кодеков, но их можно установить из репозитория. Снабжен пользовательской документацией, к которой стоит обратиться до использования системы.

Вердикт: Придется кое-чему научиться, но это отличный дистрибутив-легковес для продвинутых пользователей.





Fuduntu

Изначально разработан для переделки обычного релиза Fedora в дружелюбную версию, а название свое заработал из-за стремления занять место где-то посреди Fedora и Ubuntu. Включает функции современных дистрибутивов, поддерживая внешний вид и способ работы традиционного рабочего стола. Это один из нескольких дистрибутивов, все еще поставляемых с рабочим столом Gnome 2. В соответствии с названием, содержит инструмент распознавания оборудования Jockey от Ubuntu, который

также сможет скачать проприетарные драйверы, ради максимума производительности. Fuduntu не боится проприетарных программ; в нем имеются клиенты Steam и Netflix, запускаемые через Wine. Есть также простые в использовании инструменты управления пакетами и настройки, а устанавливается он более старым инсталлятором Fedora — *Anaconda*, что весьма неплохо.

Вердикт: Странная смесь традиционного и нового; хорошо работает на маломощных компьютерах.



Slax

Популярный дистрибутив на базе Slackware недавно воспрял от четырехлетнего сна. Он предлагает рабочий стол KDE и практикует модульный подход: чтобы добавить программу, нужно взять модуль из *Slax Software Center*, который их предлагает всего несколько штук, например, *AbiWord*, *Gnumeric* и *Google Chrome*. Несмотря на малый размер, он предлагает все ко-

деки, плагины и приложения для повседневной работы, но без офисного пакета. Здесь нет программы установки: он предназначен для работы с портативного носителя, типа USB или CD. Определив устройство с перезаписью, он автоматически сохраняет изменения на нем.

Вердикт: Дизайн похож на Porteus, но маловато приложений.



Sparky Linux

Вам нужен дистрибутив сразу и для нового, и для старого компьютера? SparkyLinux как раз для вас. Он разработан как для новых, так и для старых машин, и в нем есть два настраиваемых рабочих стола (*Enlightenment* и *LXDE*) в основном издании и *Openbox* в издании Ultra.

Его индивидуальная программа установки обращается к различным инструментам для настройки раз-

ных аспектов системы, например, к *debconf* — утилите Debian для настройки клавиатуры, и к *Gparted* для разбиения дисков на разделы. Он содержит как полнофункциональные программы, как, например, *VLC*, так и легковесные, как *AbiWord*.

Вердикт: Дистрибутив-легковес, который будет очень ровно работать на полуустаревших компьютерах.



Porteus

Дистрибутив на базе Slackware, родом из Ирландии; разработан для портативных устройств, типа диски USB или CD, но его можно установить и на жесткий диск. Уникален тем, что существует в архивированном виде и создает свою файловую систему на лету. Помимо предустановленных при-

ложений, дополнительные программы для этого дистрибутива идут в виде модулей. Просто дважды щелкните мышью, чтобы активировать или деактивировать модуль. Дистрибутив содержит менеджер пакетов для получения модулей, которые можно также сохранить на жесткий диск, чтобы они пережили перезагрузку. Благодаря своей мо-

дальной природе дистрибутив невелик по размеру и невероятно быстр, поскольку работает из ОЗУ. На его сайте имеется обширная документация по использованию.

Вердикт: Идеален для установки на портативный носитель, хотя кривая обучения в нем довольно крута.



Puppy

Puppy непревзойден в плане готовых к употреблению функций. В нем множество пользовательских приложений, приложений для блокировки рекламы на сайте и интернет-телефонии, программа для скачивания подкастов, безопасная программа скачивания, аудиоплеер и многое другое.

Дистрибутив не содержит плагина Flash, но предлагает скачать и установить его, когда вы заходите на сайт, работающий на Flash — например, на YouTube. Можно также установить плагин из менеджера пакетов дистрибутива. В Puppy есть несколько мультимедиа-плееров, в том числе *Mplayer*, для воспроизведения всех медиа-форматов.

Начинающие пользователи от программы установки Puppy могут оробеть. Здесь нет автоматического разбиения диска на разделы — запускается *Gparted* для форматирования диска. Однако каждый шаг установщика хорошо докумен-

тирован в нем самом. Загрузка и запуск приложений отличаются редкой быстротой, даже для таких массивных приложений, как *Mplayer* и *VLC*, которые можно установить через Puppy Package Manager. Пакеты именуются *pet* и имеют расширение *.pet*. Можно установить пакеты для Puppy с помощью его пользовательского инструмента Puppy Package Manager, а можно настроить его на скачивание пакетов из других репозитивов Puppy.

У него есть два независимых и очень активных форума и бездна документации. Также в нем есть подкастка по нескольким темам, например, по работе с файлами *Microsoft Office*, добавлению кодеков и программ и т. д.

Есть несколько вариантов Puppy Linux. *WaryPuppy* — один из самых подходящих для старого оборудования, и в нем всего 130 МБ. Он предлагает два X-сервера — *Xorg* и *Xvesa*, перекрывающих широкий диапазон графических устройств. Есть еще вариант *RacyPuppy*, предназна-



ченный для более нового оборудования, и *SlackoPuppy*, на базе Slackware, плюс *PrecisePuppy*, на базе последнего релиза Ubuntu LTS.

Вердикт: Действительно лучший дистрибутив из предназначенных для оборудования, золотые времена которого остались позади.





Bodhi Linux

Элегантный рабочий стол с приятным и легким менеджером окон *Enlightenment*. Bodhi, вероятно, является дистрибутивом с наилучшей интеграцией *Enlightenment*. В нем есть ряд профилей, от *Vare* и *Laptop* до *Compositing* и *Fancy*, и каждый из них оптимизирован для разных типов оборудования.

Приложений в Bodhi мало. Вы можете добавить их с помощью новаторского web-инструмента для установки программ под названием *AppCenter*.

С помощью этого инструмента вы также сможете скачать пакеты любого дистрибутива и затем перенести их в Bodhi для установки.

Поскольку Bodhi основан на Ubuntu, он использует простую в использовании программу установки Ubuntu, и на своем сайте предлагает множество пользовательской документации.

Вердикт: Минималистский, простой в использовании дистрибутив, не утомляющий глаз.



aLinux

Один из способов создать дистрибутив для старого «железа» — взять более старые программы, меньше нагружающие скудные ресурсы. aLinux использует KDE 3.5 со старой версией *Koffice*, но воспроизводит все виды мультимедиа, имеет все игры и *Wine* для работы

с приложениями Windows; даже содержит проприетарные программы вроде *Skype* и *Google Earth*. Установка проста, благодаря несложной навигации.

Вердикт: Смесь старого и нового; но ладит не со всеми старыми машинами.



CrunchBang Linux

Дистрибутив на базе Debian с легковесным менеджером окон *Openbox*. В нем есть *Gnome Mplayer*, *GIMP*, *VLC*, и его web-браузер снабжен плагином *Flash*. Дистрибутив включает *AbiWord* и *Gnumeric*, а также скрипты для установки *LibreOffice* и *Dropbox* в его меню. Он использует

менеджер пакетов *Synaptic* и модифицированную программу установки Debian.

Вердикт: Энергичный дистрибутив на базе Debian; хорошо работает на старых машинах и может быть отложен и для относительно новых.



DSL

Если вы сочли *Puppy* эзотеричным, какой же тогда DSL? Это популярный дистрибутив тоже недавно пробудился от спячки. Он использует менеджер окон *JWM* и после загрузки запускает руководство *Getting Started*, чтобы сориентировать пользовате-

лей. Объем его 50 МБ; хотя основные приложения здесь имеются, некоторые, подобно *Firefox*, установили так, что сайты вроде YouTube грузиться не будут. Можно добавить другие приложения с помощью системы *MyDSL*. DSL задуман как кочевой дистрибутив, но в нем все же есть мастер установ-

ки, основанный на *knxhinstall*, который скопирует содержимое на раздел в 300 МБ.

Вердикт: Процесс обучения довольно сложен; рекомендуется только при крайней необходимости вдохнуть жизнь в старое оборудование.



OantiX

Цель этого легковесного дистрибутива — обеспечить полнофункциональный дружелюбный к пользователю настольный ПК. Он основан на тестовом репозитории *Debian Testing*, и ядро его относительно новое по сравнению с другими дистрибутивами для тех же целей. Идет с менеджером окон *Fluxbox* и воспроизводит все виды мультимедиа. Основные функции рабочего стола управляются такими пользовательскими

инструментами, как *antiX Control Center*. Также в нем есть свой менеджер пакетов и своя программа установки — довольно простая и хорошо документированная. Есть также инструмент для создания устанавливаемого в режиме *live* образа системы.

Вердикт: Хороший быстрый дистрибутив для старых машин и тех пользователей, которых *Puppy Linux* озадачивает.



Tiny Core Linux

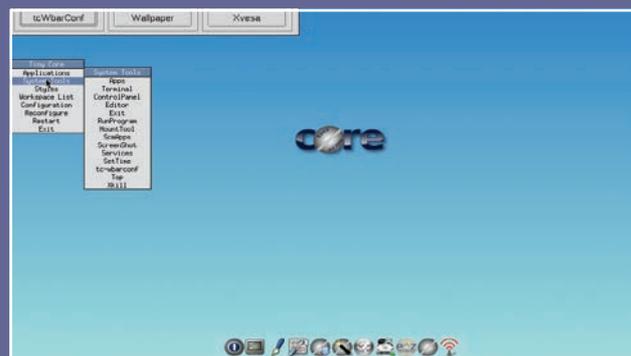
При размере всего 12 МБ, он снабжен только терминалом, текстовым редактором и программой запуска приложений на основе легковесного менеджера окон *FLWM*.

В нем есть панель управления для управления сервисами загрузки и настройки программы запуска приложений, но все остальное нужно устанавливать через менеджер пакетов, в том числе и программу установки, если вы хотите установить *Tiny Core* на жесткий диск. Дистрибутив имеет разновидность *CorePlus*,

с дополнительными драйверами для беспроводных карт, инструментом ремастеринга и поддержкой интернационализации.

И, наконец, есть вариант *Core edition* размером 8 МБ — это просто базовая система с интерфейсом командной строки, позволяющая более опытным пользователям создать с нуля собственную систему.

Вердикт: Будет работать даже на самом старом оборудовании, но настройка потребует времени.



Друзья новичков

Предназначены для снижения культурного шока от перехода на другую ОС.

OS4

Еще один дистрибутив на базе Ubuntu и Xfce для неопытных пользователей Linux. Единственное, что выделяет OS4 среди ему подобных — уникальная схема рабочего стола.

Разработчики OS4 заявляют о поддержке устройств, которые пока что не поддерживаются аж самим Ubuntu: например, устройств на WebOS и на Nook... и даже Kindle Fire.

Дистрибутив включает поддержку популярных плагинов браузера, множество приложений для воспроизведения и создания мультимедиа, и даже инструменты для создания программ. На сайте дистрибутива нет форумов и документации, но вы можете оплатить поддержку в онлайн-магазине, где также можно приобрести настольные компьютеры и ноутбуки с предустановленным OS4.

Вердикт: Готовый к работе дистрибутив с уникальным интерфейсом, но беден документацией.



Ultimate Edition

Этот дистрибутив воспринял свое имя всерьез. Он использует перенастроенный рабочий стол KDE с пользовательскими значками и указателями. Но все это заходит чересчур далеко — тема белого шрифта на черном фоне распространяется на LibreOffice и как-то не способствует приятности работы. И здесь по несколько приложений на одну задачу, что тоже сомнительное удовольствие. Хотя набор плееров мультимедиа — вещь в общем-то безобидная, наличие нескольких программ установки приложений, таких, как Ubuntu Software Center, Lubuntu Software Center и Muon Package Manager, сбивает с толку. Потратив уйму времени на настройку разных аспектов рабочего стола, разработчики нимало не изменили программу установки Kubuntu 12.04 — она даже и название показывает "Kubuntu".

Вердикт: Всесторонний дистрибутив в духе Knoppix, способный запутать мириадами возможностей выбора.



LinuxLite

Дистрибутив основан на Ubuntu 12.04 LTS, и его рабочий стол Xfce заботливо отделан. В LinuxLite имеются обычные приложения — LibreOffice, Firefox, GIMP и VLC, но поскольку он основан на релизе LTS, многие приложения устарели. Пользователи рабочего стола оценят наличие плагина Flash и кодеков, а также появление в последнем релизе клиента Steam.

Вердикт: Еще один принаряженный дистрибутив на базе Ubuntu, который способен предложить не слишком много нового.



Pear Linux

Этот дистрибутив из Франции на базе Ubuntu имеет свой интерфейс пользователя, который покажется привычным приверженцам Mac. Gnome Shell настроена здесь таким образом, чтобы напоминать OS X, и даже есть dockbar в стиле Mac OS X. Есть горячие точки для отображения настраиваемого обзора и для переключения виртуальных рабочих столов. Некоторые инструменты, например, панель и программа запуска приложений, заимствованы из проекта Elementary OS.

Имеется готовая поддержка популярных мультимедиа-кодеков, и на рабочем столе есть значок программы Launch Me First, которая выполняет некоторые задачи постинсталляции: например, обновляет репозитории и настраивает Pear Appstore, весьма похожий на Ubuntu Software Center.

Вердикт: Чудесный дистрибутив, облегчающий переход на Linux пользователям OS X.



Zorin

Благодаря Gnome, настроенному на сходство с рабочим столом Windows 7, будет удобен пользователям Windows. Имеется в нескольких вариантах: Core edition бесплатный, но другие (Multimedia, Gaming, Business и Ultimate edition) стоят €7–18. Приложение для «мимикрии» позволяет выбрать, какую проприетарную ОС надо имитировать, а Wine поработает с программами Windows. Основан на Ubuntu и использует инсталлятор этого дистрибутива.

Вердикт: Ubuntu, который заставили выглядеть, как Windows 7... если это ваш путь.



SolusOS

Разработанный тем же человеком, что трудился над Debian-версией Linux Mint, SolusOS построен на Debian Stable, но с более новыми программами. У него отглаженный и отлаженный рабочий стол Gnome с украшениями окон, нижней панелью и программой запуска приложений в стиле Windows 7. Также он содержит офисные настольные приложения, плюс Wine, PlayOnLinux и Minitube для просмотра видео с YouTube.

Gnome System Settings тщательно организованы в SolusOS Control Center. После установки дистрибутива вас приветствует пятишаговый мастер первого запуска, который сканирует оборудование и устанавливает проприетарные драйверы, и настраивает брандмауэр. С помощью Synaptic Package Manager можно установить приложения из репозитория обратных портов SolusOS или Debian.

Вердикт: Стабильный дистрибутив рабочего стола, выглядит и работает отлично.



Parsix

Одна из прелестей Parsix — приятное оформление и темы рабочего стола. Дружелюбный к новичкам дистрибутив основан на Debian Testing и предлагает самый свежий стабильный релиз Gnome. Основные версии выходят каждые полгода. Содержит кодеки и плагины, а также плеер VLC для работы со всеми видами мультимедиа. Простая программа установки честно выполняет свою работу. Пакетами управляет PackageKit от Gnome. Parsix следует Debian Security Advisories [Советы по безопасности Debian], и тут же предоставляет обновления безопасности. Можно устанавливать проприетарные приложения из его официальных репозиториях, которые также содержат пакеты из Debian Testing.

Вердикт: Дистрибутив на базе Debian с фиксированным циклом релизов, битком набитый приложениями.



Продвинутым

Для знатоков Linux, которым нужен полный контроль над их системой.

Debian

По-прежнему считается лучшим дистрибутивом для серверов, но проник и на настольные ПК. Очень гибок: настраивается и как настольный, и как сетевой/почтовый/файловый сервер. Благодаря не самым новым, зато тщательно проверенным стабильным пакетам, исключительно безопасен и ладит с самыми разными архитектурами. Идет без проприетарных драйверов и ко-

деков, но то, что он является одним из самых старых и популярных дистрибутивов Linux, имеет свои преимущества. Почти все производители программ предоставляют готовые бинарники для Debian, поэтому установка программ — сущие пустяки.

Вердикт: Обеспечивает идеально стабильную систему, подходящую для серверов.



CentOS

Происходит от RHEL и является любимцем тех администраторов, которые предпочитают систему управления пакетами RPM; предоставляет вам все, чего только можно ожидать в дистрибутиве класса предприятия. Идеально подходит для настройки серверов, потому что в нем есть *Apache*, *MySQL*, *PHP*, *Perl* и разное серверное ПО; но в нем мало подходящего для повседневного

использования, или для любителей, или для продвинутых пользователей Linux. Для сравнения, его главный конкурент, Debian, является дистрибутивом общего пользования, который в равной мере хорош на домашнем ПК или для обслуживания web-страниц.

Вердикт: Почти вытеснен альтернативами, такими как Scientific Linux.



Slackware

Предлагая пакеты, почти идентичные своим фирменным релизам, Slackware стремится предоставить наиболее близкий к Unix дистрибутив Linux. Дистрибутив предоставляет пользователю огромные возможности управления во время установки, позволяя решить, какие устанавливать пакеты и библиотеки. Будучи одним из старейших дистрибутивов Linux, Slackware исключительно стабилен и годится для серверов. В нем нет графических инст-

рументов администрирования системы, и пакеты управляются из командной строки, хотя есть основанные на *ncurses* инструменты с ограниченной функциональностью для выполнения определенных задач. Навык работы в командной строке превратит работу в нем в истинное наслаждение.

Вердикт: Стабильный, надежный, классический. Отлично подходит для серверов и опытных пользователей.



Продвинутые легковесы

GnuX

Базируется на исходнике; имеется только для архитектуры x86-64 и использует систему портов для управления пакетами, компилируясь по формуле KISS [есть суть — «будьте проще», — прим. пер.], и предпочитая простоту в автоматизации. Неграфический инсталлятор включает компиляцию ядра в установку, как в Gentoo.

Вердикт: Чудесный дистрибутив для опытных пользователей, с поддержкой только 64-битной архитектуры.



Gentoo

Здесь пользователи получают всесторонний контроль над созданием системы с самого нуля. Вместе с Arch, Gentoo является одним из самых настраиваемых дистрибутивов, и он рассчитывает на то, что вы скомпилируете ядро, настроив его на свой вкус во время установки. Дистрибутив упаковывает потрясающую систему управления пакетами в Portage. В отличие от большинства других дистрибутивов, установка Gentoo может занять от нескольких часов до нескольких дней, в зависимости от количества желаемых пакетов, поскольку те скачиваются из Интернет. Если вы никогда не пользовались им прежде, вам придется поучиться, и вы познакомитесь с начинкой Linux и несколькими новыми технологиями, родными для Gentoo, например, с системой флагов USE. Производные типа Funtoo могут стать неплохим стартовым пунктом для тех, кто не знаком или не желает связываться со сложностями Gentoo.

Вердикт: Проигрывает Arch из-за долгой установки и ряда незнакомых концепций, как, например, флажки USE.



Arch

Arch — лидер прошлого Сравнения дистрибутивов для умников: те его сильно любят за высокую степень настраиваемости. Имеет скользящий цикл релизов, но не предоставляет пакетов по умолчанию. Установка может затянуться — это зависит от желаемых пакетов: они загружаются из Интернета. Фишка Arch, обеспечивающая ему высокие рейтинги — инструмент управления пакетами *Pacman*. Установка познакомит вас с рядом файлов настройки, которые надо задать вручную. Вручную делается все, от разбиения диска на разделы и до установки программы загрузки. Arch породил несколько дистрибутивов, каждый со своими отличиями. Испытайте Cinnarch, если вам нужна вся мощь Arch в сочетании с Cinnamon.

Вердикт: Настройка быстрее, чем у Gentoo, и больше радостей для технарей, чем в любом другом дистрибутиве.



```

root@arch:~# pacman -S midori
[+] Retrieving packages from mirrors...
[+] Resolving dependencies...
[+] Looking for missing conflicts...
Targets (9): archlinux-0.80.6-1-i686  archlinux-1.6.0-4-gcc-3.6.2-2  gentoo-0.12.0-5
libmbedtls-0.9.15-1  libmbedtls-1.3.2-2  libmbedtls-0.9.15-3
webkitgtk2-1.10.2-2  midori-0.4.8-1
Total Download Size: 10.39 MiB
Total Installed Size: 54.54 MiB

Proceed with installation? [Y/n]
:: Retrieving packages from mirrors...
libmbedtls-0.9.15-1... 123.0 KiB 58.3K/s 00:02 [#####] 100%
archlinux-0.80.6-1-i686... 585.5 KiB 144K/s 00:04 [#####] 100%
midori-0.4.8-1... 203K/s 00:01 [#####] 100%
libmbedtls-1.3.2-2-i686... 442.0 KiB 139K/s 00:05 [#####] 100%
archlinux-1.6.0-4-i686... 35.0 KiB 318K/s 00:00 [#####] 100%
gentoo-0.12.0-5-i686... 92.2 KiB 26.3K/s 00:03 [#####] 100%
webkitgtk2-1.10.2-2... 5.6 MiB 313K/s 00:05 [#####] 77%
    
```

Frugalware

Созданный на базе Slackware, Frugalware верен прошлому. Предназначен для средних и продвинутых пользователей, и предлагает выбор между командной строкой или графической установкой. На DVD множество пакетов, и его легко настроить в качестве сервера.

Вердикт: Хороший дистрибутив, если вам нужны все функции Arch.



Foresight Linux

Продвинутых пользователей заинтересует в нем одно: система управления пакетами *Conary*. В отличие от аналогов, *Conary* обновляет только устаревшие файлы, экономя время и полосу; также может откатить или вернуть любое обновление одной командой.

Вердикт: Не предлагает продвинутому пользователю ничего особенного, кроме *Conary*.



LXF Games

Помимо дистрибутивов, добытых из Интернет, мы еще создали собственный. Поскольку мы проводим много времени за играми, он у нас игровой. В нем более сотни игр, которые мы смогли уместить на DVD, так что загрузите его — и играйте во что хотите, от нардов до пинбола. **LXF**



Резоны верить в ZaReason

LXF беседует с Кэти Малмроуз, руководителем ZaReason, о создании специального оборудования под Linux.



Мы любим Linux. Эта прекрасная система дает нам вволю поэкспериментировать, и она свободна. И не только от затрат, но и от ограничений коммерческого ПО. Ложкой дегтя могут стать лишь проблемы с готовностью оборудования к работе — в итоге вы рискуете погрязнуть в спецификациях, проверяя совместимость драйверов в разных версиях ядра, не говоря уже о налоге Microsoft.

Однако все может быть иначе. Существуют замечательные компании, производящие ПК специально под свободное ПО, на которых Linux уже установлен и готов к работе. ZaReason — одна из таких компаний, и нам удалось поговорить с ее руководителем, Кэти Малмроуз [Cathy Malmrose], сразу после ее доклада на FOSDEM об опасностях UEFI.

LXF: Как лично у вас возник интерес к свободному ПО, а затем и к оборудованию, на котором оно работает?

КМ: Для меня все началось в 1998 году. У нас двое сыновей — один рыжий, другой блондин — они настояли, чтобы дома было девять компьютеров, чтобы они могли играть с друзьями после школы. Мы согласились, и каждые выходные спасу

не было от «голубых экранов смерти». Девять машин и куча детишек, играющих на них — постоянно приходится работать техподдержкой. Нужно было нечто более устойчивое.

О СОЗДАНИИ ZAREAZON

«Мы не нашли машин, на которых Linux запускался сразу.»

LXF: Не могли бы вы рассказать немного об истории ZaReason? Когда стартовал проект? Чего вы хотели добиться?

КМ: Конечно. Все началось в 2007 году в Беркли, Калифорния. И тому было две причины: мы не нашли машин, на которых Linux запускался сразу из коробки, а мучиться с подбором правильных комплектующих нам не хотелось, и мы решили создать компанию, где мы бы сами их делали. А также — неподалеку находился завод по утилизации компьютерной техники, ACCRC [Alameda County Computer Resource Center], где было более 16 000 списанных машин на Ubuntu, и мы вызвались помочь им передать эти компьютеры в школы и другие организации, нуждающиеся

в бесплатной оргтехнике (а 16 000 машин — это очень много). Выяснилось, что Ubuntu способен запускаться буквально на чем угодно. Мы просто поразились запасу прочности этой ОС, вот так мы и решили создать свою компанию в 2007.

LXF: Каковы были главные препятствия продажам компьютеров с Linux?

КМ: Глобализация. Мы скорее технологи, чем предприниматели. Нам нужен новый глава компании, способный управиться с глобальным распространением, но не так-то просто найти того, кто будет делать все, что нужно, думая не только о прибыли. И кто сумеет укрепить наши позиции.

LXF: Мы слышали, вы перебираетесь в Великобританию?

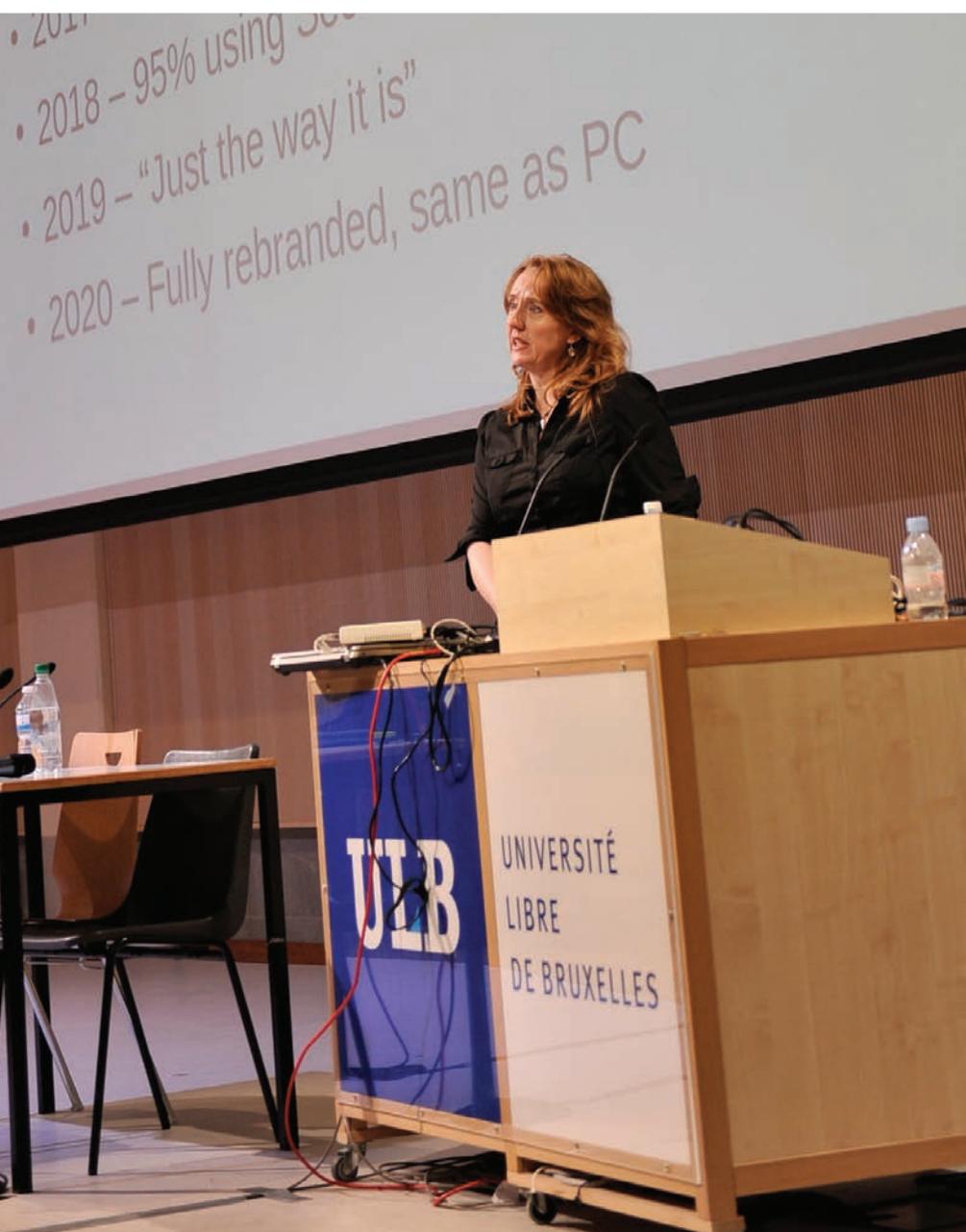
КМ: Да.

LXF: И в какие сроки?

КМ: Сразу после конференции!

LXF: Куда-то конкретно направляетесь?

КМ: Да, есть одно подходящее местечко. Нашим партнером будет существующий компьютерный



О СВОИХ ПОКУПАТЕЛЯХ

«Все, от лаборатории NASA до старичка, которого Windows довел до слез.»

магазин. У них уже налажено распространение, и организована потрясающая техническая поддержка.

LXF: А после Англии, где будет следующая остановка?

КМ: В Италии, Франции, Германии... да где угодно. Там будет видно.

LXF: Что, по-вашему, труднее: наладить производство компьютеров или найти на них покупателей?

была рабочей в любых условиях, у любого пользователя, при любой комплектации, что бы на нее ни установили, требуется уйма исследований. Многие наши клиенты решают сложные задачи.

LXF: Ваши покупатели — корпорации или частные клиенты?

КМ: Все, начиная от ракетной лаборатории NASA, до одного из наших первых клиентов, в прошлом одного из архитекторов Google, ныне владельца собственной лаборатории робототехники. И до старичка, которого Windows довел буквально

КМ: Самое трудное — это исследовательская работа. Когда я слышу от кого-нибудь, что мы просто предустанавливаем программы, меня трясет, ведь это далеко не так. Чтобы система

до слез. Мне нравится помогать конкретным людям. А также научным лабораториям. Мы оказали немалую поддержку лаборатории LBL [Lawrence Berkeley National Laboratory], которая существует на средства министерства энергетики.

LXF: Вчера в своем выступлении Амелия Андерсдоттер [Amelia Andersdotter] рассказывала о технологиях Open Source в европейских правительствах. По ее словам, комиссия по конкуренции была не в состоянии расследовать монополию использования Microsoft в данной сфере потому, что не было поставщика, готового представить открытую альтернативу в таких масштабах.

Думаете, это шанс для вас?

КМ: Если мы начнем в Евросоюзе, то будем это иметь в виду. А вот в Штатах все осложнено. Политкорректный процент контрактов должен доставаться компаниям, управляемым женщинами, и по этой причине женщин на такие должности нередко нанимают только для видимости, но не для работы, а для дела такая среда малопривлекательна.

LXF: Что бы вы хотели рассказать нашим читателям о ZaReason?

СМ: В это кризисное время мы выживаем только благодаря тому, что наши исходные клиенты были разработчиками. Знаете, если бы в 2007, когда еще были проблемы со спящим и ждущим режимом, не-технарей у нас было больше, их вряд ли утешило бы то, что над их решением работает сообщество. Но тогда именно поддержка технарей помогла нам справиться. Они получали эти машины, на грани износа, причем зная об этом, а через неделю мы получали письмо: «Привет, знаете, как делается то-то?» Или «Привет, вот это можно исправить!» Поразительно! Один парень прислал нам сравнение цен на наши и другие модели, чтобы мы шли в ногу с конкурентами. У нас была огромная поддержка, и я хочу сказать, что мы очень всем благодарны. Вот что значит сообщество, и пусть так будет и впредь.

LXF: Вы только что рассказывали о UEFI.

А как производители оборудования относятся к альтернативам?

КМ: Производители оборудования? Да никак.

LXF: Все по-прежнему на стороне Microsoft?

СМ: Это бизнес. Это так непохоже на сообщество свободного кода — совершенно иной мир. Тут нет доброжелательности. Нет убеждения в правоте своего дела. Только бизнес. Но жить так не очень хочется. Именно поэтому я ценю такие конференции [как FOSDEM] — они возвращают мне веру в человечество.

LXF: Каков, по-вашему, лучший способ уйти от Secure Boot?

СМ: Черт возьми, да их полно! Для ZaReason идеальным решением стало наладить сотрудничество с как можно большим чистым дистрибутивом, предоставив им свободу действий и взяв на себя всю грязную работу. С дистрибутивами тяжело иметь дело! Вот почему производителей оборудования



О ПРИНЦИПАХ MICROSOFT

«Это непохоже на сообщество свободного кода — совершенно иной мир.»

так мало. Между свободным, открытым ПО и бизнесом есть некоторые расхождения, но нам удается найти рабочее решение.

LXF: Вы тестируете огромное количество техники, материнские платы и прочее. На многих обнаруживается сломанный UEFI?

СМ: Пока такой проблемы нет. Как в случае с планшетами Win8 RT, где, по словам Microsoft, его можно отключить. Пока все так и есть. Но, по логике, однажды они попытаются ввести его и на настольных ПК. И это не догадки, это

КМ: Не покупать машин с Windows 8! Обратиться к поставщику Linux. Их много. Беда отчасти в том, что наши разработчики распляют свои умственные способности на то, чтобы решать проблемы на каждой машине в отдельности. Если же подходить к этому более последовательно, рассчитывая на долгосрочную перспективу, получилось бы гораздо действеннее. Производителям нужно объединяться. Мы надеемся вскоре провести такую встречу. **LXF**

прогноз — таков принцип работы корпораций. В этом их цель.

LXF: Если кто-то захочет этому помешать, что вы посоветуете им делать?

Соперники

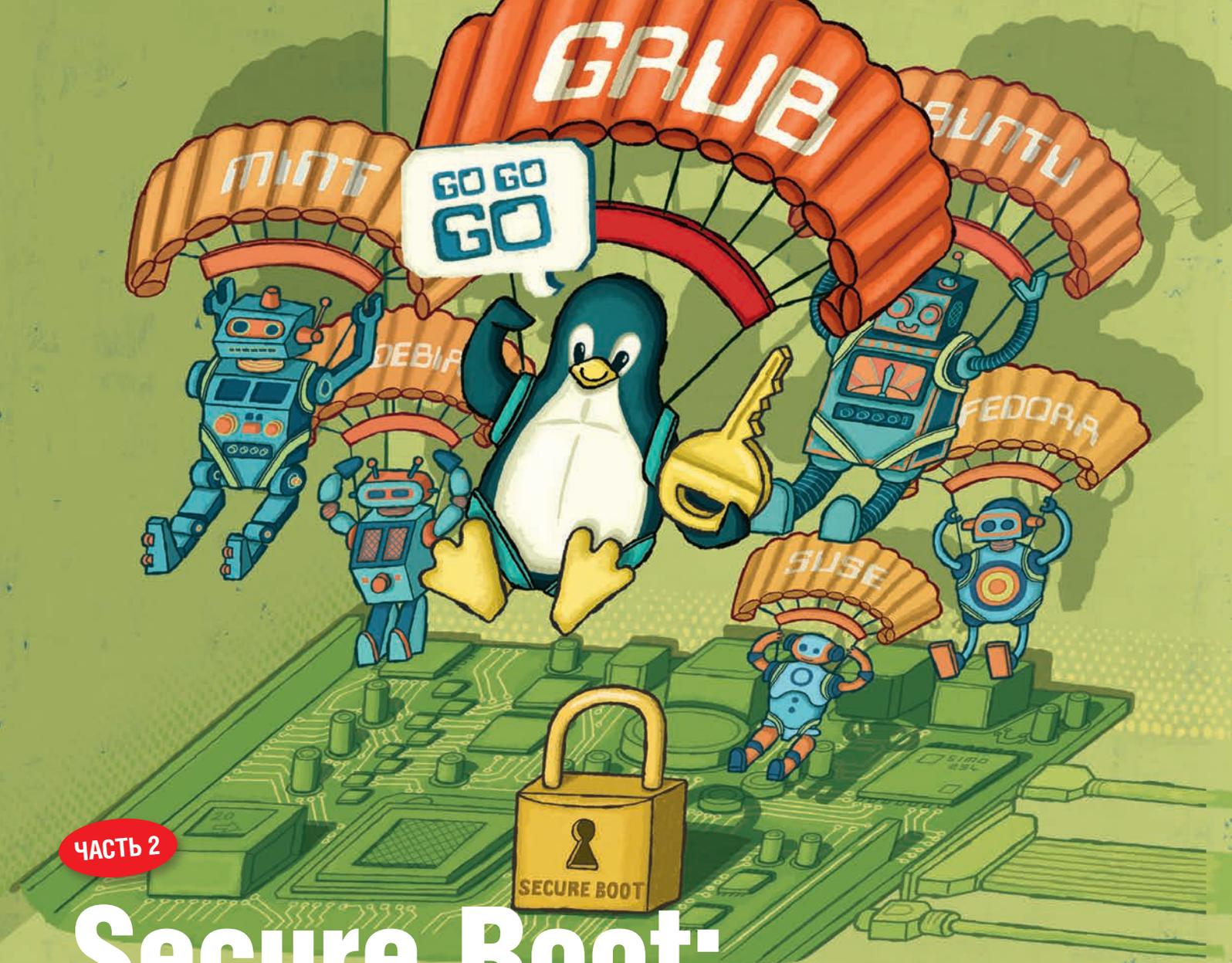
ZaReason делают прекрасные компьютеры, но они не одиноки. Развивающиеся с переменным успехом отношения настольного Linux и Dell вновь пошли в гору, с появлением XPS 13 со специальной версией Ubuntu на борту.

Из прочих, самый известный, пожалуй, System76, производящий, как и ZaReason, широкий спектр ноутбуков, настольных систем и серверов. Они также базируются в США, но поставляют свою продукцию и в другие страны.

Ну, а в Англии родной есть свои фирмы, вроде Linux Emporium и Karma Tech.

На сайте Linuxpreloaded.com вы найдете массу информации о поставщиках компьютеров с Linux по всему миру.





ЧАСТЬ 2

Secure Boot: Вся правда

Бен Эверард рассматривает новейшую функцию родом из Редмонда, вызвавшую яростные споры.

Secure Boot, реализация UEFI от Microsoft, озаботила пользователей Linux с того самого момента, когда о ней сообщили в январе 2012 года. Мы уже ее рассматривали, но поскольку страсти пока не улеглись, мы решили снова обратить на нее внимание. Но прежде чем двигаться вперед, давайте вернемся к истоку и рассмотрим, что же такое Secure Boot.

До 2012 года все ПК загружались с помощью BIOS. Это был реликт из 1980-х, скверно реализованный многими производителями, что приводило к сляпанной наспех системе, которую удерживал вместе цифровой эквива-

лент скотча. Она работала, но исключительно благодаря упорному труду разработчиков, которые могли бы потратить свое время на куда более интересные вещи, чем загрузочный интерфейс, не будь они постоянно заняты попытками заставить все эти гадости работать как следует.

Поэтому, когда Microsoft начал говорить с разработчиками о возможности заменить

старую систему, все радостно запредвкушали назревшую кончину BIOS.

UEFI во многих отношениях является гораздо лучшей системой. Она дает программам загрузки больше пространства для работы и лучше поддерживает современное оборудование. Мы рассматривали ее более подробно в LXF168. Останься все в том виде, в каком было, большинство были бы счастливы.

Однако этого не случилось. Они начали что-то мастерить. Им пришлось шагнуть дальше и добавить Secure Boot. Это расширение UEFI, которое ограничивает загрузку только системами, имеющими шифрованную подпись с надежным

«До 2012 года ПК загружались с помощью BIOS, реликта 1980-х.»

ключом. И напоследок самое интересное: мастер-ключ контролирует Microsoft. Хитро.

Конечно, как всем нам понятно, это небольшое дополнение не сделает загрузку Linux в системах Secure Boot невозможной — просто слегка ее усложнит.

Риски безопасности

Конечно, очень легко заявить, что все это — проделки Microsoft ради усиления контроля над компьютерной индустрией, и именно этой точки зрения придерживается немало народу. Однако данная мера объясняется вполне реальным риском для системы безопасности Windows: вирусами загрузочного сектора. Это вид вредоносного ПО, которое заражает Master Boot Record (MBR), сидящую в самом начале диска. Когда пользователь запускает загрузку, BIOS запускает код в MBR, который, если все нормально и система работает правильно, затем загрузит операционную систему. Однако вирус загрузочного сектора может захватить этот процесс и запустить некое вредоносное действие до загрузки ОС. Системы с Secure Boot будут загружать только тот код, в котором есть подпись, поэтому в основном эти вирусы не будут в состоянии загрузиться.

«Secure Boot — переворот ради решения проблемы, которая отмерла бы сама.»

Звучит неплохо, правда? Однако во всем этом есть пара не совсем незначительных проблем. Во-первых, надо полагаться на единственный ключ, а так уж исторически сложилось, что единственный ключ не может долго оставаться в секрете (помните, как хранились секретные ключи PlayStation и Blu-ray?).

Во-вторых, и это, возможно, важнее, вирусы загрузочного сектора были распространённым явлением в дни MS-DOS. С тех пор мир несколько изменился, и создатели вирусов тоже не исключение. Secure Boot — колоссальный переворот,

совершенный ради решения проблемы, которая со временем просто отмерла бы сама по себе. Честно говоря, в Windows есть куда более наболевшие проблемы безопасности, которые совершенно не затрагиваются этим новшеством.

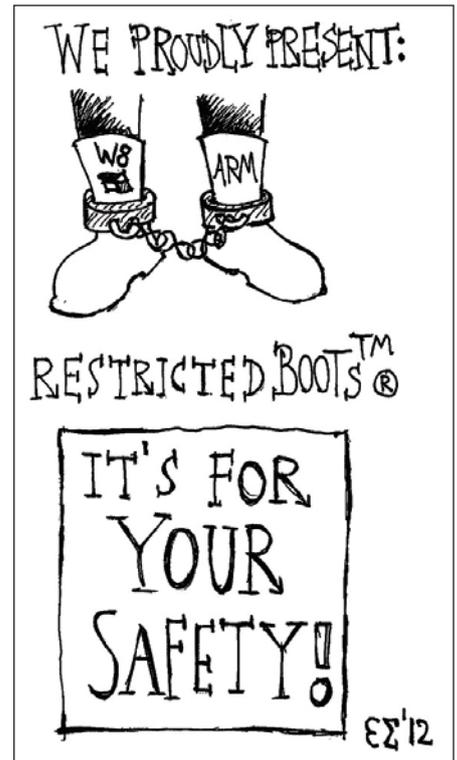
Говорит Microsoft

Итак, мир вредоносного ПО в Windows вряд ли постигнет такой уж удар. Нельзя сказать и того, что он особо эффективно сможет помешать другим системам работать на ПК. В Windows 8 Hardware Certification Guidelines [Указания по сертификации оборудования для Windows 8], Microsoft заявляет: «17. Обязательные положения. В системах, отличных от ARM, платформа ДОЛЖНА реализовывать возможность для физически присутствующего пользователя выбирать между двумя режимами Secure Boot в настройке оборудования: “Custom” и “Standard”. Режим Custom Mode позволяет большую гибкость в соответствии с нижеследующим:

а. Физически присутствующий пользователь должен быть в состоянии использовать опцию настройки оборудования Custom Mode, чтобы изменить содержимое базы данных подписи Secure Boot и ПК. Это может быть реализовано простым предоставлением опции очистки всех баз данных Secure Boot (PK, KEK, db, dbx), что переводит систему в режим setup.

18. Обязательные положения. Включение/Выключение Secure

Boot. В системах, отличных от ARM, требуется реализовать возможность отключения Secure Boot через настройки оборудования. Физически присутствующему пользователю должно быть разрешено отключать Secure Boot через настройки оборудования без владения PKpriv. Windows Server также может отключать Secure Boot дистанционно с помощью внеполосного управляющего соединения с сильной аутентификацией (желательно на основе публичного ключа), такого, как с контроллером материнской платы или сервисным процессором. Программное отключение Secure Boot



► Неудивительно, что группы по защите прав пользователей без восторга восприняли Secure Boot, считая его попыткой ограничить действия с оборудованием. Free Software Foundation обозвал этот стандарт Restricted Boot — «Скованными ботинками».

либо во время Boot Services, либо после выхода из EFI Boot Services должно быть невозможно. Отключение Secure Boot должно быть невозможно на системах ARM».

Все это означает, что (если только ваш компьютер не использует процессор ARM) сертифицированный компьютер с Windows 8 должен позволить вам как добавлять ключи для загрузки другой системы, так и отключать Secure Boot. Конечно, нет способа узнать, будут ли подобные же требования к сертификации в более поздних версиях Windows. И вот где ключевой момент: в своем нынешнем виде Secure Boot не может помешать Linux работать на компьютере, но толкает нас к такому положению вещей, когда доступ к определенному оборудованию на ПК будет закрыт производителем. Мы можем только гадать, куда заведет нас этот путь, но нетрудно предвидеть будущее, когда установка выбранной вами операционной системы на ваше же оборудование станет все более и более усложняться.

Понимание подписей

Чтобы полностью понять это, нам придется рассмотреть цифровые подписи. Если вы пишете письмо в старомодной манере, вы, вероятно, ставите в конце свою подпись. Это — ваша уникальная отметка: тогда читающий письмо поймет, что писали его именно вы. Это работает, потому что почерк у каждого свой, и совсем несложно создать некую пометку, которую вам легко запомнить и воспроизвести, а другим трудно скопировать. »

Ручное разбиение диска

Большинство программ установки Linux имеют две опции для настройки жесткого диска — автоматическую и ручную. Если выбрать первую в системе UEFI, она будет просто работать (при условии, конечно, что дистрибутив поддерживает UEFI). Однако если вы выберете вторую, то вам придется побеспокоиться о создании раздела EFI.

В Ubuntu 12.10, перейдите в окно Installation Type и выберите Something Else. Нажмите на клавишу +, чтобы добавить раздел, и задайте ему размер порядка 100 МБ. Измените выпадающее меню Use As на EFI Boot Partition. Затем делите оставшуюся часть диска по своему усмотрению, и все это должно заработать.

В openSUSE 12.3, оказавшись в Suggest Partitioning, нажмите на Create Partition Setup и выберите

Custom Partitioning. Создайте новый раздел размером около 100 МБ, а на следующем экране выберите файловую систему FAT и точку монтирования /boot/efi. Далее продолжайте по своему усмотрению; а понав на экран Live Installation Settings, выберите Booting, смените программу загрузки на Grub2-EFI и отметьте Enable Secure Boot Support.

В Fedora 18 перейдите на экран Installation Destination, выберите желаемые диски, затем нажмите на Next. Появится экран Installation Options. Отметьте I Don't Need Help; Let Me Customize Disk Partitioning, и перейдите на экран Manual Partitions. Здесь зайдите в New Fedora 18 Installation, и нажмите на +. Введите в графу Desired Capacity 100MB и точку монтирования /boot/efi, и затем продолжайте оставшуюся часть настройки.

Отключение UEFI

Требуется, чтобы любой компьютер, отличный от ARM, сертифицированный Windows 8, предусматривал возможность отключения UEFI. Это отличная опция: она означает, что будет работать любой дистрибутив. Процесс будет зависеть от конкретной реализации UEFI/BIOS, но не должен быть особо сложным.

На нашей тестовой машине мы перезагрузились и нажали на клавишу Delete, чтобы войти в setup. Затем перешли в Advanced Mode > Boot > CSM (Compatibility Support Module) и настроили Launch CSM на Enable и Boot Device control на UEFI and Legacy.

Настроенная таким образом система будет и дальше загружать дистрибутивы Linux традиционным способом. Настройка не из простых, но ненамного труднее, чем настройка BIOS на загрузку с оптического устройства, которую нам приходилось делать на наших старых компьютерах.

➤ Отключение UEFI — не самая простая задача, но в конечном итоге с ней стоит повозиться.



Секретный ключ

В цифровом мире подписи играют практически ту же роль (то есть удостоверяют, кто прислал сообщение), но реализованы совершенно иначе. Начинаются они с хэша (именуемого также необратимым шифром), который уменьшает цифровой объект (будь то текстовый документ, Zip-файл, ядро операционной системы или что-то еще) и превращает его в цифры. Восстановить по хэшу исходный документ невозможно (отсюда и название — необратимый шифр), а если вы измените начальный объект, то и хэш получится другой.

Во второй стадии процесса подписания участвует пара публичный/приватный ключ. Эти ключи несколько похожи на пароли, и их можно использовать для шифрования документов, но они работают таким образом, что все зашифрованное с помощью публичного ключа может быть дешиф-

ровано с помощью приватного, и наоборот. Публичный ключ можно вручить кому угодно, а приватный ключ должен храниться в секрете.

Для подписания какого-либо двоичного объекта вам нужно пройти обе эти стадии. Сначала вы зашифруете объект, а затем шифруете его с помощью приватного ключа, чтобы создать свою уникальную подпись. Эта подпись будет послана вместе с двоичным объектом. Затем получатель объекта и подписи дешифрует подпись с помощью публичного ключа, чтобы раскрыть ваш хэш, и заново хэширует сам объект. Если эти два хэша окажутся одинаковыми, то будет понятно, что объект — тот самый, который вы подписали. Если нет, то либо объект был изменен, либо подпись подделана; а стало быть, от объекта лучше отказаться.

В случае Secure Boot публичный ключ Microsoft включен в прошивку материнской платы, поэтому

все, что они ни подписывают своим приватным ключом (будь то операционная система или новый ключ), перед использованием может быть проверено.

Ботинки на платформе

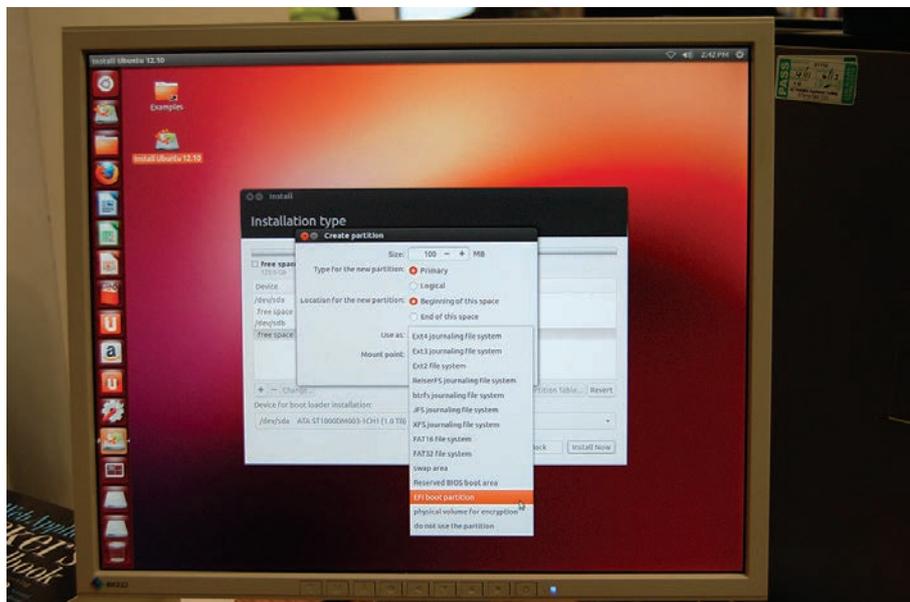
Теперь давайте поближе рассмотрим, как работает Secure Boot. Он крутится вокруг четырех вещей, содержащихся в прошивке материнской платы: Platform Key (PK), Key Exchange Key (KEK), базы данных подписи (db) и базы данных отозванных ключей (dbx). Ни один из этих компонентов не зафиксирован жестко в ПК, и их может изменить физически присутствующий пользователь, даже не зная ни одного из существующих ключей (например, вы должны перезагрузить систему и войти в setup, так что это нельзя сделать через SSH, VNC или любой другой удаленный протокол).

Platform Key — ключ самого высокого уровня, и его можно использовать для подписания новых KEK. Key Exchange Key находится на следующем уровне. Его можно использовать для подписи новых пунктов в db и dbx. Материнская плата может хранить более одного KEK.

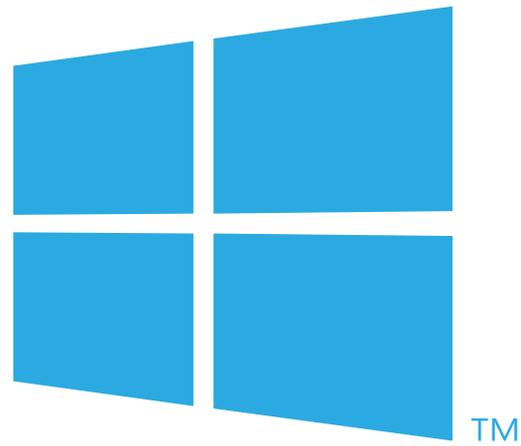
Db и dbx — основные части, которые, собственно, и выполняют работу. Db содержит публичные ключи, которым система доверяет подписывать код и согласна его запускать. A dbx, наоборот, содержит публичные ключи, которым очевидно нет доверия (например, те, которые точно были скомпрометированы).

По умолчанию, когда вы покупаете компьютер, сертифицированный Windows 8, PK будет ключом производителя. Там будет один KEK — Microsoft'овский, и одна запись в db, и это будет Microsoft'овский ключ UEFI.

Если вы запустите Windows 8, материнская плата проверит подпись ядра Windows 8 с помощью ключей в db и dbx. Она обнаружит наличие совпадений с записями в db, а в блокирующей базе dbx совпадений не будет, и она разрешит системе загружаться.



➤ Разбивая диск на разделы, помните, что UEFI требует отдельного раздела — он не работает в MBR, как это было в прежние времена.



► Где бы вы ни увидели на компьютере Windows 8, это означает, что на нем есть Secure Boot и что по умолчанию он активирован.

смогут и дальше работать с будущими версиями Secure Boot.

В нашей статье мы случайно упустили из вида одно из главных ограничений Secure Boot. То есть требование не отключать его на системах ARM. На это можно смотреть по-разному. Мы можем просто игнорировать это, поскольку компьютеров Windows RT не так уж много, и вне зависимости

от наличия Secure Boot, немало найдется желающих их приобрести. Мы также могли бы подчеркнуть тот факт, что многие из самых продаваемых ARM-компьютеров (до Windows 8) за-

блокировали программы загрузки (среди них iPad и iPhone). Мы даже можем предположить, что на многих из самых продаваемых компьютеров ARM, не являющихся телефонами или планшетами, есть неблокируемые программы загрузки (Chromebook, RaspberryPi). И все это, вместе взятое, говорит о том, что нам не стоит переживать по поводу закрепления Secure Boot на машинах ARM.

Есть, однако, еще один аспект проблемы. Microsoft, вероятно, не смог бы настоять на том, чтобы производители заблокировали программу загрузки на x86 машинах. По крайней мере, в ЕС это наверняка бы сочли злоупотреблением доминирующей позицией на рынке, а из-за этого у Microsoft уже были неприятности. Однако с машинами ARM они не претендуют на доминирующую позицию на рынке, и поэтому могут вполне легально злоупотреблять своей не-доминирующей позицией на рынке сколько пожелают. Помня об этом, некоторые стали полагать, что Secure Boot на ARM — то, что Microsoft хочет делать, а на x86 — то, что они вынуждены делать. В будущем, полагают эти люди, версия x86 будет все ближе и ближе подходить к тому, что имеется на ARM, пока в конечном итоге не запретит пользователям делать что-либо с собственным компьютером. Кэти Малмроуз [Cathy Malmrose], руководитель ZaReason, одна из таких людей, и она недавно пообщалась с нами. Ее видение проблемы вы найдете на стр. 36.

Пока Linux работает прекрасно на машинах x86 Windows 8 — но мы все равно будем отслеживать изменения Secure Boot в будущих версиях. **LXF**

Выбор из трех

Итак, как поступает Linux? Ну, опций несколько. Дистрибутив может создать собственный KEK, который пользователи могут загрузить в свою систему вручную, и затем подписать с его помощью свое ядро. Потом разработчики могут попытаться заставить производителей подписать этот KEK своим ПК, и таким образом распространить его наравне с их собственными средствами установки. Однако первое из перечисленного технически сложно, а последнее требует наличия такой влиятельности, которой могут похвастаться немногие дистрибутивы (если такие вообще есть).

Есть и третий способ. За одновременную оплату в размере \$99 Microsoft подпишет загружаемое ПО — однако здесь существует некая загвоздка: ПО само должно соответствовать их представлениям о безопасности. То есть должно продолжать цепь подписей и не позволять непроверенному коду захватить компьютер. Если бы не это условие, сообщество Linux могло бы просто подписать программу загрузки и затем использовать ее так же свободно, как мы используем на данный момент Grub. Конечно, если бы не эта загвоздка, и если бы была подписана нормальная программа загрузки, создатели вирусов могли бы использовать эту подписанную программу загрузки для запуска их вредоносного ПО, что ниспровергнуло бы самую суть Secure Boot.

Решение, которого пока придерживаются большинство коммерческих дистрибутивов, заключается в использовании программы предзагрузки. Ubuntu, Fedora и OpenSuse — все делают именно так. Эта прослойка загружается прежде всего остального, и делает только одно: предоставляет дополнительный ключ, индивидуальный для дистрибутива. Дополнительным ключом можно подписать обычную программу загрузки Linux (как правило, Grub 2), которая, в свою очередь, загружает подписанное ядро Linux, и цепь доверия замыкается.

С этим решением, однако, имеются проблемы. Во-первых, оно означает, что пользователь не может скомпилировать собственное ядро (поскольку его не подписать с помощью ключа дистрибутива). Во-вторых, это означает, что Microsoft может прекратить действие подписи программы предзагрузки (то есть добавить ее в dbx) в любой момент,

когда им этого захочется. Что потенциально способно оставить вас с носом.

Есть, впрочем, еще один способ — он только только появился, и на момент написания этой статьи используется далеко не всеми дистрибутивами. Но поскольку он не включает подписания чего-либо в дистрибутиве, мы ожидаем, что он станет излюбленным методом для небольших или некоммерческих дистрибутивов Linux.

Этот метод использует программу предзагрузки, подобную той, что разработана в Linux Foundation. Она тоже подписана ключом Microsoft, однако вместо специально предназначенных для дистрибутивов ключей она использует добавленные пользователем хэши. Это означает, что когда пользователь обновляет ядро, ему придется добавить новый хэш в базу данных программы предзагрузки. В основном это может делаться автоматически, но тогда будет нарушена цепь доверенности, создаваемая подписями. В теории это должно быть сведено к тому, чтобы пользователь нажал на кнопку, подтверждая, что хочет это сделать. Однако происходить это должно во время загрузки, а стало быть, не автоматизируется. Тогда цепь доверия сохранится, потому что это —

«Нас приятно удивило то, что по большей части все это работает.»

действие, выполненное физически присутствующим пользователем, и не может быть выполнено каким-либо вредоносным ПО.

Будем надеяться, что в будущих релизах эта программа и программа предзагрузки от Linux Foundation будут скомбинированы в одну программу предзагрузки, которую смогут использовать многие дистрибутивы.

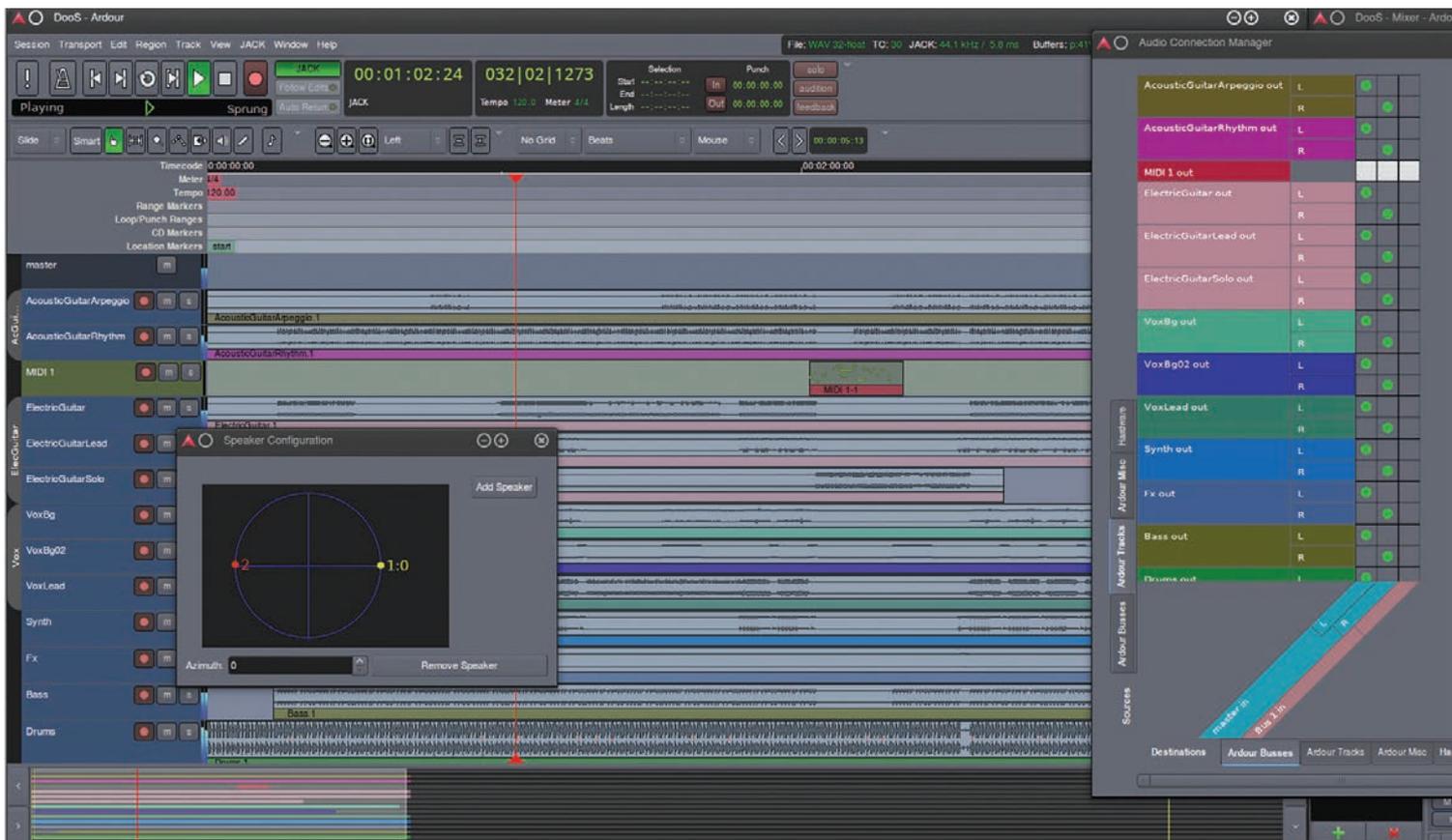
Как это работает?

Последнее, что мы рассмотрим — как это все работает на практике. Честно говоря, мы не питали особых надежд, когда получили ПК с Windows 8. Мы загрузили подборку дистрибутивов, записали их на DVD и вступили в бой. Но нас приятно удивило то, что по большей части все это работает. Или, по крайней мере, работало, если выбрать опции по умолчанию. В Ubuntu, openSUSE и Fedora мы смогли установить на наш ПК с включенным Secure Boot безо всякой мороки с ключами, программами загрузки и вообще с чем бы то ни было. Однако у нас возникли проблемы, когда мы попытались создать настраиваемые вручную разделы на диске, или двойную загрузку (см. врезки на странице).

Когда было объявлено о Secure Boot, многие сочли, что он перекроет доступ Linux на ПК. К счастью, это не тот случай, и ряд основных дистрибутивов уже поддерживают эту систему. Еще больше дистрибутивов готовы начать ее поддерживать уже сейчас, когда программа предзагрузки от Linux Foundation готова к использованию. Мы можем только надеяться, что эти решения

Опасности

Наш опыт показал, что загрузка UEFI послужила причиной возникновения проблем с программами установки, от которых мы уже слегка отвыкли. Случайно нам удалось создать незагружаемые системы, и даже настроить их так, чтобы Windows не устанавливалась повторно. В новой технологии это отчасти неизбежно. Но это также означает, что вам надо быть крайне осторожными. Мы настоятельно рекомендуем до начала установки создать резервную копию всего диска, которой вы сможете воспользоваться для восстановления системы, если возникнут проблемы.



Внутри Ardour 3

Откройте для себя лучшее аудиоприложение для Linux с небольшой помощью его автора, Пола Дэвиса, и Грэма Моррисона.

Писать музыку на ПК или ноутбуке сегодня может каждый: и на OS X, и на Windows есть море приложений, которые доведут воплощение ваших идей от пары нот до полноценного релиза за считанные часы. И не надо быть профессионалом или даже иметь музыкальное образование, ведь благодаря технической революции всем многообразием инструментов может овладеть любой. За последние 10–15 лет старое поколение музыкальных секвенсоров было сметено виртуальными аналогами всех средств звукозаписи — синтезаторов, эффектов в реальном времени, бесчисленных регуляторов, виджетов и визуальных отображений. Результатом стало появление на свет новых музыкальных

форм и целого поколения музыкантов, посвятивших себя созданию, сведению, мастерингу и производству собственной музыки.

Linux сыграл в этой революции значительную роль. Ведь именно он был в сердце ставшей прорывом рабочей станции-синтезатора OASYS от Korg и Receptor от Muse Research — внешнего хоста для плагинов, а также многих других экспериментальных проектов, таких как контроллер Eigenharp. Есть и скромная коллекция настольного ПО, с потрясающим трекером *Renoise*,

«Корона многоканальной записи на жесткий диск принадлежит Ardour.»

модульным MIDI- и аудиосеквенсором *EnergyXT*, плюс надвигается мощная *Bitwig Studio*, грозя произвести фурор. Но корона многоканальной записи на жесткий диск принадлежит исключительно *Ardour*.

Автор проекта — Пол Дэвис [Paul Davis], и он же бессменно им руководит, имея несколько ассистентов. Мы встретились с ним, дабы рассказать про *Ardour* и заодно поподробнее узнать, какова история проекта и в чем его суть.

Ardour возник в 1990-х, когда Пол пытался записать вывод своих синтезаторов на Linux-ПК, но не нашел для этого подходящего ПО. «Я понял, что писать звук на кассету — не то, что мне нужно, — говорит он. — И знал, что входит в моду прямая запись на диск,



поэтому нехотя купил подержанный 486-й с частотой 75 МГц. Linux я использовал с 1994 года, так что установил Red Hat и стал перебирать разные доступные программы звукозаписи (на дворе был конец 1997 года). И был разочарован: либо ничего не работало, либо имело безнадежно ограниченную функциональность».

Это и привело Пола к покупке кое-каких аудиоустройств и первым строкам кода. «В конце 1999 я увидел звуковую карту RME Hammerfall на 24 входа и 26 выходов. Решил, что это стоящая вещь, купил, а затем адаптировал и расширил исходный драйвер от Винфрида Рича [Winfried Rietzsch], чтобы включить его в тогда еще новую систему ALSA. Но когда все получилось, я понял, что нужно какое-то приложение, чтобы этим пользоваться: так и родился Ardour (по крайней мере, в уме). Изначально прямой целью было создать замену/альтернативу для двух цифровых магнитофонов с ADAT-интерфейсом, способных записывать 24 дорожки 24-битного аудио с частотой сэмплов до 96 кГц».

Пол продолжает рассказ о своих трудностях роста: «Создать многоканальный магнитофон оказалось задачей простой, но на деле практически бесполезной, ведь все, что он умел — это записывать и воспроизводить. Тогда (а по большому счету, и сейчас) было мало аудиоредакторов, способных справиться с обработкой такого объема данных, и стало ясно, что Ardour требуется свой редактор. Казалось, что с этим все будет просто, и, пожалуй, это было мое самое большое заблуждение. Если бы я знал тогда, что значит

написать цифровую рабочую аудиостанцию (а не устройство записи на жесткий диск и не редактор аудиофайлов, например), я бы в жизни за это не взялся».

Теперь же, в 2013, прошло уже 6 лет после выхода *Ardour 2.0*. А появившийся в марте этого года *Ardour 3* являет собой важнейший этап, и не столько благодаря реализации поддержки MIDI-данных, превращающей его из приложения для работы с аудио в лучшее средство создания музыки в Linux, но и потому, что *Ardour* — единственное открытое приложение, пригодное для профессиональной многоканальной записи и редактирования. Это свободный эквивалент таких приложений, как *Cubase* от Steinberg, и такой же синоним для приложений аудио, как *Photoshop* или *GIMP* — для графики. *Ardour* позволяет записывать и импортировать множество аудиодорожек, чтобы затем наложить их на хронометраж и обработать.

В отличие от широко известного *Audacity*, в арсенале которого — базовые функции редактирования аудио и встроенные эффекты, *Ardour* устроен как настоящая студия звукозаписи. В нем есть микшерский пульта, студийный магнитофон и набор эффектов, а редактирование происходит онлайн, без ущерба для самого файла. Просто подключите гитару, наложите эффекты искажения и задержки, и тут же услышите результат. Нажмите запись, чтобы записать свое исполнение или пре-эффекты, для последующего микширования и мастеринга. Добавьте к этому возможность параллельной записи нескольких дорожек, плюс возможность вернуться и добавить другие инструменты, и вот вам *Ardour* во всей красе.

Работа с аудио

Новичкам, во многом как и в *GIMP*, поначалу трудно освоиться с *Ardour* из-за его кажущейся необъятности. Как признает сам Пол, «любую цифровую рабочую станцию аудио, которая имеет хоть какое-то право так называться, освоить очень трудно. Приходит на ум тот же *Garageband* — милая программа, но легких путей к ее освоению у новичков практически нет».

Хотя новые пакеты почти всегда можно добыть из репозитория вашего дистрибутива, сначала вам придется поколдовать с настройкой аудио (ну, или установить Ubuntu Studio 13.04 и просто начать работать). *Ardour* не может взаимодействовать с аудиосистемой Linux, ALSA, напрямую. Для этого служит пристойка под названием *Jack*, с помощью которой можно подключить совместимый синтезатор, соединив его аудиовыходы со входами *Ardour*, чтобы записать вывод. Та же функция используется в *Ardour* и для внутренних подключений, что делает его необычайно гибким во взаимодействии с другими аудиопрограммами. И не диво, что *Jack* создал тоже Пол Дэвис.

Мы спросили Пола, хотелось ли бы ему, чтобы Linux-аудио стало больше похоже на платформу CoreAudio на технике Apple, которую *Ardour* также поддерживает.

«Любую цифровую рабочую станцию аудио освоить очень трудно.»

«С одной стороны, (Linux) открытый, понятный и постижимый, — ответил Пол. — С другой, у CoreAudio гораздо лучше устроено само ядро — хотя к моменту, когда до него добираются разработчики приложений, отличия уже не столь разительны. И там, и там *Jack* предлагает лучший вариант абстракции ввода/вывода с учетом особенностей конкретного приложения».

Но не играют ли эти сложности скорее на руку версии OS X, нежели Linux? Пол так не считает.

«Судя по количеству загрузок с ardour.org, их шансы равны. Хотя многие пользователи Linux получают *Ardour* из своих дистрибутивов; этими данными мы не располагаем. Как только выйдет »

Новые функции

- » Запись, воспроизведение и редактирование MIDI
- » Поддержка виртуальных инструментов
- » Матричный диалог настройки маршрутизации сигнала
- » Мониторинг записи
- » Переработанное управление соло
- » Внутренняя передача
- » Возможность наложения слоев
- » Удаление тишины
- » Таблицы соответствий MIDI-данных
- » Улучшение поддержки Mackie Control

Какая из них у Пола любимая? «Во-первых, наложение областей. Новая архитектура соло/прослушивания тоже потрясающая. Хотя, как и все хорошие функции, они просто скромно и тихо выполняют свои задачи. Быть может, поддержка MIDI?».



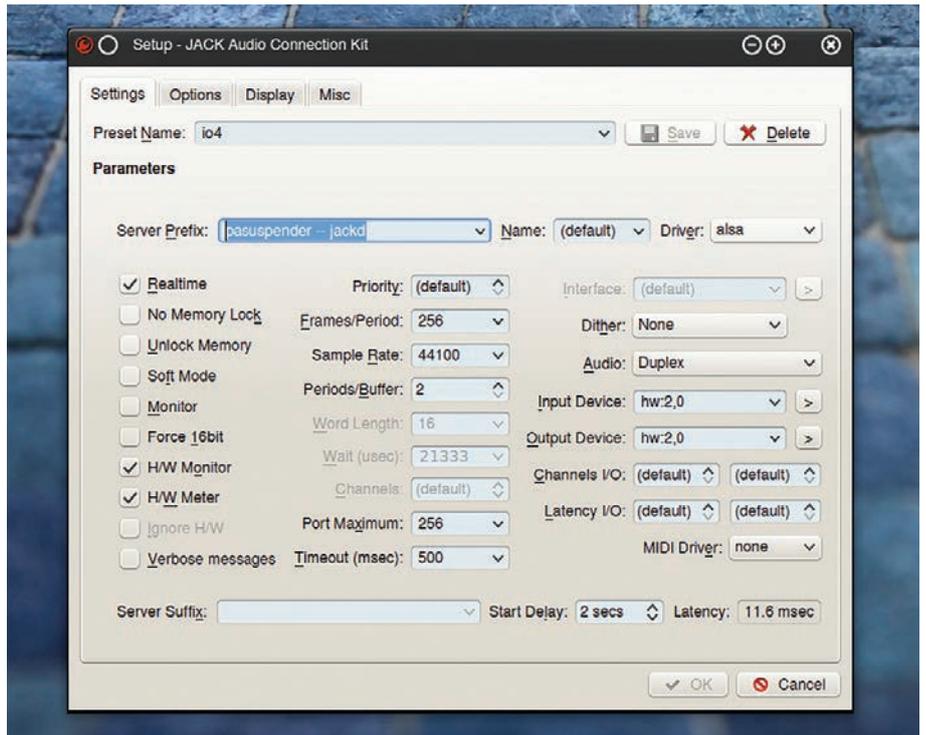
» Магией *Jack* внутри *Ardour* можно управлять через менеджер подключений.

Внутри Ardour 3

Ardour 3.0 для OS X, у нас появится новый способ расчета: программа будет автоматически регистрироваться на ardour.org при запуске».

Основная проблема Jack — его несовместимость с PulseAudio, лежащего в основе большинства дистрибутивов Linux. PulseAudio может непрерывно взаимодействовать с ALSA, а Jack — нет. Поэтому на время сеанса записи PulseAudio приходится отключать. На некоторых дистрибутивах, вроде Ubuntu Studio, Jack запускается еще при загрузке, так что это не проблема. Но большинству придется немного помочь.

Лучший выход, который мы нашли — это зайти на страницу установки QjackCtl и убедиться, что в поле префикса на сервере указано: 'pasuspender -- jackd'. Содержание поля зависит от способа функционирования Jack, но для большинства дистрибутивов это сработает. На той же странице следует выбрать 'Duplex' аудио, чтобы разрешить параллельно запись и воспроизведение, а также указать нужные устройства ввода/вывода. Мы также обнаружили, что лучший способ подобрать работающую конфигурацию — использовать выпадающий список справа от обеих полей. Возможно, вам также захочется поэкспериментировать с уменьшением показателей для Frames/Period и Periods/Buffer, что сократит период задержки между вводом сигнала и его воспроизведением (известный как латентность [latency]), но наш совет — сначала довести до ума рабочие настройки. Для тестирования текущих настроек щелкните OK, чтобы закрыть



» Улучшение качества аудио обычно уменьшает время задержки сигнала.

окно Настройки, и 'Start' в главном окне QjackCtl. Запустится сервис Jack; при наличии каких-либо проблем вы увидите сообщения о них, и вам позволят запустить сам Ardour 3.

Мы поинтересовались у Пола, желает ли он, чтобы Jack официально вошел в состав аудиостека Linux и пользователи могли бы не вникать во все это, а просто запускать приложение.

Обзор Ardour

Transport [Транспорт]

Сделайте трек доступным для записи и нажмите, чтобы начать захват аудио, и включите метроном.

Timeline [Временная шкала]

Определяйте точки начала и конца записи, или делайте разметку треков для CD.

Groups [Группы]

Объединяйте дорожки, например вокал и гитару, чтобы ими было проще управлять.

MIDI Tracks [MIDI-треки]

Это главное новшество позволяет писать музыку на аппаратных и программных синтезаторах.

Audio Tracks [Аудиотреки]

Поддерживается импорт и экспорт аудио, выделение областей и редактирование с неограниченным откатом.

Effects [Эффекты]

Добавляйте эффекты на дорожки и шины, изменяйте параметры внутри каждого окна.

Overview [Обзор]

Управляйте масштабом областей с помощью мыши, чтобы видеть полную картину.

Mixer [Микшер]

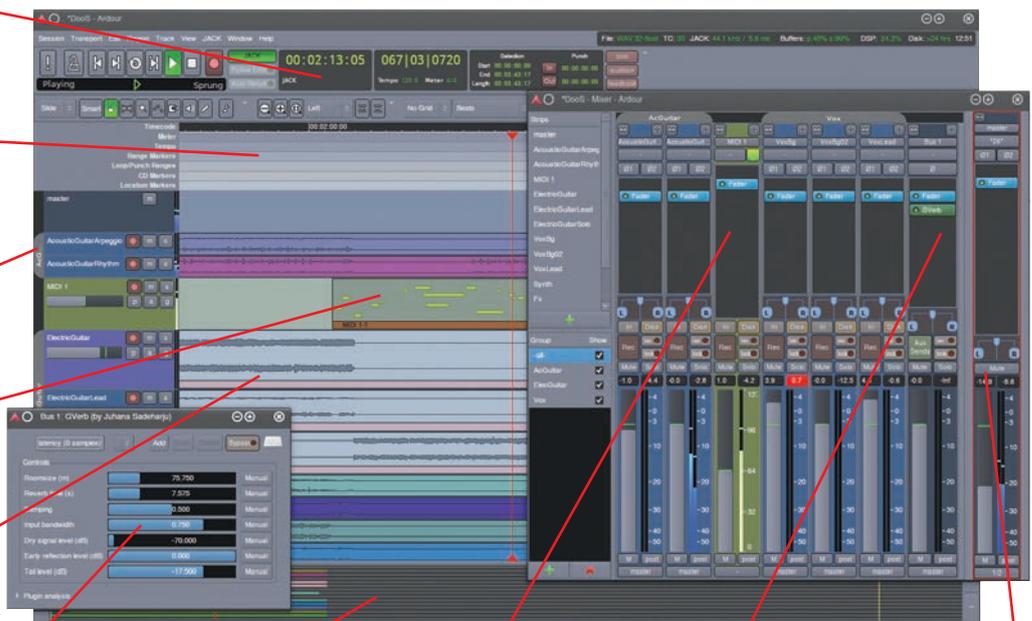
Регулируйте громкость каждого трека, добавляйте новые и меняйте порядок.

Busses [Шины]

Шины можно использовать для добавления эффектов к группе дорожек.

Master Bus [Мастер]

Данная шина используется для сведения звука в один файл.



Как помочь Ardour

Ardour довольно настоятельно просит вас внести свой вклад, когда вы скачиваете приложение с ardour.org. Объясняется это попросту тем, что не может же Пол Дэвис заниматься проектом вовсе без денег. Без поддержки Ardour умрет.

«В начале работы над Ardour у меня были свободные средства, благодаря тому, что я принимал участие в основании Amazon.com, — объясняет Пол. — Со временем личные обстоятельства изменились, прежняя система финансирования становилась все менее жизнеспособной, и пять-шесть лет назад мне потребовалось получать хоть какие-то средства с этого проекта. Мне оказывали разные виды финансовой поддержки: на начальном этапе это были просто пожертвования на добавление конкретных функций в Ardour, затем — спонсорская поддержка от компаний, в нем заинтересованных, проекты Google Summer of Code, действующие партнерства с коммерческими организациями, а также плата за скачивание».

Но золотых гор это не принесло: «Кроме пары коротких периодов коммерческой поддержки, я никогда не получал за Ardour больше средней

американской зарплаты, а часто и того меньше. Но я считаю, что при отсутствии финансовой независимости лучший способ выручить деньги с такого проекта, как Ardour, требующего постоянного внимания, это собрать с миру по нитке».

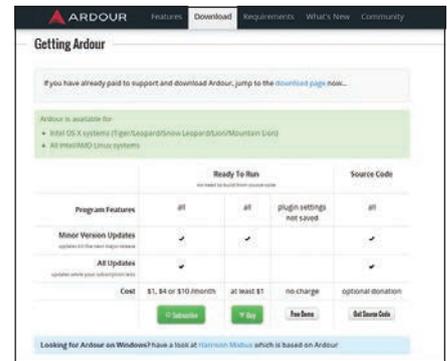
Если вы не можете финансово поддержать Ardour, есть другие способы помочь. Самый простой — это пользоваться им самому и рассказывать друзьям, лично и в Сети. Помощь разработчиков также приветствуется, но нужно иметь опыт.

«Несомненно, исходный код Ardour очень сложен, поэтому тем, кто не имеет большого опыта в программировании, влиться сложно, — говорит Пол. — Даже при работе с интерфейсом часто требуется понимать принципы использования потоков и требований реального времени».

Если же вы не программист, для вас и тогда дело найдется.

«В проекте есть множество сфер, где могут помочь и те, кто далек от разработки, — добавляет Пол. — У нас большие проблемы с документацией. Проекту FOSS не удалось создать приличного учебника для новичков, а наше руководство-справочник

неполное и во многом устаревшее. Пользователям Ardour, как новым, так и опытным, очень бы пригодились качественные и подробные обучающие видео — например, такие, как у Blender (инструмент для 3D-анимации).



Хотя просьбы об оплате и раздражают, но, как объясняет Пол Дэвис, без вложений Ardour просто не выживет.

«Я предлагал дистрибутивам это делать еще в середине 2000-х, — объяснил он. — PulseAudio гораздо лучше подходит обычным пользователям ПК и ноутбуков, а также лучше приспособлен к мобильной среде. Печально раз за разом видеть халтурно выполненную интеграцию Jack и PulseAudio».

Пол продолжает: «Подозреваю, отчасти проблема в том, что многие дистрибутивы восприняли PulseAudio как панацею от всех проблем с аудио в Linux. Но это не так. Чтобы в Linux все было как надо, придется значительно перекроить ALSA по модели CoreAudio. Пока же ни PulseAudio, ни Jack нельзя считать универсальными, и дист-

и множество других новшеств, повышающих качество микширования без лишних усилий. У нас с Harrison'ом финансово взаимовыгодное сотрудничество: если они создадут программную экосистему на основе Ardour, выиграют все. А если они будут просто поглощать пользователей Ardour, хуже все равно не будет. Harrison — одна из немногих компаний, понимающих, как эффективнее использовать открытую природу Ardour, и мне хочется, чтобы Mixbus развивался параллельно Ardour».

Данный релиз — огромный скачок в развитии Ardour, становящегося серьезным кросс-платформенным инструментом для работы со звуком, и уникальным в среде Linux. А его долгая разработка многому научила Пола.

«Работая над Ardour, я годами пристально следил за миром Open

Source, и у меня сложилось впечатление, что в нем есть три основных категории программных проектов, — рассказал он. — Первую составляют масштабные и широко применяемые проекты. Сюда входят ядро Linux, Firefox и разные версии OpenOffice. В их успехе заинтересованы миллионы пользователей и множество коммерческих организаций. Может, деньги туда рекой и не текут, но большое число пользователей и партнеров делают свое дело, и эти проекты сильны своим человеческим ресурсом.

Во вторую категорию попадают небольшие программы. Сразу приходит на ум слово «приложе-

ние» — маленький инструмент, позволяющий выполнять всего пару задач, зато хорошо. Такие проекты в основном создаются одним-двумя людьми, довольно быстро, и, что удивительно, когда-нибудь завершаются. Финансовая поддержка здесь, как правило, не проблема — она возникает по мере необходимости, или же создатели вовсе в ней не нуждаются, занимаясь проектом на досуге, параллельно с основной работой.

Третья категория — как раз та, к которой принадлежит Ardour, а также GIMP, Blender и прочие инструменты медиа-производства. Это тоже большие, многосоставные проекты, но широкой аудитории у них нет, по причине их узкой специализации. На развитие такого проекта многие коммерческие фирмы особенно не надеются, ведь у самых заинтересованных клиентов есть собственные варианты подобных инструментов. Над Ardour я работаю не один, но я более чем уверен, что брось я это дело, темпы развития замедлились бы настолько, что он по сути умер бы. Для программ данной категории найти варианты финансовой поддержки их разработки весьма нелегко».

2013 обещает стать прекрасным годом для Ardour: еще более расширится поддержка оборудования и выйдет первый стабильный релиз 3.0. Остается только надеяться, что ему окажут должную помощь, ведь он ее заслуживает. Если вы еще не опробовали этот релиз, сделайте это как можно скорее. А если вы боитесь не разобратся с интерфейсом и принципом его работы, не пропустите полное руководство по Ardour 3 для новичков, начиная со следующего номера. LXF

«Найти варианты финансовой поддержки разработчики весьма нелегко.»

рибутивы должны постараться (местами — сильно), чтобы пользователи смогли легко и уверенно работать с обоими».

Средств для работы с аудио в Linux хоть отбавляй, но когда речь заходит о драйверах, многие производители отключаются. К счастью, многие стандартные USB-аудиоустройства, вроде Alesis iO4, достаточно просто подключить. А если вам требуется более профессиональный вариант, Пол Дэвис рекомендует оборудование RMS для аудиоввода/вывода.

Сотрудничество с производителями так или иначе пошло на пользу проекту, утверждает Пол:

«Мы уже породнились с Harrison Consoles, посредством Mixbus. Это особая версия Ardour, включающая ПО для цифровой обработки сигнала с их прославленных микшерских пультов,





Что за штука...

Ubuntu Accomplishments

Джоно Бэкон ныряет в море знаний о сообществе и компьютере ради призов, выдаваемых Ubuntu Accomplishments.

В Так, Бэкон, ну и что это за Ubuntu Accomplishments?

О Ubuntu Accomplishments [Достижения Ubuntu] — это игрофикация, позволяющая в увлекательной форме опробовать различные возможности вашего компьютера и сообщества. Фишка в том, чтобы при просмотре добраться до них и порадоваться призу, получаемому за каждое достижение; ну, а потом полюбоваться всеми своими трофеями.

В Вы сказали «игрофикация» — а что это, собственно, такое?

О Игрофикация — это методика создания игр, но не по привычным сценариям. Исходя из того, что людям нравится играть, вы можете воплотить любую тему так, чтобы она смогла их увлечь. Некоторые группы пытаются обыграть даже самые заурядные вещи, чтобы превратить их в захватывающие, и весьма успешно. Если уж быть точным, Ubuntu Accomplishments — не настоящая

игрофикация, поскольку суть ее не в соревновании «кто кого»; но и большинство подобных систем нацелено на обретение навыков при выполнении задач за награду.

В Ясно... и что за трофеи я могу добыть в Ubuntu Accomplishments?

О В целом, они делятся на две категории. Первая — это деятельность в сообществе. Тут целый спектр номинаций: скажем, за свой первый

отлов ошибки, участие в Ubuntu Developer Summit, присоединение к Совету сообщества, членство в Ubuntu и прочее. Они охватывают все сферы деятельности сообщества Ubuntu, в том числе разработку, тестирование, документацию, управление, ответы на вопросы [Ask Ubuntu] и прочее. Другая категория связана с программами в вашей системе — например, играми, поставляемыми с Ubuntu. В Ubuntu Accomplishments эти призы появились сравнительно недавно, и их пока немного, но постепенно их число увеличивается.

В Ясно, а как компьютер узнает о ваших достижениях?

О На это имеется два варианта ответа: простой и подробный.

Говоря по-простому, для определения ваших достижений в сообществе фиксируются некоторые реквизиты (такие как почтовый адрес на *Launchpad* или ссылка на ваш профиль в Ask Ubuntu), после чего система будет автоматически их отслеживать и присуждать призы.

Что касается программ, есть два способа. Во-первых, проводится периодическая проверка достижений. Либо приз присуждается самим приложением.

«В Ubuntu dash есть специальная линза, отображающая ваши трофеи.»

В ...а можно подробный ответ, для тонких ценителей?

О В Ubuntu Accomplishments есть внутренний демон, загружающий набор готовых достижений. Например, *ubuntu-community-accomplishments* включает все достижения, связанные с сообществом. Каждое достижение содержит два ключевых файла:

- » информацию о самом достижении (документация, доступная через обозреватель);
- » исполняемый скрипт для проверки его выполнения.

При запуске обозреватель Ubuntu Accomplishments подключается к демону и сначала определяет, требуется ли аутентификация для проверки достижений. Когда вы добавите свои реквизиты, поочередно будут запускаться скрипты для проверки, достигли ли вы чего-нибудь. Если достижения есть, генерируется файл приза и сохраняется в вашей системе.

В Секундочку, а если взять да подделать достижения, подсунув фальшивый файл?

О А-а, тут не все так просто. Когда демон фиксирует достижение, файл приза генерируется и отправляется в Ubuntu One, персональный облачный сервис Ubuntu, где он автоматически синхронизируется с сервером проверки данных.

Этот сервер независимо подтверждает получение приза, и если файл проходит проверку, ему присваивается GPG-ключ и он вновь синхронизируется с компьютером.

Обозреватель достижений будет отображать только те, что распознаны при помощи GPG-ключа. Если приз не распознан, он будет скрыт. Это позволяет эффективно бороться с поддельными достижениями.

К сожалению, проверить достоверность программных достижений на вашем компьютере невозможно; но подделывая их, вы по сути обманываете сами себя.

В Выходит, мне придется платить за учетную запись в Ubuntu One, чтобы этим пользоваться?

О Нет. Использование Ubuntu One до 5 ГБ совершенно бесплатно, а призы за достижения, работающие в фоновом режиме, представляют собой всего лишь крошечные текстовые файлы, занимающие ничтожную часть этого пространства.

Еще одно преимущество Ubuntu One в том, что ваши призы синхронизируются на всех ваших устройствах Ubuntu.

В А вот эти скрипты, что проверяют достоверность призов — они все работают постоянно?

О К счастью, эта система достаточно разумна, чтобы не гонять все скрипты подряд. В Accomplishments есть удобная схема зависимостей, позволяющая настроить последовательность завершения одного задания, прежде чем можно будет перейти к другому. Например, стать членом Ubuntu можно, только имея учетную запись в *Launchpad*, так что проверка этого достижения будет осуществлена только после подтверждения регистрации.

Система будет активировать только достижения, доступные вам в данный момент; заблокированные запускаться не будут. Это станет возможным после удовлетворения всех зависимостей и снятия блокировки.

Скрипты проверки будут также запускаться каждые 15 минут, составляя небольшое число периодически исполняемых.

В Ну, а как же я смогу просмотреть свои трофеи?

О Есть несколько способов. Во-первых, при помощи встроенного обозревателя Ubuntu Accomplishments. Он включает полный перечень возможностей (еще не совершенных достижений), а также призов за уже выполненные задания.

Помимо этого, в меню Ubuntu dash есть специальная линза, отображающая все ваши возможности и призы. Она предоставляет быстрый доступ ко всей информации даже без открытия приложения. Наконец, сейчас идет работа над созданием web-интерфейса для ваших трофеев на <http://trophies.ubuntu.com>, где можно будет просматривать свои призы, а также демонстрировать их другим, в том числе и в социальных сетях.

В Значит, это работает только для пользователей Ubuntu?

О Безусловно, изначально система была задумана для Ubuntu, но никаких специфических для Ubuntu технологий там нет. Платформа написана на Python и PyGTK. Единственный Ubuntu-компонент — Ubuntu One, но он доступен и в других дистрибутивах (даже на Windows и Mac).

По сути, хотя система официально доступна из Ubuntu, при желании ее можно перенести куда угодно.

В Хорошо. Это любопытно. С чего же мне начать?

О Финального релиза Ubuntu Accomplishments пока нет, но есть несколько промежуточных, с которыми можно поиграть. Они доступны для всех версий Ubuntu с 12.04 и выше.

Устанавливается Ubuntu Accomplishments легко, нужен только установленный PPA. Просто скопируйте и вставьте следующие команды:

```
sudo add-apt-repository
ppa:ubuntuaccomplishments/releases

sudo apt-get update
sudo apt-get install accomplishments-
daemon accomplishmentsviewer
ubuntu-community-accomplishments
ubuntu-desktop-accomplishments
```

Установив пакеты, ищите 'Ubuntu Accomplishments' в меню dash и загружайте клиент. Если вы не зарегистрированы в Ubuntu One, можете сделать это в самом приложении, чтобы синхронизировать свои трофеи с Ubuntu One. Наконец, у вас запросят детали аутентификации, чтобы можно было проверять достижения.

А затем подождите пару минут, пока призы пройдут проверку и начнут появляться на экране.

В Учитывая, что это до сих пор в разработке, где мне найти помощь?

О Вам поможет подписка на рассылку с <https://launchpad.net/~ubuntu-accomplishments-contributors>, либо IRC-канал #ubuntu-accomplishments на Freenode.

В Понятно. А я сам могу внести вклад в развитие проекта?

О К счастью, этому посвящен большой объем отличной документации на <https://wiki.ubuntu.com/Accomplishments> — ну и можете рассчитывать на каналы связи, чтобы спросить других, как начать. 



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.

Привет, мир

В мире открытого ПО сильно традиция начинать знакомство с языком с написания первой программы, которая отображает (в любом допустимом виде) слова “hello world [привет, мир]”. Среди фанатов есть разные мнения о том, писать ли эти слова с заглавной буквы и нужно ли ставить восклицательный знак. Насколько я знаю, начало этой традиции положил Брайан Керниган [Brian Kernighan] в классической книге «Язык программирования C [The C Programming Language]».

Сам код обычно прост — всего одна-две строки — но чтобы заставить его работать, нужно сначала настроить среды разработки и выполнения, а также освоить процесс редактирования, сборки и запуска приложения на заданном языке. Я сам часто ощущаю глубокое удовлетворение, видя слова “hello world” в новых и сложных для освоения рабочих средах. Это как первый волнующий поцелуй с новой подружкой. После него все остальное — уже рутина.

В Википедии есть статья http://en.wikipedia.org/wiki/List_of>Hello_world_program_examples, предлагающая примеры программы “hello world” на (примерно) 102 языках, хотя не все приведенные там варианты стоит воспринимать всерьез.

Существует и обратная традиция — когда программу пытаются максимально усложнить. Примеры есть на <http://www.gnu.org/fun/jokes/helloworld.html>. С этим связан подход намеренного «затуманивания [obfuscation]», который принял форму искусства и требует предельных знаний языка. Примеры можно увидеть на <http://codegolf.stackexchange.com/questions/307/obfuscated-hello-world>.

На худой конец, можно применить административный подход:

```
mail Энди
Энди, будь добр, напиши мне программу распечатки 'hello world'
Она нужна мне к утру.
^D
chris.linuxformat@gmail.com
```

Эзотерическое системное администрирование из причудливых заворотов кишок серверной



Коротко о Tomcat

Берегите мышь — мы выпускаем *Tomcat*, знакомимся с Java и пишем сервлет.

Начиная с самого начала: *Tomcat* [англ. котяра] — это web-сервер. Как и *Apache*, он владеет языком протокола HTTP и отвечает на запросы браузера о статическом контенте — HTML-файлах и изображениях. Но *Tomcat* более узко специализирован, чем *Apache*: его истинное призвание — быть контейнером сервлетов Java. Это означает, что он способен хранить динамически генерируемые web-страницы, использующие Java как язык на стороне сервера. Существует стандарт, подробно описывающий среду, где будут запускаться эти приложения,

и *Tomcat* изначально разрабатывался как реализация данного стандарта.

В первой части нашей серии из двух статей мы начнем с установки пакетов, необходимых для запуска *Tomcat*, а затем посмотрим, как развернуть простое web-приложение из двух страниц. Как и на других своих уроках в этом журнале, я буду работать с чистой установкой CentOS 6.2, и если вы хотите следовать за мной, можете начать с установки CentOS на виртуальную машину.

Собираем части вместе

Сперва установим необходимые компоненты. Все необходимое есть в репозиториях CentOS, и нам нужно лишь вытянуть оттуда необходимые пакеты с помощью *yum*. В таблице приведены пять пакетов, которые нам потребуются; итак, приступайте и выполните команду **yum install** с каждым из них. Я подожду, пока вы закончите...

Ну вот, *Tomcat* установлен, и нужно запустить сервер, позаботившись, чтобы после перезагрузки системы он запускался автоматически. Нам помогут следующие две команды (запустите их от имени суперпользователя-root):

```
# service tomcat6 start
# chkconfig tomcat6 on
```

Так как мы установили пакет *tomcat6-webapps*, который формирует домашнюю страницу по умолчанию, теперь можно открыть браузер, зайти на <http://localhost:8080> и получить в награду

Страницы Java

Tomcat также может выдавать содержимое, структурированное в виде серверных страниц Java, которые состоят из HTML-разметки со вставками кода на Java в тэгах `<%` и `%>`. Эта концепция расширяется с помощью стандартной библиотеки тэгов (JSTL); тэги реализуют операции, общие для многих web-приложений, такие как обработка XML-документов и доступ к реляционным базам данных. Различие между сервлетом и JSP в целом в том, что сервлет включает HTML в каркас на Java, тогда как JSP включает код Java в каркас на HTML. Подход JSP будет более знакомым вам, если вы уже поднатрели, скажем, в ASP, .NET или LAMP.

**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

› Установите эти пакеты с *yum*, чтобы получить рабочую конфигурацию *Tomcat*. Пакет *javasdk* нужен только для компиляции своих сервлетов. Но вы же будете их делать, правда?

Пакет	Содержимое
<code>tomcat6</code>	Основной серверный пакет, устанавливает все основные компоненты сервера
<code>tomcat6-admin-webapps</code>	Web-приложения менеджер хоста (<i>host-manager</i>) и менеджер (<i>manager</i>)
<code>tomcat6-docs-webapp</code>	Web-приложение с документацией по <i>Tomcat</i>
<code>tomcat6-webapps</code>	Web-приложения <i>ROOT</i> и примеры (домашняя страница и примеры программ)
<code>java-1.6.0-openjdk-devel</code>	Утилиты разработки Java, в том числе <i>javac</i> , компилятор Java

домашнюю страницу, показанную на снимке экрана. Учтите, что по умолчанию *Tomcat* слушает порт 8080 (а не 80), поэтому его нужно указать в адресе. На этой странице есть ссылки на документацию *Tomcat* (которую мы также установили, из пакета *tomcat6-docswebapps*), и здесь вы можете легко увлечься и потратить остаток дня на чтение. Также есть ссылки на несколько примеров кода.

Наш первый сервлет

Приступим к делу и посмотрим, как развернуть на *Tomcat* простой сервлет. Чтобы немного усложнить традиционную “hello world”, мы создадим web-приложение из двух страниц. Первая страница — чистый HTML: на ней есть форма с текстовым полем, где пользователь должен ввести свое имя, и кнопка для отправки формы сервлету. Вторая страница — сервлет. Она просто приветствует пользователя по имени, полученному из текстового поля на форме.

Форма

Начнем с формы. Ниже приведен ее код. Пока просто поместите ее в файл `formexample.html` в своем каталоге `home`. Позже мы скопируем ее куда полагается.

```
<html>
<head>
  <title> Простой пример формы </title>
</head>
<body>
  <h1> Простой пример формы </h1>
  <form action="pootele" method="GET">
    Введите ваше имя:
    <INPUT type="text" name="name" />
    <INPUT type="submit" name="Submit" value="Submit Name">
  </form>
```

```
</body>
</html>
```

Сам сервлет

Теперь займемся кодом нашего сервлета. В нем всего одна «активная» строка — она обращается к данным, введенным в форму. Однако чтобы быть сервлетом, класс Java должен наследоваться от базового класса `HttpServlet`. Поэтому вокруг этой единственной строки кода потребуется небольшая обертка. Для начинающих — нам надо определить класс, наследуемый от `HttpServlet`. Внутри этого класса следует написать метод `doGet()`, который *Tomcat* будет вызывать, когда браузер пользователя выполнит запрос HTTP GET к нашей странице. Также можно написать метод `doPost()` для обработки запросов HTTP POST. Затем идет внутренний слой обертки — он необходим, потому что мы хотим вернуть браузеру правильно сформированный документ HTML, и нужно включить основные структурные тэги HTML, такие как `<head>` и `<body>`.

Короче, вот каким получается код нашего сервлета (номера строк приведены для удобства ссылок, они не являются частью файла):

```
1. import java.io.*;
2. import javax.servlet.*;
3. import javax.servlet.http.*;
4. /* Очень простой сервлет для доступа в поля формы */
5. public class FormExample extends HttpServlet {
6.   public void doGet(HttpServletRequest request,
7.     HttpServletResponse response)
8.     throws IOException, ServletException
9.   {
10.    response.setContentType("text/html");
11.    PrintWriter out = response.getWriter();
12.    out.println("<html> <head>");
13.    out.println("<title> Form Example </title>");
14.    out.println("</head> <body>");
15.    out.println("<h1> Hello " + request.getParameter("name") +
16.    "</h1>");
17.    out.println("</body> </html>");
18.  }
```

Рассмотрим основные строки этого файла. В строке 5 задается наш класс (`FormExample`), а строки 6–17 содержат метод `doGet()`. Этот метод получает два параметра: `request` (строка 6) и `response` (строка 7). Мы используем объект `request` для доступа к полю формы (строка 15), а объект `response` — для доступа к потоку `PrintWriter` (строка 11), с помощью которого содержимое отправляется обратно браузеру (строки 12–16).

А если не работает?

Если что-то работает не так, как должно, попробуйте поискать сообщения об ошибках в `/var/log/tomcat/catalina.out`. Также учтите, что браузер кэширует страницы, и это может дурно повлиять на вашу стратегию тестирования, если тестировать невнимательно. Типичный сценарий таков: вы запускаете тест, видите в браузере

ошибку, исправляете ошибку в программе, снова запускаете тест, но все еще видите (кэшированную) страницу с ошибкой. Если у вас есть малейшее подозрение, что именно это и происходит, советую очистить кэш браузера. Например, в *Firefox* выберите Tools > Clear Recent History (Инструменты > Очистить недавнюю историю).

Если вы следуете за мной, создайте этот файл, назовите его **FormExample.java** и скомпилируйте его командами

```
$ export CLASSPATH=./usr/share/java/servlet.jar
$ javac FormExample.java
```

В том случае, если все пройдет хорошо, вы будете вознаграждены файлом **FormExample.class**. Он содержит скомпилированный байт-код Java для этого класса. Учтите, что Java разбрасывает имена файлов. Код класса **FormExample** обязан находиться в файле **FormExample.class**.

Еще один файл, без которого нам никак не обойтись для запуска приложения — дескриптор развертывания **web.xml**. В нем можно задать все, что угодно, но мы лишь связываем «шаблон URL-адреса» с именем сервлета и имя сервлета — с классом, который его реализует. (Конечно, на XML это выражается гораздо длиннее.) Для нашего примера дескриптор выглядит так:

```
<web-app>
<description> Form Example Servlet </description>
<display-name> Form Example </display-name>
<servlet>
<servlet-name>FormExampleServlet</servlet-name>
<servlet-class>FormExample</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>FormExampleServlet</servlet-name>
<url-pattern>/pootle</url-pattern>
</servlet-mapping>
</web-app>
```

Иерархия развертывания

У нас есть три файла (форма, сервлет и дескриптор развертывания), и мы готовы развернуть приложение. В спецификации J2EE определяется очень четкая структура каталогов, показанная на рисунке. Чтобы упростить себе жизнь, мы размещаем «корневой каталог документа» приложения в **var/lib/tomcat6/webapps**, потому что любое приложение в этом каталоге автоматически развертывается при запуске *Tomcat*.

Базовый набор каталогов можно создать всего несколькими командами:

```
# cd /var/lib/tomcat6/webapps
# mkdir -p FormExample/WEB-INF/classes
```

Здесь можно немного поиграть с правами доступа, чтобы получить возможность копировать файлы в этот каталог, не будучи администратором. Вот возможное решение. Для начала сделайте вашего пользователя членом группы *tomcat* (эту команду нужно выполнять от имени *root*):

```
# usermod -a -G tomcat chris
```

Замените имя пользователя на свое (если только вам не повезло быть моим тезкой). Потом измените группу файлов в каталоге развертывания на *tomcat* и добавьте право для группы на запись таким образом:

```
# cd /var/lib/tomcat6/webapps
# chown -R root:tomcat FormExample
# chmod -R g+w FormExample
```

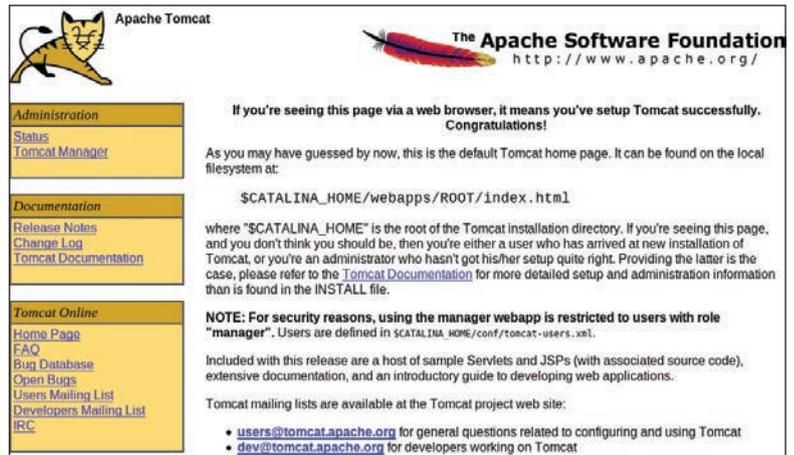
Далее вернитесь в каталог, где компилировали файл Java, и скопируйте его в должный каталог:

```
$ cp FormExample.class /var/lib/tomcat6/webapps/FormExample/WEB-INF/classes
```

(Кстати, вы ведь знаете, как пользоваться автодополнением, не так ли? Меня очень огорчит, если кто-то набирал всю строку вручную...)

Затем скопируйте форму в должный каталог. Это простой статический контент, и он размещается в каталоге **HelloWorld**, но вне каталога **WEB-INF**. Для доступа к нашему приложению мы просто обратимся к этому файлу.

```
$ cp formexample.html /var/lib/tomcat6/webapps/FormExample
```



➤ Домашняя страница *Tomcat* дает вам приятное ощущение того, что все работает, а также несколько полезных ссылок.

Наконец, скопируйте туда же дескриптор развертывания. Этот файл должен попасть в каталог **WEB-INF**:

```
$ cp web.xml /var/lib/tomcat6/webapps/FormExample/WEB-INF
```

Мы скопировали все файлы в должный каталог, и самый простой способ развернуть приложение — перезапустить сервер:

```
# service tomcat6 restart
```

Теперь (настал момент истины!) откройте <http://localhost:8080/FormExample/formexample.html>, введите свое имя в текстовое поле и нажмите на кнопку, чтобы отправить форму сервлету.

Можно сделать ссылку на форму под названием **index.html** таким образом:

```
$ cd /var/lib/tomcat6/webapps/FormExample
$ ln formexample.html index.html
```

Это позволит вам обращаться к приложению по более короткому адресу — <http://localhost:8080/FormExample>.

Если вы хотите создать чуть более динамичную страницу, вернитесь к исходному коду сервлета и добавьте строку

```
out.println("The time is " + new java.util.Date());
```

которая выведет текущее время и дату. Затем перекомпилируйте файл и скопируйте его в должное место в иерархии развертывания. Перезапустите сервер, и приложение будет развернуто.

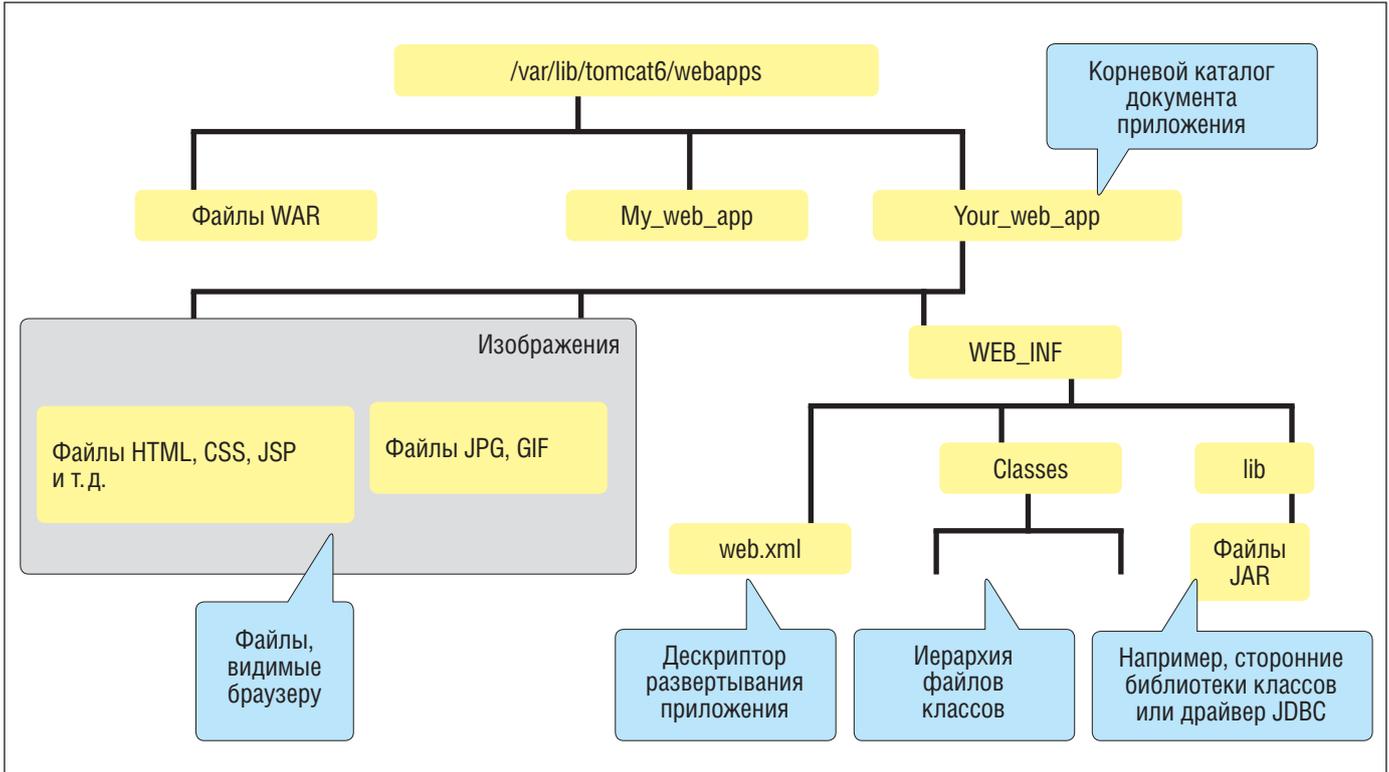
Менеджер

Перезапустить сервер для каждого разворачивания приложения — довольно brutальная вещь. Сейчас я покажу способ получше. В *Tomcat* есть приложение Менеджер (мы установили его как часть пакета *tomcat6-admin-webapps*), доступное на <http://localhost:8080/manager/html> или по ссылке «Tomcat Manager» на домашней странице. Однако прежде чем им воспользоваться, нужно кое-что настроить, потому что *Tomcat* даст доступ к этому приложению, только если вы аутентифицируетесь как пользователь — член роли 'manager', а в конфигурации по умолчанию такого пользователя нет. Чтобы это исправить, откройте файл **/etc/tomcat6/tomcat-users.xml** и добавьте туда такие строки:

```
<role rolename="manager"/>
<user username="tomcat" password="tomcatpw"
roles="manager"/>
```

Убедитесь, что добавляете эти строки в тэг **<tomcatusers>**. Имя пользователя не имеет значения, если он член роли *manager*. Перезапустите сервер еще раз и перейдите по ссылке 'Tomcat Manager' на домашней странице. Откроется окно менеджера веб-приложений *Tomcat* со списком всех запущенных приложений. Здесь можно остановить или запустить приложение, перезагрузить или свернуть его.

Это также можно делать с командной строки, заходя на соответствующие адреса менеджера. Например, чтобы получить



➤ Это стандарт! Web-приложения Java должны быть организованы в определенную структуру каталогов, показанную здесь.

список всех работающих приложений, можно запустить **wget** таким образом (я немного сократил вывод):

```
$ wget --user tomcat --password tomcatpw -O - http://localhost:8080/manager/list
OK - Listed applications for virtual host localhost
/:running:0:ROOT
/manager:running:6:manager
/sample:running:0:sample
/docs:running:0:docs
/examples:running:0:examples
/FormExample:running:0:FormExample
/host-manager:running:0:host-manager
/HelloWorld:running:0:HelloWorld
```

Другой пример — перезагрузим наше приложение *FormExample*:

```
$ wget --user tomcat --password tomcatpw -O - http://localhost:8080/manager/reload?path=/FormExample
OK - Reloaded application at context path /FormExample
```

WAR [война] (и мир)

Альтернатива развертыванию приложения в иерархию «распакованных» каталогов — создание единого файла WAR и развертывание его. Файл WAR (Web Application Archive — архив web-приложения) — просто файл JAR (Java Archive — архив Java) иерархии развертывания. Файлы JAR похожи на архивы tarball,

и синтаксис команды **jar**, которая их создает, почти полностью идентичен синтаксису **tar**.

Преимущество файлов WAR перед файлами tarball в том, что можно распространять и разворачивать целое web-приложение в одном файле. Один из недостатков — сложнее будет выполнять небольшие изменения (например, изменить всего одну строку кода в одном сервлете), потому что каждый раз после этого придется снова генерировать и развертывать файл WAR.

Давайте развернем наше приложение *FormExample* в виде файла WAR. Сначала создайте архив. Так как приложение уже разложено по необходимой структуре каталогов, это легко:

```
$ cd /var/lib/tomcat6/webapps/FormExample
$ jar cf ~/FormExample.war .
```

Здесь я создал архив в своем домашнем каталоге. Проверьте содержимое архива командой

```
$ jar tf ~/FormExample.war
```

Теперь удалите исходную структуру каталогов и перезапустите *Tomcat* (от имени root):

```
# rm -r /var/lib/tomcat6/webapps/FormExample
# service tomcat6 restart
```

Сейчас наше приложение *FormExample* уже не развернуто. Затем перейдите на домашнюю страницу *Tomcat* и на этой странице воспользуйтесь ссылкой для входа в приложение менеджера. Прокрутите страницу вниз до раздела Deploy [Развертывание]. Введите **/FormExample** в окне Context Path [Контекстный путь] и **/home/chris/FormExample.war** в поле WAR URL, затем нажмите Deploy [Развернуть]. Прокрутите страницу вверх и проверьте наличие сообщения, подтверждающего, что приложение было развернуто. (Если не получилось, проверьте права доступа к домашнему каталогу и к файлу и убедитесь, что они доступны для чтения пользователю tomcat.) Теперь приложение должно снова стать активным.

На этом пока все. Через месяц мы поработаем с конфигурационным файлом сервера, установим несколько фильтров для входа в систему и управления доступом и посмотрим, как спрятать *Tomcat* за более традиционным web-сервером вроде *Apache*. Увидимся! **LXF**

Соперники

Tomcat — отнюдь не единственный контейнер сервлетов. Альтернативы с открытым исходным кодом включают *Jetty* (проект, ныне поддерживаемый Eclipse Foundation) и *GlassFish* (исначально созданный Sun, а теперь коммерческий продукт от Oracle, доступный также в поддерживаемой сообществом открытой версии). Кроме того, есть *JBoss* (сейчас им владеет Red Hat), *Oracle Web Server* (который BEA WebLogic продала Oracle), сервер *Resin* от Caucho и *Websphere* от IBM.

Наши эксперты помогут вам с любым приложением Linux!



ЕВГЕНИЙ БАЛДИН
Подтвердивший
свою квалификацию
физик.

Debian для космического ноутбука

Спасем космос!
Открытое письмо Йона Тихого

В альтернативной кибердисциплине, посвященной выяснению, что же лучше — GNU/Linux или нечто произведенное в стенах большой и мягкой корпорации, постоянно всплывала информация, что вообще-то в космосе на международной орбитальной станции ноутбуки астронавтов работают под управлением закрытой операционной системы.

Но понимание современной ситуации добралось и до NASA, где наконец-то решили заменить ноутбуки на станции экземплярами с предустановленным дистрибутивом Debian GNU/Linux. Все, теперь GNU/Linux отсюда не вытравить уже никак. Пал еще один бастион. Хотя странно скорее то, что он так долго продержался: ведь в космосе нужна стабильная и надежная система, которую можно полностью контролировать.

В космосе теперь все будет хорошо, а на Земле пока еще много чего нужно будет сделать. С другой стороны, доверие к открытым проектам постоянно повышается, причем не только из-за идеологии или бесплатности, но из-за того, что они элементарно лучше.

11 мая 2013 года русский раздел Википедии прошел рубеж в миллион статей! Народная Википедия — самая крупная из когда-либо существовавших энциклопедий на русском языке. Она не только бесплатная и свободная — она во многих аспектах лучшая, и со временем трудами коллективного разума число таких аспектов будет только увеличиваться.

e.m.baldin@inp.nsk.ru

В этом месяце вы научитесь...



Командовать 56
Бен Эверард представляет интерфейс командной строки в руководстве к терминалу для начинающих.



Добывать музыку 60
Накопилась гора компакт-дисков, а к CD-плееру и подойти некогда? **Джон Лэйн** учит эти диски обдирать.



Спать на ходу 64
Ваш компьютер умеет отдыхать, даже когда не выключен. **Нейл Ботвик** пояснит, как это работает.



Витать в облаках 66
Роб Добози все свои вычисления проделывает в удаленных дата-центрах. При посредстве пары простых шагов вы тоже такое освоите.



Тестировать 70
Джоно Бэкон показывает, как заставить разрабатываемые приложения выполнять свои обязанности. Этак лучше, чем после слушать жалобы.



Делать чертежи 74
В вас забродили соки художественных наклонностей? Присоединяйтесь к **Никку Вейчу** — он исследует лидера программ векторной графики в Linux.



Избегать Dropbox 78
Джеймс Литтон синхронизирует свои файлы на всех своих компьютерах, не прибегая к услугам третьих лиц.



Развертывать сервер 82
Настройка сервера — штука весьма сложная, и **Маянк Шарма** предпочел просто взять да загрузить готовенький.



Вычислять параллельно 86
Михайл Остапкевич и **Евгений Балдин** поэлементно складывают два вектора под руководством хост-программы.



Сохранять данные 90
В беспокойстве за ценную информацию **Игорь Спири**н организует резервное копирование. И вам советует организовать.

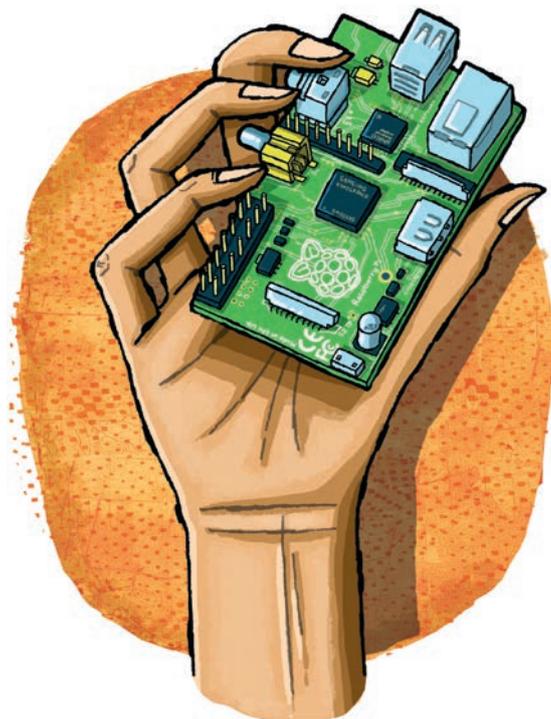
Raspberry Pi:

Бен Эверард показывает, как освоить интерфейс командной строки вашего Raspberry Pi и высвободить всю его мощь, даже не прикасаясь к мыши.



Наш эксперт

Бен Эверард бросил должность ИТ-консультанта, чтобы провести два года в Танзании, устанавливая в школах системы на базе Ubuntu. Теперь он применяет свои умения в бурлящем котле открытий, каковым являются Башни Linux Format.



Как вы, несомненно, уже обнаружили, у Raspbian имеется графический интерфейс, подобный интерфейсу Windows или Mac OS X. В нем вы можете выполнять большую часть своих повседневных задач. Здесь есть менеджер файлов, web-браузер, текстовый редактор и куча других полезных приложений. Однако иногда вам нужен интерфейс, который был бы немного мощнее, и вот тут-то и выходит на сцену Command Line Interface (CLI). Также он известен под названием Terminal [терминал] или Shell [оболочка].

Это чисто текстовый интерфейс, где вы печатаете команды и получаете ответ на них. Не будем вас обманывать: сначала

он кажется несколько запутанным. Но переживать не стоит: как только вы начнете практиковаться, он будет становиться все более понятным, и время, потраченное на его изучение, с лихвой окупится в будущем.

Первым делом надо открыть терминал. Щелкните по LXTerminal на рабочем столе Raspbian.

Тут появится строка примерно такого вида:

```
pi@raspberrypi ~ $
```

Это подсказка командной строки. Когда вы ее видите, вы понимаете, что система готова принимать команды.

Введите **pwd** и нажмите на кнопку Enter. Вы должны увидеть

```
/home/pi
```

Если вы изменили свое имя пользователя, то строка будет другой. Кажущаяся таинописью команда **pwd** означает Print Working Directory, и система просто выводит директорию, где вы в настоящий момент находитесь. Когда вы запускаете терминал, он перейдет в вашу домашнюю директорию (в разделе Raspbian Storage вы найдете больше информации о том, что означает **/home/**).

Теперь мы знаем, где мы находимся; следующим логическим шагом будет перемещение по директориям. Это делается с помощью команды **cd** (change directory). Попробуйте ввести:

```
cd ..
```

```
pwd
```

Вы должны увидеть, что система вернула **/home**. Это потому, что мы дали команду изменить директорию (**cd**) на **..**, а две точки всегда указывают на родительскую директорию. Чтобы вернуться назад в домашнюю директорию, можно ввести **cd pi**. Есть еще один способ сделать это. Знак **~** (тильда) всегда указывает на домашнюю директорию, поэтому где бы вы ни находились в файловой системе, вы можете ввести **cd ~** — и попадете к себе «домой».

Теперь наберите **ls** и нажмите на Enter. Выведется список всех файлов из текущей директории. Одно из огромных преимуществ команд в том, что мы можем сказать им, как именно они должны себя вести. Это делается с помощью флагов — они идут после команды и начинаются с **-**. Например, если нужно перечислить все файлы в текущей директории (включая скрытые файлы, которые в системах, основанных на Unix, начинаются с **.**), используете флаг **-a**. Поэтому в терминале введите **ls -a**.

На сей раз вы должны увидеть больше файлов. Еще один флаг для **ls** — это **-l**. Он дает нам больше информации о каждом файле. Попробуйте его, введя **ls -l**. Вы даже можете комбинировать флаги, как, например, в **ls -la**.

Учимся выбирать команды

Сейчас вы, вероятно, размышляете о том, откуда вам знать, какими командами и какими флагами надо воспользоваться для решения определенной задачи. Здесь есть хорошая новость и плохая новость. Хорошая новость в том, что выяснить, какие флаги имеются для команды, обычно не проблема. Большинство из них поддерживают флаги **-h** или **--help**, которые дадут вам информацию о флагах, принимаемых командой, и их применении. Например, запустив **ls --help**, вы увидите длинный список флагов и того, что они делают, в том числе:

Интерактивные программы

Большинство команд, с которыми мы здесь имеем дело, не интерактивны. Это означает, что вы запускаете их и ждете, когда они закончат работу. Однако так работают не все программы командной строки. Например, при первом запуске Raspbian запускался инструмент настройки, который работал в терминале. Есть несколько других программ, которые работают так же. Традиционно наиболее распространены текстовые редакторы, позволяющие работать с файлами, не имея графического соединения. Иногда

они довольно сложны, но окажутся просто превосходными, если вы проводите много времени, работая в командной строке; правда, освоить их трудно. Есть также простой в использовании текстовый редактор для терминала под названием *nano*. Чтобы запустить его, в командной строке введите **nano**, а далее должно следовать имя файла. Затем можете работать с текстовым файлом и делать любые нужные вам изменения. Нажмите **Ctrl+X**, чтобы сохранить результаты своей работы, и вернуться в оболочку.

ОСВОИМ КОМАНДЫ

-a, --all не игнорировать записи, начинающиеся с .

...

-l использовать формат длинного списка

Второй способ получить информацию по команде — набрать **man**. Это сокращение от manual [руководство]. Он использует один аргумент, то есть слово после команды, перед которым нет дефиса. Затем он отображает информацию по команде, введенной в качестве аргумента. Чтобы посмотреть man по **ls**, введите **man ls**. Можете перемещаться по страницам с помощью клавиш со стрелками или клавиш Page Up и Page Down, чтобы прокручивать их быстрее. Чтобы найти в справочнике man слово или фразу, введите /, а затем нужную фразу. Например, /-l найдет все совпадения для -l. Можете использовать клавишу N и Shift+N, чтобы перемещаться вперед и назад по вхождениям -l.

По мере знакомства с большим количеством команд было бы полезно заглядывать в **help** и в **man**, чтобы узнать, что они делают. Конечно, вы всегда можете получить информацию по команде в Google, если сочтете текстовую подсказку немного неудобной, однако если вы останетесь в терминале, это поможет вам лучше его освоить.

Как же мне узнать?..

Помните, мы сказали, что есть хорошая новость и плохая новость? Что ж, плохая новость в том, что не зная, как называется нужная команда, найти ее будет весьма хитро. Одной из удобных функций является поиск по ключевому слову **man**. Он осуществляется с помощью флага **-k**.

Чтобы найти в вашей системе все программы, относящиеся к 'browser', запустите **man -k browser**. Вы увидите, что в списке будут перечислены не только программы командной строки, но и графические программы. Это потому, что между ними на самом деле нет разницы. Можно запускать окна из терминала и даже управлять ими.

Если у вас есть *Iceweasel* (ребрендинговая версия *Firefox*) на вашем Pi (по умолчанию она там не установлена), вы можете открыть TuxRadar.com в новой вкладке в открытом окне *Iceweasel* с помощью команды **iceweasel --new-tab www.tuxradar.com**.

Теперь мы быстренько познакомим вас с несколькими полезными командами. **rm** удаляет (ReMoves) файл. **mkdir** создает



➤ В командной строке даже фильмы можно смотреть. Чтобы начать воспроизведение этой классики, просто введите **telnet towel.blinkenlights.nl**, и не забудьте прихватить попкорн.

Автозаполнение

Когда имеешь дело с длинными именами файлов, надоедает каждый раз набирать эти названия, если нужно применить к ним команду. Чтобы отчасти облегчить жизнь, терминал использует автозаполнение.

Это означает, что если вы начинаете набирать имя файла и нажимаете на клавишу Tab, система попытается автоматически дополнить остальное имя. Если есть только один файл, соответствующий тому,

что вы уже набрали, то остальная часть имени будет заполнена автоматически (попробуйте набрать **cd /h** и затем нажать на Tab).

Если совпадений более одного, остаток имени будет заполнен автоматически настолько, насколько они совпадают. Если снова нажать Tab, вам покажут возможные опции (попробуйте набрать **cd /m** и затем дважды нажать на Tab).

новую директорию. **cp** копирует файл из одного места в другое. Ему нужно два аргумента: исходный файл и новый файл. **cat** выводит содержание одного или более текстовых файлов. Он может принять сколько угодно аргументов, каждый для одного текстового файла, и передать их содержимое в терминал.

less — более дружелюбный способ просмотра текстовых файлов. Он позволяет прокручивать файл вверх-вниз с помощью клавиш со стрелками. Чтобы выйти из программы и вернуться в командную строку, нажмите на Q. Мы будем использовать все эти команды в примерах ниже, так что задерживаться на них не будем.

find — полезная команда для поиска файлов на вашем компьютере. Вы используете ее в формате **find <где искать> <флаги>**. Например, чтобы найти все файлы на вашем компьютере, которые были изменены за день до этого, запустите

```
find / -mtime 1
```

На стр. 59 вы найдете более подробную информацию о том, что это значит, и о разных флагах, которые можно использовать.

Больше мощи

Вы можете подумать: «Все это отлично делается через графический интерфейс, и мне незачем запоминать всякие мутные команды и флаги». И вы будете правы — но лишь потому, что мы пока не познакомим вас с более мощными функциями. Первая из них — шаблоны. Это символы, которые вы можете вставлять и которые соответствуют другим символам. Звучит довольно непонятно, но мы попытаемся пояснить это на примерах.

Во-первых, мы создадим новую директорию, переместимся в нее и создадим несколько пустых файлов (мы используем команду **touch**, которая создает пустой файл с именем каждого аргумента). Подсказка — вы можете использовать автозаполнение (см. врезку сверху), чтобы не набирать постоянно длинные названия, как, например, во второй строке.

```
mkdir wildcards
```

```
cd wildcards
```

```
touch one two three four
```

```
touch one.txt two.txt three.txt four.txt
```

Затем запустим **ls**, чтобы увидеть, какие файлы есть в новой директории. Вы должны увидеть восемь.

Первый шаблон, который мы будем использовать — *****. Он соответствует любой строке от нуля до большого количества

символов. В простейшем случае он соответствует любому файлу в директории. Попробуйте запустить:

```
ls *
```

Пока это не особо удобно, но мы можем поставить его среди других символов. Как вы думаете, чему будет соответствовать `*.txt`? Подумайте, а потом запустите:

```
ls *.txt
```

чтобы проверить правильность ваших выводов. Как насчет `one*`? Снова запустите

```
ls one*
```

чтобы увидеть, правы ли вы. Шаблоны можно использовать с любой программой командной строки. Особенно они удобны для сортировки файлов. Чтобы скопировать все текстовые файлы в новую директорию, запустите

```
mkdir text-files
```

```
cp *.txt text-files
```

Затем проверим, правильно ли это сделано, командой

```
ls text-files/
```

Шаблон: ?

Второй шаблон, который мы рассмотрим, это `?`. Он соответствует любому одиночному символу. Как вы думаете, чему будет соответствовать

```
ls ???
```

Попробуйте догадаться, а затем запустите команду, чтобы проверить правильность своей догадки.

Мы также можем сами создавать шаблоны, соответствующие символам, которые нам нужны. `[abc]` будет соответствовать маленьким буквам А, В и С. Как вы думаете, чему будет соответствовать `ls [ot]*`? Теперь попробуйте запустить

```
ls [!ot]*
```

Что нового внес восклицательный знак? Он должен был вернуть все, что *не* начинается с маленьких букв О или Т.



► Используйте команду `sudo`, чтобы переключиться с обычного пользователя `pi` на суперпользователя `root`.

Все команды, которые рассмотрели, выводили отображение результата в терминал. По большей части нам именно это и надо, но иногда полезно сделать с ним что-то другое. В Linux можно сделать две других вещи: отправить результат в файл или в другую программу.

Чтобы отправить его в файл, используйте символ `>`, за которым следует имя файла. Запустите

```
ls > files
```

```
cat files
```

и вы должны увидеть, что создан новый файл под названием `files`, и в нем содержится результат команды `ls`.

Вторая опция, возможность отправить результат в другую программу, являет собой еще одну действительно мощную функцию командной строки Linux, поскольку она позволяет соединить несколько команд в цепочку, сделав из них одну супер-команду.

Есть множество команд, работающих с текстом и предназначенных для работы таким способом. Они не относятся к теме нашего урока, но если вы будете продолжать использовать командную строку, вы обязательно с ними столкнетесь и начнете понимать, как их можно соединить вместе.

Рассмотрим очень простой пример. Если вы запустите `find /` (но пока что не делайте этого!), вам выведется список всех файлов системы.

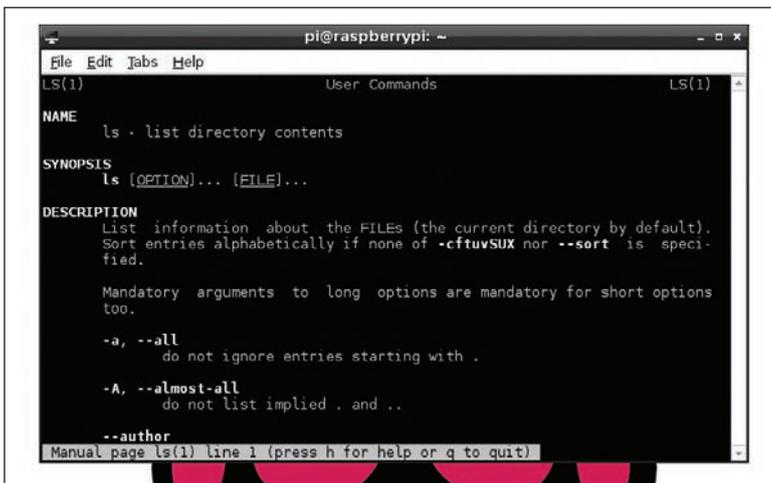
У вас получится длинейший список из файловых имен, которые будут быстро бежать по экрану. Однако вместо того, чтобы выводить его на экран, мы можем отправить его в другую команду, которая сделает результат более удобным для чтения. Мы можем использовать для этого команду `less`, которую рассматривали ранее. Запустите:

```
find / | less
```

Идем дальше

Мы смогли коснуться только основ использования командной строки, но для начала вам этого должно хватить, и будем надеяться, что вы увидите, насколько мощен интерфейс командной строки, когда начнете его понимать.

Если вы хотите узнать больше (а это следует сделать!), вам в помощь есть множество ресурсов, как в печатном виде, так и онлайн. Очень хорошим стартовым пунктом станет www.linuxcommand.org. Их книгу (The Linux Command Line) можно приобрести в книжных магазинах, или найти бесплатно онлайн (www.linuxcommand.org/lc3_learning_the_shell.php).



► `man` — вероятно, самая важная команда в любой Unix-подобной системе, поскольку это ключ к пониманию любой другой команды. Не пожалейте времени на ознакомление с используемой структурой и языком: в будущем это окупится.

Только Sudo'm

Используя Raspberry Pi повседневно, вы можете работать с файлами в своей домашней директории (например, `/home/pi`). Вы также сможете просмотреть большую часть файлов в системе, но не сможете изменить их, и не сможете установить программы. Все потому, что в Linux есть система прав

доступа, не позволяющая обычному пользователю менять системные настройки. Это предотвращает случайную порчу настроек. Однако бывают случаи, когда вам нужно это сделать. Вы можете использовать `sudo` для запуска команды от имени суперпользователя (иногда его называют `root`), у которого

есть право делать с системой практически все, что угодно. Чтобы воспользоваться этим, поставьте перед командной `sudo`. Например, `sudo apt-get install synaptic` установит пакет `synaptic` и сделает его доступным для всех пользователей.

Справочник LXF по командам — вырежи и запомни!

Навигация и файлы

- » **cd** Change directory — Смена директории. Например, **cd movies** переходит в папку **movies**. **cd ~** переходит в домашнюю директорию, **cd /** переходит в директорию **root**, а **cd ..** переходит на одну директорию назад.
- » **ls** List files — Вывод списка файлов. Сама по себе она выводит файлы из текущей директории. **ls movies** перечисляет файлы в директории **movies**. **ls -a** перечисляет все файлы (в том числе скрытые), а **ls -l** выдает более подробную информацию о каждом файле.
- » **cp** Copy files — Копирование файлов. **cp orig-file new-file** копирует **orig-file** в **new-file**.
- » **wget** Загружает файл из Интернет. Чтобы скачать домашнюю страницу Google в текущую директорию, используйте **wget www.google.com**.
- » **df -h** Отображает количество свободного места на устройстве.
- » **pwd** Отображает текущую директорию.

Поиск файлов

- » **find <location> <flags>** среди полезных флагов: **-mtime <number>** находит файлы, измененные за последние **<number>** дней. **<number>** может быть, например, 2 (ровно два дня назад), -2 (менее двух дней назад) или +2 (более двух дней назад).
- » **-name <filename>** находит файлы под названием **<filename>**.
- » **-iname <filename>** соответствует файлам с именем **<filename>**, регистр букв (строчные или заглавные) неважен.
- » **-writable** находит записываемые файлы. Опций намного больше. Более подробный их список вы найдете в справочнике **man**. Например, **find / -mtime -2 -writable** находит все файлы в системе, которые были изменены менее двух дней назад и могут быть записаны текущим пользователем.

Удаленная работа

- » **ssh** Войдите на удаленный компьютер с помощью Secure Shell (SSH протокол). **ssh pi@192.168.1.2** войдет в качестве пользователя **pi** на компьютер с IP-адресом **192.168.1.2**. Помните: это будет работать, только если на удаленном компьютере работает сервер SSH.
- » **scp** Secure CoPy. **scp file pi@192.168.1.2:/home/pi** скопирует файл в директорию **home/pi** на компьютере с **192.168.1.2**. **scp pi@192.168.1.2:/home/pi/file** скопирует файл **home/pi/file** с машины **192.168.1.2** в текущую директорию. Помните: это будет работать, только если на удаленной машине работает сервер SSH.

Шаблоны

- » * Соответствует любой строке символов, или никакому символу.
- » ? Соответствует любому единичному символу.
- » [abc] Соответствует a, b или c.
- » [abc] Соответствует любому символу, кроме a, b или c.
- » [A-Z] Соответствует любому символу в диапазоне A–Z (т.е. любой заглавной букве).
- » [a-z] Соответствует любому символу в диапазоне A–z (т.е. любой заглавной или маленькой букве).
- » [one, two] Соответствует словам one и two.

Информация о компьютере

- » **top** Отображает программы, в данный момент потребляющие наибольшее количество тактов CPU и памяти.
- » **uname** Отображает информацию о ядре. **uname -m** выведет информацию об архитектуре, на которой работает.
- » **lscpu** Выводит информацию о CPU.

- » **dmesg** Отображает сообщения ядра (может пригодиться для обнаружения проблем с оборудованием).

Текстовые файлы

- » **head** Отображает первые десять строк текстового файла. Замените десять на любое другое число с флагом **-n**. Например, **dmesg | head -n 15** отобразит первые 15 строк журнала сообщений ядра.
- » **tail** Отображает последние десять строк текстового файла. Может использовать флаг **-n**, как и с **head**, а также отслеживать изменения в файле с помощью флага **-f** (follow). Например, **tail -n15 -f /var/log/syslog** отобразит последние 15 строк файла журнала системы, и продолжит это делать по мере изменения файла.
- » **less** Позволяет проматывать текстовый файл.
- » **cat** выводит содержание текстового файла в терминал.
- » **nano** Дружелюбный к пользователю текстовый редактор командной строки (Ctrl+X позволяет выйти и сохранить изменения).

Специальные клавиши

- » **Ctrl+C** Убивает любую программу, работающую в терминале.
- » **Ctrl+D** Отправляет признак конца файла в любую программу, работающую в терминале.
- » **Ctrl+Shift+C** Копирует выделенный текст в буфер обмена.
- » **Ctrl+Shift+V** Вставляет текст из буфера обмена.

Установка программ

- » **tar xzvf file.tar.gz**
- » **tar xjf file.tar.bz**
- » **./configure** Когда вы распаковываете исходный код программы, обычно создается новая директорию с программой внутри нее. Войдите в эту директорию с помощью **cd** и запустите **./configure**. Ваша система будет проверена на предмет наличия в ней всего необходимого для компиляции программ.
- » **make** Эта команда скомпилирует программу.
- » **make install** (требуется **sudo**) Эта команда переместит вновь скомпилированное ПО в соответствующее место в вашей системе, чтобы вы могли запускать его, как обычную команду.
- » **apt-get** Эту команду можно использовать для установки и удаления программ.

Например, **sudo apt-get install iceweasel** установит пакет **iceweasel** (ребрендинговая версия **Firefox**). **sudo apt-get purge iceweasel** удалит пакет. **apt-get update** скачает обновленный список пакетов из репозитория (неплохая идея; всё бы так делалось!). **apt-get upgrade** обновит все пакеты, у которых есть более новая версия в репозитории.

- » **apt-cache search <keyword>** будет искать в репозитории все пакеты, соответствующие ключевому слову. **LXF**

```

ben@chrispad:~/kickstarts
File Edit View Search Terminal Help
top - 10:50:26 up 10 days, 20:18, 2 users, load average: 0.14, 0.05, 0.05
Tasks: 166 total, 1 running, 165 sleeping, 0 stopped, 0 zombie
%Cpu(s): 2.0 us, 0.8 sy, 0.0 ni, 94.4 id, 2.8 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem: 2055372 total, 1399612 used, 655760 free, 76960 buffers
KiB Swap: 10311332 total, 170304 used, 10141028 free, 957332 cached

  PID USER      PR  NI  VIRT  RES  SHR  S   %CPU  %MEM    TIME+  COMMAND
12724 ben       20   0   657m 163m 12m  S   2.6   8.2   102:18.50 gnome-shell
12414 root      20   0   81824 8764 2772  S   1.3   0.4   15:17.24 Xorg
30707 ben     20   0   177m 7108 4620  S   1.0   0.3   0:39.43 gnome-terminal
1724  root      20   0   0      0     0    S   0.3   0.0   0:09.12 kworker/1:1
12523 ben     20   0   5040 1484 632   S   0.3   0.1   0:00.89 dbus-daemon
12692 ben     20   0   180m 5152 3124  S   0.3   0.3   0:28.73 gnome-settings-
19013 root      20   0   0      0     0    S   0.3   0.0   0:02.14 kworker/0:1
21343 ben     20   0   6424 1288 956   R   0.3   0.1   0:00.02 top
1  root      20   0   6500 2224 1280  S   0.0   0.1   0:03.20 systemd
2  root      20   0   0      0     0    S   0.0   0.0   0:00.06 kthreadd
3  root      20   0   0      0     0    S   0.0   0.0   0:02.06 ksoftirqd/0
5  root      0 -20  0      0     0    S   0.0   0.0   0:00.00 kworker/0:0H
7  root      0 -20  0      0     0    S   0.0   0.0   0:00.00 kworker/u:0H
9  root      0   0   0      0     0    S   0.0   0.0   0:04.67 migration/0
9  root      20   0   0      0     0    S   0.0   0.0   0:00.00 rcu_bh
10  root      20   0   0      0     0    S   0.0   0.0   0:25.23 rcu_sched
11  root      0   0   0      0     0    S   0.0   0.0   0:03.00 watchdog/0

```

» Используйте команду **sudo**, чтобы переключиться с обычного пользователя **pi** на суперпользователя **root**.

Оцифруйте СВОЮ

Со стопкой CD и новым музыкальным плеером, Джон Лэйн постигает технологию, обеспечивающую звук, и учится оцифровке прямо в Linux...



Наш эксперт

Набредя на Linux в далеком 1994, Джон Лэйн так в нем и застрял, и уже не упомнит, когда последний раз пользовался Windows. Он пишет и консультирует о Linux и открытым ПО.



Кончина последнего крупного — некогда процветающего — музыкального магазина показывает, что большинство людей ныне покупают музыку в Сети; но хотя скачивание — это современно, быстро и весело, многие из нас до сих пор предпочитают проверенных служителей музыки: компакт-диски. Но тут встает проблема — как перенести наши люби-

мые мелодии туда, где мы их чаще всего проигрываем: на медиаплееры, будь то компьютеры, телефоны или другие цифровые носители.

Способствовать переезду вашей подборки CD на жесткий диск готовы немало инструментов. На нашем уроке мы рассмотрим некоторые из них и обсудим способы создания полноценной фонотеки. Мы объясним, как правильно импортировать музыку, чтобы она проигрывалась на всех основных музыкальных плеерах, включая самый популярный, хоть и проприетарный — iTunes.

При оцифровке музыки выполняется три основных шага. Первый — импортирование, или «обдирание [ripping]», то есть копирование музыки с диска в файл данных на вашем компьютере; затем кодирование — преобразование в подходящий формат; и, наконец, добавление тэгов метаданных, описывающих содержимое. Большинство инструментов риппинга выполняют все три действия, и многие думают, что это один процесс.

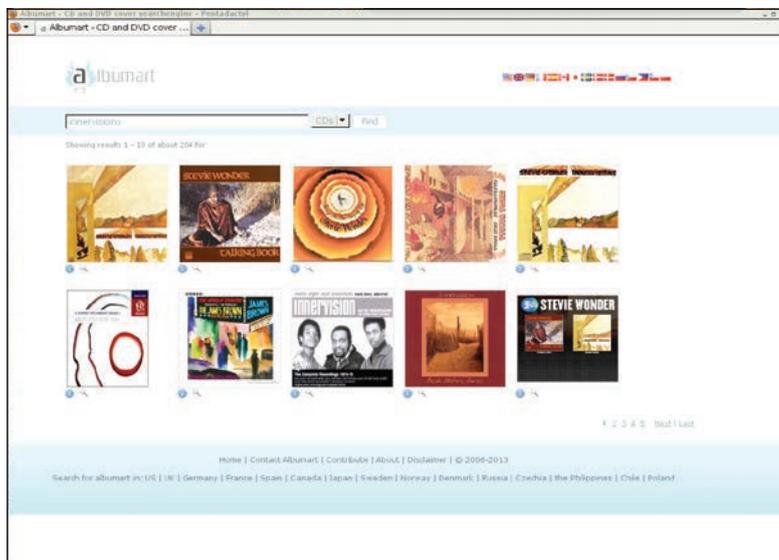
Скрытая информация

На большинстве CD нет той информации, которую хотелось бы видеть во время воспроизведения: об исполнителе, альбоме, названии трека. В качестве решения, в середине 1990-х была создана база данных под названием CDDB. Пополнять ее сведениями мог кто угодно, и многие так и делали. Хотя CDDB была открытой и действовала под GPL, сам сервис был приватизирован; сегодня он называется Gracenote и принадлежит Sony. Несмотря на то, что база была продуктом общих усилий, она более не является открытой и доступна только по лицензии. В качестве альтернативы был создан другой сервис, FreeDB — общедоступная база данных, защищенная GPL.

На более новых дисках иногда содержится основная информация о треках, благодаря использованию расширения базового CD-формата (известного как спецификация Red Book) — именуемого CD-Text. Предоставляемые им сведения могут быть хорошим подспорьем при отсутствии информации во FreeDB, однако они могут быть неполными или доступными не для всех дисков.

В целом, метаданные во FreeDB довольно качественные и верные, но иногда случаются расхождения и ошибки. И если вы хотите, чтобы ваша музыка была правильно организована и подписана, придется вручную проверить данные по каждому диску.

Обязательной частью метаданных теперь являются обложки альбомов. Большинство медиа-плееров отображают эти сведения, но во FreeDB (как и в Gracenote) их нет. На самом деле, нет какого-то одного надежного источника графических данных по альбомам, но есть много ресурсов, где можно найти не только их, но и приложения для работы с медиа, которые помогут их разместить (iTunes, например, получает их из iTunes Store). Также можно использовать Google (через поиск картинок) и Amazon. Движок поиска изображений из альбомов на данном ресурсе доступен и на www.albumart.org; можно найти и другие. Поскольку все они не идеальны и иногда выдают не те изображения, необходимо опять же проверять их, чтобы поддерживать порядок в вашей фонотеке.



» Ищите обложки альбомов на Amazon через albumart.org.

КОЛЛЕКЦИЮ CD

Некоторые медиа-приложения и проигрыватели не справляются с графическими данными в альбомах. Хотя они могут (и должны) быть встроены непосредственно в музыкальные файлы, кое-где они хранятся в проприетарных базах данных (да, это про iTunes!). Или же приложения ищут в папках файл с именем `folder.jpg` или `cover.jpg`. Лучше, конечно, хранить метаданные по изображениям внутри самих файлов (так они вряд ли потеряются, а большинство плееров их читает), но следует учесть и то, насколько при этом увеличится размер самого файла и, как следствие, всего альбома — едва ли имеет смысл хранить огромное изображение, если в медиа-плеере отображается лишь мелкая иконка.

В каком формате?

Наверняка все знают об MP3, ведь это самый распространенный формат оцифровки музыки. Название MP3 — аббревиатура от MPEG-1 (или 2) Audio Layer III, и это общепринятый стандарт кодирования аудиоданных. Не путайте MP3 и MPEG-3 — это совершенно иной стандарт, он предназначен для видео!

MP3 — проприетарный формат, покрытый патентами нескольких компаний. Вы можете обнаружить, что в вашем любимом дистрибутиве Linux он не устанавливается по умолчанию, из-за патентных ограничений, однако можно установить соответствующий кодек самим из неофициальных репозиториях. Альтернативный формат — Ogg Vorbis, с ним таких проблем не возникает. Если ваши проигрыватели его поддерживают, возможно, стоит рассмотреть этот вариант, вместо MP3.

MP3 и Ogg Vorbis — сжатые форматы с потерями, т.е. в процессе преобразования часть данных теряется при сжатии, для уменьшения размера файла. Речь идет о звуковых частотах, недоступных для человеческого уха, и среди меломанов ведутся бурные споры о последствиях. Вы можете контролировать объем пропускаемых данных при помощи изменения битрейта — чем он выше, тем меньше потери, но больше сам файл.

Форматы без потерь, напротив, ничего не утрачивают и, как следствие, занимают гораздо больше места. Если место у вас не в дефиците, стоит сначала конвертировать в формат без потерь, а затем с потерями — для тех устройств, где это требуется. Преобразование из одного формата в другой называется транскодированием.

WAV, или Waveform Audio File Format — формат без потерь для хранения несжатого аудио, и, следовательно, самый объемный. Обычно при импорте создаются как раз WAV-файлы, которые затем конвертируются в более экономный формат. Но при желании их можно оставить и в исходном виде.

Есть также форматы без потерь, способные сжимать данные и получать файлы меньшего размера (но не в той мере, как форматы с потерями). В мире открытого ПО одним из таких вариантов выступает FLAC, Free Lossless Audio Codec, не ограниченный па-

Новый Opus

В сентябре прошлого года Mozilla опубликовала для обсуждения анонс RFC 6716. В нем описывался новый кодек Opus, «предназначенный для широкого спектра интерактивных аудиоприложений, включая IP-телефонию, видеоконференции,

внутриигровой чат и даже живые музыкальные трансляции. Он поддерживает передачу от узкополосных голосовых сигналов с низким битрейтом, на 6 кбайт/с, до высококачественной стереомысли на 510 кбит/с».

тентом кодек под лицензией BSD. В числе прочих можно отметить ALAC, Apple Lossless Audio Codec, ставший открытым в 2011 году.

Первое, что вы должны решить — будете ли вы экономить место на диске ценой снижения качества записи: нужен ли вам формат с потерями или без потерь? Скорее всего вы остановитесь на MP3 — он пока что самый поддерживаемый. Если хотите попробовать и другой вариант, выбирайте FLAC.

Формат MP3 содержит данные о файле, преобразованном с помощью одноименного кодека. Кодек — это слово-комбинация из «кодирования» и «декодирования»; им обозначаются алгоритмы данных процессов применительно к аудио- или видеоданным. То, что сейчас принято называть кодеком, имеет отношение только ко второму процессу и необходимо для воспроизведения файла.

С другими форматами все не так просто: они известны как контейнеры или метафайлы. Значительная часть преобразованного аудио (да и видео, в данном случае) представляет собой метафайлы. В файлах подобных специальных форматов содержатся



Скорая помощь
Список официальных тэгов находится на <http://id3.org/id3v2.4.0-frames>.

```

Music
[music]$ eyeD3 -v 01.Superfly.mp3
01.Superfly.mp3 [ 5.75 MB ]

Time: 03:54  MPEG1, Layer III  [ ~203 kb/s @ 44100 Hz - Joint stereo ]

ID3 v2.4:
title: Superfly  artist: Curtis Mayfield
album: Beautiful Brother - The Essential Curtis Mayfield  year: 2000
track: 1/15  genre: Soul (id 42)

OTHER Image: [Size: 51301 bytes] [Type: image/jpeg]
Description: cover.jpg

-----
ID3 Frames:
<Lead performer(s)/Soloist(s) (TPE1): Curtis Mayfield>
<Album/Movie/Show title (TALB): Beautiful Brother - The Essential Curtis Mayfield>
<Title/songname/content description (TT2): Superfly>
<Track number/Position in set (TRCK): 01/15>
<Content type (TCO): (42)Soul>
<Release time (TDRL): 2000>
<Attached picture frame (APIC)>
<Tagging time (TDIG): 2013-02-21T21:44:18>
[music]$
  
```

» EyeD3 — лучший инструмент для полного контроля над вашими тэгами ID3.

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

различные связанные потоки данных, которые можно кодировать по-разному. Например, видео-метафайлы часто содержат видеопоток плюс один или несколько аудиопотоков. Другие метафайлы организованы как единый поток, кодированный особым образом; самый расхожий пример — WAV-файл, содержащий несжатое аудио в виде RIFF-потока. То же и с Ogg Vorbis — название отсылает к лежащему в его основе контейнеру Ogg, с аудиоданными, полученными методом сжатия с потерями Vorbis. Ogg может также включать и сжатое без потерь аудио (FLAC) и видео (Theora). Мы рассмотрим кодирование в MP3, FLAC и Ogg Vorbis.

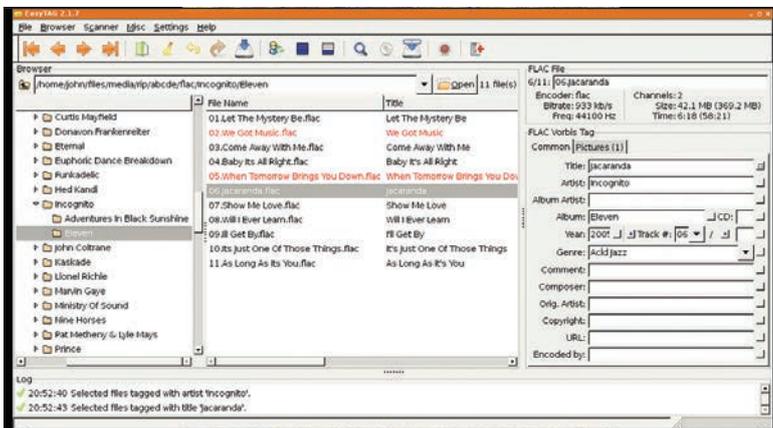
Подходящие инструменты

Linux располагает множеством инструментов и графических приложений для оцифровки музыки. В большинстве из них для импорта, кодирования и индексирования используются абсолютно одни и те же утилиты командной строки. Мы испытаем инструмент под названием *abcde* (сокращенное от A Better CD Encoder). Это приложение, работающее в терминале (или, если быть точными, скрипт *Bash*), которое объединяет эти утилиты и делает их прозрачными и доступными — вы наверняка найдете его в репозиториях своего дистрибутива. Также мы опробуем приложение *EasyTag* для управления тэгами метаданных внутри преобразованных музыкальных файлов. Это небольшое приложение на GTK+, которое должно без проблем работать в большинстве настольных систем. Временами нам придется редактировать тэги, которые *EasyTag* не поддерживает; тогда мы будем использовать утилиты командной строки напрямую.

Установите приложения и требуемые зависимости, через свой менеджер пакетов:

```
(ubuntu) $ sudo apt-get install abcde easytag
(fedora) $ sudo rpm -i abcde easytag
(arch) $ sudo pacman -S abcde easytag
```

➤ **EasyTag** позволяет быстро редактировать тэги в нескольких файлах.



Идентификатор диска

Идентификатор диска (ID), используемый FreeDB, носит имя CDDDB1, чтобы не путать с CDDDB2 от Gracenote. Это 32-битное число, обычно представленное как восьмизначное шестнадцатеричное. Оно вычисляется при помощи алгоритма, учитывающего время начала каждого трека и общую продолжительность всего диска, и не является уникальным, поэтому запросы на его основе часто дают неверный результат. Чтобы узнать его, воспользуйтесь утилитой командной строки (это будет первое значение вывода):

```
$ cd-discid /dev/cdrom
d1117c0e 14 150 2222 413...
```

Сперва выводится ID; далее указано количество треков, время начала каждого и общая продолжительность CD в секундах.

Мощью приложения *abcde* обусловлена его сложность. Его бесчисленные аргументы унаследованы от всех утилит, которые оно вобрало. К счастью, в нем есть файл настроек, который можно применить и забыть, не мучаясь со всеми параметрами. Мы создали пример такого файла, который подойдет для импорта компакт-диска с параллельным преобразованием в MP3, FLAC и Ogg Vorbis. Сохранив файл настроек как *~/.abcde.conf*, можно ставить диск в привод, импортировать данные, кодировать в любой из трех форматов и добавлять данные из FreeDB всего одной простой командой:

```
$ abcde
```

Вид преобразованных треков для большинства плееров значения не имеет — они опираются на тэги метаданных внутри файлов. Однако привести его к единообразию весьма полезно. Наша конфигурация импортирует диск в *~/music/<format>/<artist>/<album>*, записав каждый трек в файл вида *<track number>, <name>, <format>*.

Номера треков записываются в двузначных числах (по необходимости, с добавлением нуля), чтобы они всегда были в правильном порядке. Все это прописано в *~/.abcde.conf*, а если вам нужна другая структура, можете поменять соответствующие настройки — ищите **OUTPUTDIR**, **OUTPUTFORMAT**, **VAOUTPUTFORMAT** и **PADTRACKS**.

Что такое тэги

Говоря о тэгах, мы имеем в виду метаданные, которые хранятся внутри преобразованных файлов и описывают их содержимое. Способ их применения в разных контейнерах варьируется от гибкой спецификации Vorbis Comment, во FLAC и Ogg Vorbis, до более строгой структуры ID3-тэгов, в файлах MP3.

Общепринятые тэги

Тэги, используемые различными медиа-приложениями:

Тэг	Название тэга
TPE1	Основной/(ые) исполнитель/(и)
TPE2*	Группа/оркестр/сопровождение
TPE3	Уточнение к имени дирижера/исполнителя
TALB	Название альбома, фильма или шоу
TIT2	Название произведения/описание контента

TRCK	Номер дорожки / позиции в альбоме
TPOS	Номер части альбома
TCMP*	Обозначение сборника в iTunes
TCON	Жанр музыки
TDRL	Дата выхода
TYER	Год выхода
TPE2	Группа/оркестр/сопровождение
TCOM	Композитор
APIC	Обложка

TDTG	Сведения о добавлении тэгов
COMM	Комментарии
TENC	Кем закодировано
UFID	Уникальный идентификатор диска
TPE2	обычно читается как Основной исполнитель, несмотря на то, что озаглавлен как Группа/оркестр/сопровождение, а TSPM используется в iTunes для обозначения Сборника; это не общепринятый тэг ID3.

➤ **Пропустили номер?** Узнайте на с. 108, как получить его прямо сейчас.

Метаданные ID3 определяются неофициальным стандартом, опубликованном на id3.org и называемом ID3v2 (там есть и более старый стандарт — ID3v1, но его можно спокойно игнорировать). ID3v2 существует в двух вариантах: версия 2.3 — основной вариант, а 2.4 — расширенный, с обратной совместимостью (правда, некоторые плееры на нем спотыкаются). Файлы FLAC могут помимо собственных тэгов содержать тэги ID3, но такое встречается сравнительно редко.

Добавляя в музыкальные файлы общепринятые тэги, о нюансах разных форматов можно не задумываться — поддерживаемый набор стандартен, отличается только сам метод.

Общепринятыми называют тэги, распознаваемые большинством приложений. Наша конфигурация *abcde* пользуется информацией из FreeDB и автоматически прописывает номер трека, название, жанр, имя исполнителя и название альбома — плюс номер диска для многодисковых альбомов. Для альбомов одного исполнителя этого достаточно; со сборниками немного сложнее. Для их правильного отображения в некоторых приложениях (в том же iTunes) понадобятся добавочные тэги.

Сборники

Треки из сборника должны быть индексируются как входящие в его состав, а если он включает несколько исполнителей — закреплены за каждым из них.

Увы, *abcde* ни того, ни другого не умеет, поэтому мы обратимся к более низкоуровневым утилитам командной строки: *id3tag* для файлов MP3, *metaflac* для файлов FLAC и *vorbiscomment* для файлов Ogg Vorbis. С их помощью можно составлять список, добавлять, изменять или удалять тэги в одном или многих файлах; можно управлять всем набором тэгов с id3.org, а также многими неофициальными, и даже создавать собственные.

Вот примеры создания списка тэгов в MP3, Ogg Vorbis и FLAC файлах:

```
$ eyeD3 -v 01.Superfly.mp3
```

```
$ vorbiscomment 01.Superfly.ogg
```

```
$ metaflac --list 01.Superfly.flac
```

При отсутствии специальных тэгов сборники могут во многих приложениях вызвать проблему, поскольку те руководствуются именем исполнителя и названием альбома. Без спецобработки альбом из десяти треков разных исполнителей будет читаться как десять разных альбомов, что внесет смуту в вашу упорядоченную фонотеку. Решением станет добавление двух тэгов: Основной Исполнитель (тэг TPE2) и неофициальный тэг Сборник (TCMP).

Сделать это можно из командной строки:

```
$ eyeD3 --set-text-frame=TABC:mytagvalue myfile.mp3
```

Улучшенная abcde?

Abcde не предусматривает способа добавить к трекам тэг Основной Исполнитель, а мы выяснили, что для многих плееров это важно. Также здесь не поддерживаются изображения. Но есть расширенная

версия, которая позволяет добавлять тэги в сборники одного и нескольких исполнителей, подписывать жанр и вставлять графические элементы (обложку альбома): <http://tinyurl.com/atvj4t>.

```
$ vorbiscomment -a --tag 'MYTAGNAME=mytagvalue' myfile.ogg
```

```
$ metaflac --set-tag='MYTAGNAME=mytagvalue' myfile.flac
```

В файлах FLAC и Ogg вы можете называть тэги как угодно, но для MP3 необходимо придерживаться спецификации ID3, согласно которой они должны состоять из четырех символов, начиная с T. За более подробной информацией обращайтесь к соответствующей документации о структуре тэгов: для Vorbis Comments, использующихся во FLAC и Ogg Vorbis, смотрите <http://www.xiph.org/vorbis/doc/v-comment.html>; для ID3, применяющейся для MP3 — <http://id3.org/id3v2.4.0-structure>. Если вы хотите только отредактировать общепринятые тэги, вам больше подойдет графический инструмент. Очень разносторонний редактор тэгов — *EasyTag*, он умеет работать с FLAC, Ogg и MP3 и позволяет править тэги сразу для многих файлов, и можно прописать тэг Альбом сразу на все композиции.

Измененные файлы будут выделены красным. Новичков может удивить, что такое произойдет со всеми файлами MP3 еще до того, как вы там что-либо поменяли. Причина в том, что *EasyTag* по умолчанию добавляет тэги v1.1 и v2.4 и обновляет v2.3 на v2.4. Но вы можете отказаться от этого, используя меню Настройка > Параметры > Настройка тэгов.

В окне *EasyTag* есть поля для общепринятых тэгов, где напротив каждого находится маленькая квадратная кнопка. Если выделить несколько файлов и нажать на кнопку выбранного тэга, он будет добавлен ко всем сразу. Имеются также инструменты для добавления тэгов по именам файлов, и наоборот. Все изменения копяются в памяти, пока вы не велите *EasyTag* их применить. Есть очень мощные инструменты для быстрой обработки файлов пакетами. Но помните, что большая сила — это и большая ответственность: по небрежности легко ошибиться и испортить тэги во множестве файлов.

Abcde справится практически со всем, что касается импорта, кодирования и добавления тэгов; но имея под рукой и более низкоуровневые инструменты, вы почувствуете себя гораздо свободнее. Музыка, оцифрованная в Linux, будет прекрасно звучать даже на вашем iTunes. **LXF**

Альтернативные свободные базы данных CD

Есть несколько свободных баз данных CD, и вы можете научить *abcde* использовать их вместо FreeDB. Все они выполнены на основе последней версии CDDb перед тем, как ее приобрели Gracenote, и после собирали данные самостоятельно. Возможно, вам захочется опробовать другие сервисы, если у вас есть диски, не опознанные текущим. Во всех свободных сервисах используется протокол CDDb1. Чтобы задействовать любой из сервисов, скопируйте его адрес в поле **CDDBURL** в `~/abcde.conf`.

Название	Сайт	CDDBURL для abcde.conf
FreeDB	www.freedb.org	http://freedb.freedb.org/~cddb/cddb.cgi
GNuDB	www.gnuld.org	http://gnuld.gnuld.org/~cddb/cddb.cgi
MusicBrainz	http://musicbrainz.org	http://freedb.musicbrainz.org
TrackType	http://tracktype.org	http://tracktype.org/~cddb/cddb.cgi

Особого внимания заслуживает MusicBrainz, поскольку опережает FreeDB в плане полноты информации и имеет свое приложение для добавления тэгов, *Picard*, на которое стоит взглянуть (http://musicbrainz.org/doc/MusicBrainz_Picard). Документацию по протоколу CDDb ищите на <http://ftp.freedb.org/pub/freedb/latest/CDDBPROTO>.

Режимы спячки:

Нейл Ботвик научит вас сделать нечто поинтереснее, чем тупо выключить свой компьютер по окончании работы.



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.

Несколько лет назад популярно было утверждение, что компьютеру лучше работать 7/24. Предполагалось, что термический и механический стресс от включения системы и раскручивания дисков намного больше, чем от круглосуточной работы. Это, возможно, актуально и сейчас, но данная рекомендация появилась, когда компьютерам требовалось намного меньше энергии и она была намного дешевле. А еще есть вопрос оперативности. Компьютеры загружаются намного быстрее, чем раньше, и все равно остается приличная задержка от момента включения до момента, когда можно приступить к работе. Перевод компьютера в спящий режим будет означать, что он намного быстрее очнется, когда он снова вам понадобится, и при этом вы будете значительно экономить электроэнергию.

Стандарт ACPI определяет ряд спящих режимов, нумеруя их от S0 до S5, и этот номер описывает состояние компьютера. S0 — нормальная работа, или состояние бодрствования. S5 — выключен, но на материнскую плату все равно подается питание; это состояние, в которое входит компьютер при выключении или удержании нажатой кнопки питания, но без отключения от сети устройства питания. Нас интересуют S3 и S4. S3 именуется Standby [Ожидание], Sleep [Сон] или Suspend to RAM [Приостановка в ОЗУ], в зависимости от терминологии, принятой ОС; мы будем называть этот режим Suspend. Как и следует из названия, состояние компьютера сохраняется в ОЗУ, питание которого остается подключенным. Питание CPU и других устройств отключено. Это самый быстрый способ отправить компьютер в спящий режим и пробудить его снова, поскольку работает только ОЗУ. Недостаток в том, что некая энергия все же потребляется для снабжения ОЗУ. Для настольного ПК это нормально; но если надолго оставить в таком состоянии ноутбук, батарея постепенно разрядится и вы потеряете свои данные, и придется вам давать команду **fsck** для своего жесткого диска при следующей загрузке.

S4, именуемое обычно Hibernate [Спячка] или Suspend to disk [Приостановка на диске], записывает состояние памяти и CPU на жесткий диск перед выключением. Это дольше, но после выключения компьютер потребляет столько же энергии, что и в состоянии S5 — почти ничего. Если вы хотите знать, зачем тут вообще питание, ответ такой: материнской плате нужна энергия, чтобы разбудить его после нажатия на кнопку питания, через сеть или из BIOS. Мы будем называть этот режим Hibernate. Желая вы-



» Усыпляя свой ноутбук, блокируйте рабочий стол — на случай, если его разбудит посторонний.

звать режим Suspend или Hibernate, вы можете сделать это, написав псевдо-файл на `/sys/power/state`, используя значения **mem** или **disk**.

```
echo "mem" >/sys/power/state  
echo "disk" >/sys/power/state
```

Для этого надо быть суперпользователем-root, и тем из вас, кто использует **sudo**, придется прибегнуть к некоей хитрости, чтобы сделать это от имени root. Обычное `sudo echo "mem" >/sys/power/state` не сработает, выдав ошибку отказа в доступе, поскольку при запуске пользователем команды произошла ошибка переадресации. Используйте

```
sudo sh -c "echo mem >/sys/power/state"
```

Пробуждение

При погружении в сон состояние вашей системы сохраняется в первом же доступном разделе подкачки [swap]. Чтобы заново загрузиться, вы должны показать ядру, где хранится образ возобновления при перезагрузке, добавив

```
resume=/dev/sdaN
```

в опции ядра вашей программы загрузки. Вы можете или создать отдельную опцию в меню загрузки для возобновления работы, или добавить ее к настройкам по умолчанию. Это безопасно, потому что если раздел `swap` не содержит действующего образа сна — а так и будет, если вы ранее выключали компьютер — то ядро проигнорирует эту опцию и загрузится как обычно. Альтернатива — если вы компилируете собственное ядро, то можете прописать раздел возобновления в разделе Управление энергопотреблением [Power management] в разделе Возобновления по умолчанию [Default resume partition]. Вы также можете вызвать полную перезагрузку после сна, добавив в опции ядра **noresume**, однако это чревато проблемами, поскольку компьютер не выключался ранее должным образом, будучи вместо этого погружен в сон.

Вы, наверное, уже недоумеваете, какой вам толк от всех этих подробностей, если можно просто выбрать Hibernate или Suspend в меню рабочего стола. Основная суть в том, что вы можете сделать нечто большее, чем просто погрузить свой компьютер в сон. Фактически, можно запустить любую команду как перед приостановкой работы, так и после ее возобновления. Возьмем в качестве примера ноутбук, которым вы пользуетесь дома. Вы вошли в систему; возможно, **gpg-agent** сохранил в памяти ваши ключи,

Использование событий ACPI

Вы можете переназначить действия после нажатия горячей клавиши погружения в сон и кнопки питания, добавив опции в конфигурацию **acpid** (но предварительно обеспечьте запуск **acpid** как сервиса при загрузке). Поместите следующий файл в `/etc/acpi/events/power`:

```
event=button[ /]power.*
```

```
action=/etc/acpi/actions/hibernate  
and this in /etc/acpi/events/sleep  
event=button[ /]sleep.*
```

```
action=/etc/acpi/actions/suspend
```

Поместите свои скрипты/символические ссылки в `/etc/acpi/actions` и перезапустите **acpid**. Теперь при нажатии на кнопку питания запустится скрипт `Hibernate`.

Под контролем

если вы подписывали почту, вы соединены с беспроводной сетью или, возможно, смонтировали файловую систему. Итак, вы погружаете компьютер в сон, кладете его в сумку и идете на работу. Как минимум, у вас будут задержки начала работы с ним, потому что сетевые файловые системы недоступны — особо злобствует по этому поводу NFS. В худшем случае — ваш ноутбук украдут, и все данные будут раскрыты. Если, вместо того, чтобы использовать функцию приостановки работы вашего ноутбука, вы настроите горячую клавишу для запуска скрипта, то сможете позаботиться об этой ситуации. Вот пример:

```
#!/bin/sh
[[ "$(basename $0)" == "suspend" ]] && SLEEP="mem" ||
SLEEP="disk"
# gpg-agents забывают пароли, если им послан сигнал SIGHUP
pkill -HUP -f 'gpg-agent --daemon'
# Блокировка рабочего стола
DISPLAY=:0" su nelz -c slock &
# Размонтирование сетевых файловых систем
umount -a -t nfs -t cifs
# Перевод wicd оффлайн
/usr/share/wicd/scripts/50-wicd-suspend.sh
# Сон или спячка
echo ${SLEEP} >/sys/power/state
# Запустить при пробуждении
/usr/share/wicd/scripts/80-wicd-connect.sh
```

Сохраните скрипт как `/usr/local/bin/suspend` и создайте символическую ссылку на `/usr/local/bin/hibernate`; таким образом один скрипт может использоваться для обеих функций в зависимости от имени, использованного для его вызова, то есть как раз того, что делает первая строка.

Скрипт выполняет несколько простых операций: он отправляет SIGHUP в `gpg-agent`, заставляя сбросить все хранящиеся фразы-пароли; затем он блокирует рабочий стол (я здесь использую очень простой `slock`, но вы также можете запустить скрин-сейвер или функцию блокировки с помощью вызова `dbus`). Затем он размонтирует все файловые системы NFS и Samba, выводит сетевой режим в оффлайн, потому что можно разблокировать из другого места (в этом примере использован `wicd`, но то же самое можно сделать и с помощью Network Manager), и, наконец, пишет или `mem`, или `disk` в `/sys/power/state`.

В этот момент скрипт встает на паузу, потому что компьютер спит, но после пробуждения он продолжит свою работу, которая в данном примере возобновит соединение с любой доступной сетью. Сюда можно подставить что угодно, для запуска любой команды, которую вы ни пожелаете, после возобновления работы. На рабочем столе, совместимом с Freedesktop (а большинство из них именно таковы), вы можете заблокировать скрин-сейвер с помощью такой команды:

```
dbus-send --dest=org.freedesktop.ScreenSaver --type=method_
call /ScreenSaver org.freedesktop.ScreenSaver.Lock
```

Эта команда, так же, как и в приведенном выше примере со `slock`, должна быть запущена, когда пользователь запускает рабочий стол. Вызов скрипта `suspend` с помощью горячей клавиши, определенной на рабочем столе — именно тот случай, но если вы используете `acpid`, то нужно настроить переменную `DISPLAY` и указать пользователя, запускающего команду (так как `acpid` запускает команду от имени `root`), таким образом:

Пакет pm-utils

Есть промежуточная стадия между передачей всего на откуп среде рабочего стола и записью в файлы в `/sys`. Пакет `pm-utils` предоставляет ряд команд для работы с приостановкой работы — в частности, `pm-hibernate` и `pm-suspend`. Они могут также добавить ряд шагов для работы с оборудованием, которое не слишком хорошо работает со стандартной приостановкой работы или переходом в спящий режим. Есть также команда `pm-suspend-`

`hybrid`. Она записывает образ спящего режима в `swap`, но потом вместо перехода в спящий режим просто приостанавливает работу. Это позволяет быстро вывести компьютер из сонного состояния; но если ваши батареи разрядятся, вам все равно придется возобновлять работы из данных спящего режима. Получается медленный переход в режим приостановки работы, но быстрое пробуждение — отличная альтернатива стандартному спящему режиму.

```
DISPLAY=:0 su username -c dbus-send --dest=org.freedesktop.
ScreenSaver --type=method_call /ScreenSaver org.freedesktop.
ScreenSaver.Lock
```

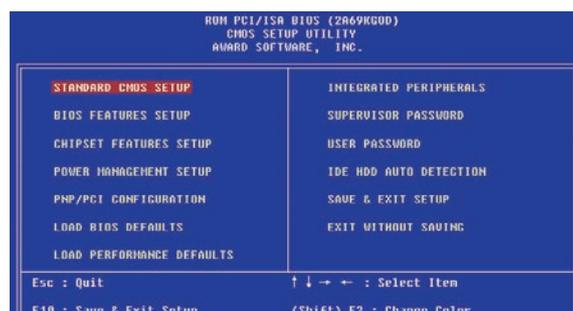
Есть и другие способы разбудить спящую машину. В большинстве BIOSов есть опция пробудки компьютера в заданное время. Это означает, что вы можете, например, оставить свой сервер спать ночью, но разбудить его для работы со скриптами резервного копирования, когда другие компьютеры будут запланированы на из запуск.

Wake-on-LAN

Более удобен Wake-on-LAN, который позволяет разбудить спящий компьютер через сеть. На компьютере, который вы погрузили в сон, не нужно делать ничего, кроме как убедиться, что Wake-on-LAN включено в настройках BIOS. На другом компьютере в сети установите пакет `wakeonlan` и разбудите компьютер командой

```
wakeonlan -i 192.168.1.255 aa:bb:cc:dd:ee:ff
```

где IP-адрес — это адрес вещания для вашей сети (обычно это первые три цифры, общие для вашей сети, за которыми следует 255), а разделенная двоеточиями строка — MAC-адрес адаптера Ethernet на том компьютере, который вы намерены разбудить. Это разошлет «волшебный пакет» Wake-on-LAN всем компьютерам в адресном диапазоне, но к жизни вернется только один, с тем самым MAC-адресом. Итак, вы можете дать выспаться одному из компьютеров в своей LAN, даже если он потом понадобится другому компьютеру, потому что этот другой компьютер сможет при необходимости сам его разбудить. **LXF**



► Вы можете войти в BIOS, чтобы включить Wake-on-LAN, или настроить запланированное пробуждение.

AWS: ЗНАКОМИМСЯ

Роб Добози излагает все, что нужно знать об облачных вычислениях, и приглашает ознакомиться с ними, не потратив ни гроша.



Наш эксперт

Роб Добози — технический архитектор SAP, и работает с разными операционными системами, платформами и базами данных. Он работает в Linux на всех серверах, ПК, ноутбуках, роутерах, мобильных телефонах, телевизорах, медиа-плеерах и Raspberry Pi, которые только есть у него дома.



Облачные сервисы — один из самых громких терминов в мире ИТ. Для тех, кто не вполне представляет себе, что это и как ими пользоваться, мы и подготовили этот учебник. Осознанно или нет, облачные сервисы в какой-либо форме применяют многие. Вы пользовались Ubuntu One, Dropbox, Netflix, Google Docs или Apple iCloud? Все это — облачные сервисы. Что же делает сервис облачным? Национальный институт стандартов и технологии [National Institute of Standards and Technology, NIST] полагает, что он должен иметь такие характеристики:

» **Самообслуживание по требованию** Позволяет предоставлять потребителю такие ресурсы, как время работы сервера или место

на диске, без человеческого участия — например, через браузер или программный интерфейс приложения [Application Programming Interface, API]. Это означает, что вы можете приобрести место для хранения данных через удобный для вас браузер.

» **Универсальный интерфейс** Делает ресурсы доступными через сеть и независимыми от оборудования. Поэтому не должно иметь значения, используете ли вы Linux или Windows, ПК, планшет или телефон. Практически каждый, кто предоставляет облачный сервис, предлагает на выбор клиенту для Windows (некоторые для Linux), Android или iOS.

» **Объединение [pooling] ресурсов** Позволяет провайдеру обслуживать множество клиентов с помощью многопользовательской модели и виртуализации, где ресурсы динамически распределяются по запросу. Провайдеры используют мощные серверы, работая с программами виртуализации (например, KVM или vSphere), которые позволяют запускать несколько виртуальных систем, используемых разными клиентами.

» **Эластичность** Предоставляет возможность быстро изменять объем ресурсов по требованию. Это может увеличить производительность и повысить надежность (благодаря запасным мощностям).

» **Учет потребления** Означает, что облако автоматически измеряет, отслеживает, контролирует и сообщает о потреблении ресурсов, обеспечивая прозрачность этих данных как для провайдера, так и для клиента.

Облачные сервисы предоставляются по трем моделям:

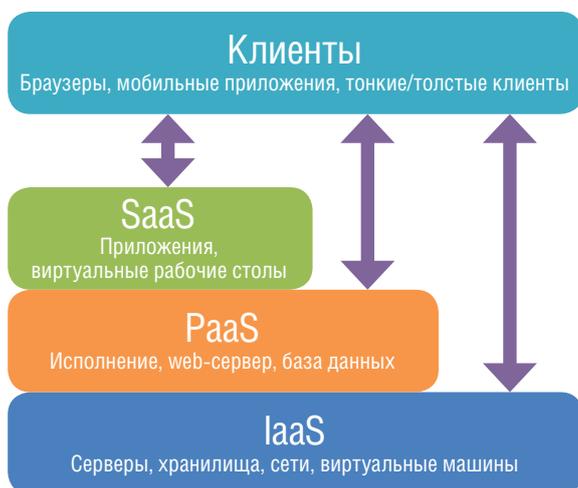
» **Инфраструктура как услуга [Infrastructure as a Service, IaaS]** — базовая модель сервиса. IaaS предоставляет по требованию такие виртуальные ресурсы, как CPU, диск или сеть. За управление ресурсами, операционные системы и любые приложения отвечают пользователи. Среди примеров IaaS — Amazon EC2, Rackspace Cloud Servers или Google Compute Engine.

» **Платформа как услуга [Platform as a Service, PaaS]** предоставляет пользователям платформу для разработки, обычно включающую операционную систему, среду исполнения (например, PHP или Python), базу данных и web-сервер. Вы разрабатываете и применяете приложения к облаку, и вам не надо беспокоиться о лежащих в основе компонентах. Примерами PaaS являются Amazon Elastic Beanstalk, Microsoft Azure, Google App Engine и Force.com.

» **ПО как услуга [Software as a Service, SaaS]** — самая продвинутая модель сервиса; с ее помощью пользователям предоставляются приложения. Обычно пользователи не могут влиять на выбор платформы или базы данных, а в некоторых случаях даже на настройки приложения, но они получают рабочее приложение, которое выполняет их требования, и при этом им не надо беспокоиться об инфраструктуре. Оплата приложений производится за каждого пользователя или за каждую транзакцию. Среди примеров — Google Apps, Salesforce.com и Netflix.

Последнее, о чем стоит сказать — это способы развертывания, определяющие, как облако создается и предлагается пользователю. Общественное облако доступно широкой публике. Частные облака используются исключительно одной организацией (лицом). Гибридные облака — комбинация общественных и частных.

» **Облачный сервис похож на слоеный пирог, причем пользователь находится на самом верху.**



с Облаком

Представляем Amazon Web Services (AWS)

AWS — это платформа, предлагающая гибкие сервисы ИТ-структуры в виде web-сервисов. Звучит красиво, но что это значит? По сути, Amazon предоставляет набор сервисов, доступных через стандартизованный интерфейс. В разных языках программирования интерфейсы разные, но на нашем уроке мы рассмотрим сетевую консоль управления AWS. Следующий вопрос, который у вас, вероятно, возник: какие это сервисы? Ну, когда я заходил туда в прошлый раз, на сайте AWS было перечислено 36 продуктов, или сервисов. Они варьировались начиная с сервера, включая базы данных и хранение, и заканчивая размещением и сетевыми сервисами. Все это довольно сложно и запутанно, поэтому мы рассмотрим некоторые из них более подробно.

Эластичные облачные вычисления (Elastic Cloud Computing, EC2)

Именно их имеют в виду большинство людей, говоря “Amazon cloud”. Они позволяют без особых проблем занимать мощности Amazon. Эти мощности расположены в разных регионах: на Востоке США (Северная Вирджиния), на Западе США (Орегон и Северная Калифорния), в ЕС (Ирландия), Азиатско-Тихоокеанском регионе (Сингапур, Токио и Сидней), Южная Америка (Сан-Паулу) и облако правительства США (GovCloud). Для нас в ЕС наиболее подходящим регионом выглядит Ирландия — особенно если вы планируете обработку личных данных, поскольку вам придется выполнять требования Закона о защите данных [Data Protection Act] от 1998 года. Каждый регион разделен на несколько зон Доступности [Availability]. Каждая зона является физически независимой инфраструктурой и разработана с целью создания высокой степени доступности.

Указав регион и зону, которую вы хотите использовать, выберите оптимальную для себя опцию оплаты. Есть три возможности: On-Demand [По запросу], Reserved [Зарезервированная] и Spot [Немедленный платеж]. Если выбрать On-Demand, то начисляется почасовая оплата, без долгосрочных соглашений или предоплат. Вариант Reserved требует одноразовой предоплаты, которая резервирует один или три часа, со значительно более низкой почасовой ставкой, чем в варианте On-Demand. Вариант Spot позволяет клиентам конкурировать в борьбе за свободные мощности (прямо как на eBay). Теперь поговорим о типах EC2. Именно здесь кроется некая сложность. Есть типы M1, M3, Micro, High-Memory, High-CPU, Cluster Compute, Cluster GPU и High I/O. Большинство из них могут иметь варианты Small [Малый], Medium [Средний], Large [Крупный], Extra Large [Экстра-крупный] и Double/Quadruple Extra Large [Дважды/Четырежды экстра-крупный]. Разница в том, сколько ресурсов отводится каждому варианту. Например, M1 Large Instance имеет 7,5 GiB памяти; 2 ядра с 4 EC2 Compute Units (ECUs) в общей сложности и 850 ГБ места для хранения данных.

Ваш выбор зависит от ваших потребностей в памяти, месте на диске и мощности процессора.

Последнее, что нужно выбрать — операционная система. Есть различные версии Linux и Windows. Все они предоставлены

и запакованы в образ, именуемый Amazon Machine Image, или AMI. AMI могут содержать просто голую ОС — например, Ubuntu, или содержать также и приложения — например, MySQL или Oracle. Одни AMI бесплатны, а за другие надо платить ежемесячно или на почасовой основе. Выбирайте с умом! Все AMI можно найти на рынке AWS, и вы сами можете создать свой собственный.

Хранение

Amazon на данный момент предлагает четыре разных вида хранения, которыми можно пользоваться с экземпляром EC2 или без него. Вероятно, лучше всего известен Simple Storage Service (S3). S3 — это интернет-хранение данных, предназначенное для web-приложений. Среди множества пользователей S3 — такие известные имена, как Dropbox, Ubuntu One и Minecraft. Новейшее дополнение к арсеналу сервисов хранения AWS — это Glacier, недорогой сервис хранения для архивных данных и резервных копий. Он оптимизирован для данных, доступ к которым требуется нечасто, и время загрузки у него несколько часов (как у ленточного накопителя). Ни S3, ни Glacier не могут напрямую использоваться экземпляром EC2, поэтому мы обратим более пристальное внимание на два оставшихся сервиса. Первый — это хранилище для экземпляров виртуальных машин, которое автоматически распределяется при их запуске. Размер хранилища зависит от типа экземпляра (микроекземпляр, который мы используем на данном уроке, вообще не имеет хранилища).

Важнейшее, что надо запомнить о хранилище экземпляра — то, что оно существует, только пока работает виртуальная машина. Если в силу каких-либо причин она выключится, то и хранилище экземпляра будет удалено. А значит, его следует использовать только для чтения или для временных данных, например, /tmp. Самый важный и разносторонний тип хранилища — это Elastic Block Store (EBS), который предоставляет объемы хранения на уровне блочного устройства, виртуального диска. В отличие от хранилища экземпляра, EBS существует независимо от экземпляра EC2. Хотя объемы хранения EBS автоматически воссоздаются в Availability Zone, они не вполне надежны. Amazon заявляет о ежегодной потере данных в размере 0,1–0,5%. Это намного лучше, чем обычный жесткий диск на ПК у вас дома, но все же это не 100%. Чтобы решить эту проблему и чтобы включить резервное копирование, EBS имеет способность создавать моментальные снимки данных, хранящихся в S3. Эти моментальные снимки инкрементны, то есть сохраняются только изменения, сделанные после последнего моментального снимка. Это экономит время, требуемое для создания моментального снимка, и необходимый ему объем хранилища.

Почем?

Со всеми этими сервисами и опциями вам, вероятно, будет любопытно узнать, во сколько они вам обойдутся. Хороший вопрос. Политика ценообразования AWS состоит из целого ряда элементов — что делает ее гибкой, сложной и запутанной. Во-первых, есть стоимость экземпляра, которая складывается из стоимости

»

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

AMI (если только вы не используете бесплатную AMI), предоплаты для зарезервированных вариантов и почасовой стоимости для зарезервированных вариантов и вариантов по запросу. Затем нужно накинуть стоимость хранилища для вашего объема и моментальных снимков EBS. Также начисляется небольшая оплата за каждый миллион запросов I/O по каждому объему EBS. Мы не коснулись ни одного из сетевых сервисов, но вам придется платить за сетевую передачу данных между Availability Zones и за весь

исходящий интернет-трафик (входящий трафик бесплатный). И, наконец, нужно оплатить добавочные расходы — например, на мониторинг или балансировку нагрузки. Может показаться, что Amazon заставляет платить абсолютно за все, однако стоимость весьма разумная. Например, вышеупомянутый вариант M1 Large с 20-ГБ хранилищем EBS, 10 ГБ на моментальные снимки, 130 миллионами запросов I/O в месяц и 30 ГБ интернет-трафика будет стоить примерно \$182 в месяц, если вы будете работать с ним по 12 часов в день. Круглосуточная работа без выходных обойдется в \$306. Это может показаться слишком дорого, если вы просто хотите попробовать облачные вычисления. Но не отчаивайтесь! Amazon предлагает новым пользователям бесплатное пользование сроком на 12 месяцев. Он предлагает вам один вариант Micro, 30-ГБ хранилище EBS, 2 миллиона запросов I/O и 1-ГБ хранилище моментальных снимков. Наш урок использует бесплатный экземпляр, так что можете спокойно изучать основные функции AWS и совершенно ничего не тратить.

DECU — EC2 Compute Units

Клиенты арендуют компьютерные мощности на почасовой основе вместо того, чтобы приобретать специальный процессор, и Amazon пришлось предложить общую единицу измерения мощности CPU. Ее назвали EC2 Compute Unit, или ECU. Один ECU примерно эквивалентен мощности CPU 1,0–1,2 ГГц 2007 Opteron или Xeon.

Курс на облако

Ну вот, мы рассмотрели все кирпичи, из которых складывается основа; сложим же их вместе — и перейдем к их использованию.

Для первого запуска экземпляра EC2 вам придется использовать AMI. Большинство экземпляров имеют локальное кратковременное хранилище, исчезающее по выключении экземпляра. Тома EBS могут прилагаться к экземпляру, и любой сделанный моментальный снимок хранится в S3. Приложение, работающее внутри экземпляра, может также использовать S3 API для хранения любых данных в блоках памяти S3. С точки зрения расходов важно помнить, что вы платите за место, отведенное под тома EBS, моментальные снимки и блоки памяти S3 независимо от того, используете ли вы их или нет (отмечены значком \$ на рисунке).

Хорошая новость в том, что если у вас не интенсивно используемый зарезервированный экземпляр, вы платите только за время, когда ваш экземпляр находится в рабочем состоянии.

Давайте начнем с регистрации на сайте AWS. На <https://aws.amazon.com> щелкните по кнопке Sign Up наверху страницы. Если вы делаете покупки в Amazon, то большая часть вашей информации там уже имеется. Вам придется указать данные своей кредитной карты, но не волнуйтесь — Amazon не заставит вас ни за что

платить, пока не истечет срок бесплатного пользования. По завершении проверки по номеру телефона и оставшихся шагов регистрации откройте консоль на <https://console.aws.amazon.com>. Это главный интерфейс управления всеми сервисами AWS. Теперь щелкните по EC2 Service, чтобы запустить EC2 Dashboard.

Прежде чем что-либо делать, измените регион в верхней правой части с региона по умолчанию (N. Virginia) на Ирландию (Ireland), поскольку все экземпляры EC2 будут созданы в выбранном на текущий момент регионе.

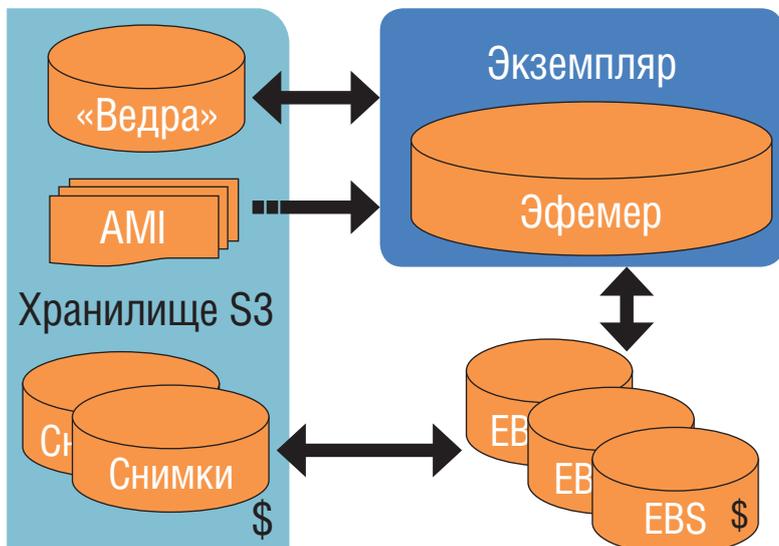
За шесть шагов, описанных на следующей странице, мы создадим новый экземпляр и убедимся, что он работает.

Когда экземпляр заработает, вы должны суметь соединиться с ним через SSH на его публичное имя DNS. Не паникуйте, если вам не удастся соединиться с ним сразу после запуска: у DNS может уйти одна-две минуты на обновление. Щелкните правой кнопкой по варианту и в меню выберите Connect. Есть две опции соединения. Вторая позволяет напрямую соединиться из браузера, если у вас установлена Java. Первая, которую мы собираемся использовать, предназначена для соединения из командной строки. Следуйте инструкциям на экране. Единственное отличие в том, что в системах на базе Ubuntu надо заменить **root** на **ubuntu**:

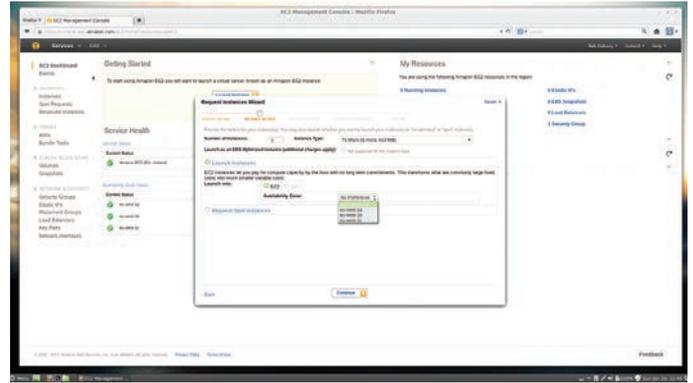
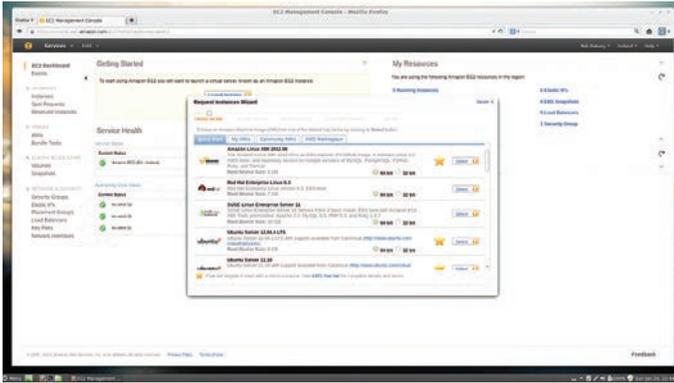
```
cd Downloads
chmod 400 MyKeyPair.pem
ssh -i MyKeyPair.pem ubuntu@<public_DNS_of_your_instance>
Теперь можете исследовать свой экземпляр. Попробуйте проверить память (free) и место на диске (df -h). Чтобы попробовать сделать что-то поинтереснее, давайте установим web-сервер Apache и соединимся с ним:
apt-get install apache2
service apache2 start
```

Как только вы увидите сообщение, подтверждающее, что сервер работает, попробуйте соединиться с ним с помощью своего web-браузера. Чтобы убедиться, что это ваш сервер, измените файл `/var/www/index.html` с помощью своего любимого редактора, например, *Vi* или *nano*, и перезагрузите страницу в своем браузере. Если вы не видите сделанных вами изменений, проверьте, сохранили ли вы файл в редакторе, и попробуйте снова. Если у вас получилось — отлично! Используйте свой облачный сервер для какого-нибудь крутого проекта. Помните, что пределы скрыты далеко за облаками!

➤ Именно так экземпляры EC2 Instances и разные виды хранилищ работают вместе.



➤ Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

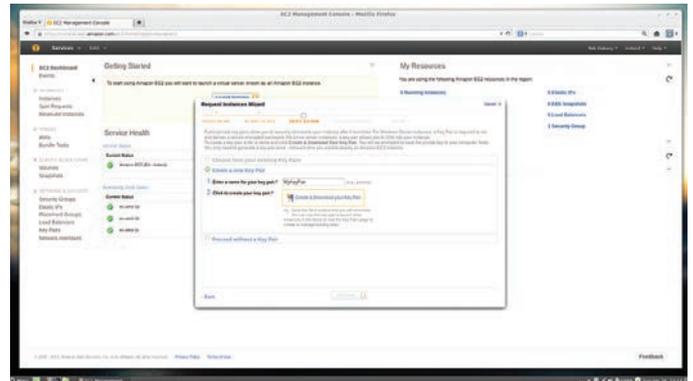
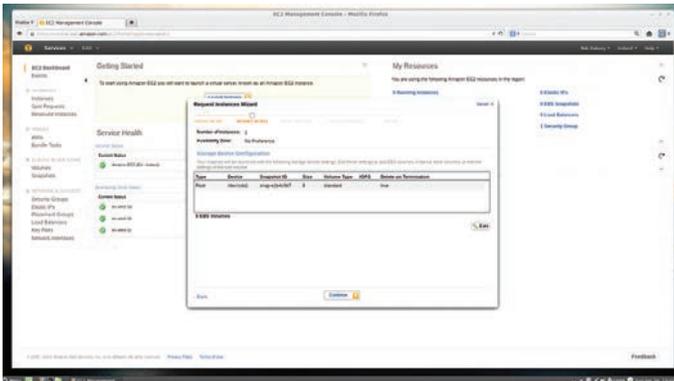


1 Запустите Instance Wizard

Сначала щелкните по кнопке Launch Instance [Запустить экземпляр]. По умолчанию будет выбран классический Wizard [Мастер]. Нажмите на Continue [Продолжить]. На стадии AMI можете выбрать, с помощью какого AMI вы хотите запустить свой экземпляр. В этом руководстве мы будем использовать Ubuntu Server 12.04.1 LTS.

2 Данные об экземпляре

На шаге Details [Подробнее] может выбрать количество экземпляров, которые вы хотите запустить (мы остановились на одном) и тип экземпляра (мы будем использовать T1 Micro, поскольку он бесплатный). Здесь вы можете решить, в какой Availability zone [Зоне доступности] вам нужно запустить свой экземпляр. На данный момент можете выбрать No Preference [Без предпочтений].

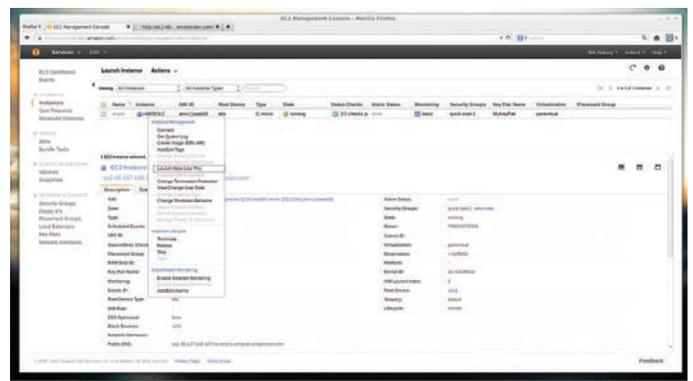
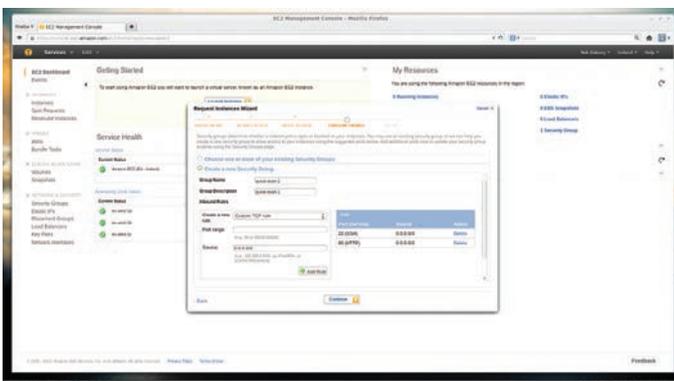


3 Расширенная информация по экземпляру

В первом окне Options [Варианты] щелкните по опции Termination Protection [Защита от прерывания] и нажмите на Continue, чтобы перейти к настройке хранилища. Здесь вы можете изменить тома EBS, прилагаемые к экземпляру, но сейчас просто нажмите Continue. Третье окно позволит вам добавлять теги к экземпляру, чтобы вы смогли распознать его впоследствии.

4 Создание пары ключей

Пара публичный/приватный ключ требуется для того, чтобы открыть SSH-соединение, поскольку варианты EC2 не используют пароль по умолчанию. Выберите имя для вашей пары ключей (вы можете иметь разные имена для разных систем) и нажмите Create & Download your Key Pair [Создать и загрузить вашу пару ключей]. Сохраните файл, когда этого потребует ваш браузер.



5 Настройки брандмауэра

Группы безопасности определяют то, к чему может быть получен доступ в вашем экземпляре. По умолчанию разрешен только SSH (порт 22). Давайте добавим HTTP из выпадающего меню. Не забудьте нажать Add Rule [Добавить правило], чтобы и 22, и 80 появились в списке портов.

6 Обзор и запуск

Проверьте настройки и запустите экземпляр. На странице Instance Overview [Обзор экземпляра] вы можете увидеть подробную информацию по экземпляру и его статусу. Запуск займет у вас не более одной-двух минут. Щелкнув правой кнопкой по экземпляру, вы сможете выполнять основные действия, такие как запуск/остановка или изменение настроек экземпляра. LXF

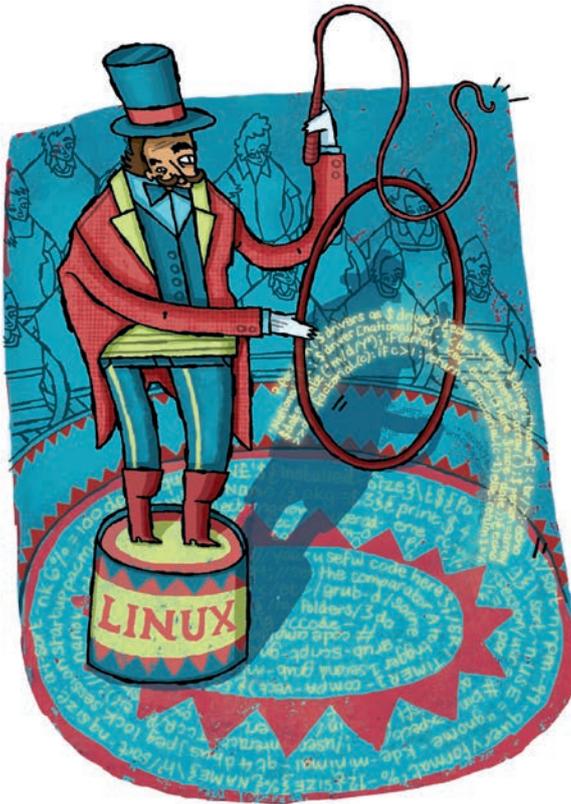
Тестирование:

Джоно Бэкон исследует мир автоматического и ручного тестирования, чтобы помочь вам сделать свое приложение прочным, как скала.



Наш эксперт

Джоно Бэкон — менеджер сообщества Ubuntu, автор *The Art Of Community* [Искусство Сообщества] и основатель ежегодной встречи Community Leadership Summit.



Качество — ключевой фактор для любой программы. Неважно, насколько современны ваши функции, насколько впечатляет ваш интерфейс или насколько уникально ваше приложение: если оно ненадежно работает, им перестанут пользоваться.

Однако качество важно не только для обеспечения пользователю приятной работы — оно важно также для того, чтобы ваше сообщество чувствовало себя счастливым и преданным вашему приложению. Плохое качество отражается на всех, кто отстаивает интересы сообщества, а вам ведь не надо, чтобы ваше сообщество постоянно получало жалобы на плохое качество программы и на ошибки в ней.

Создание высококачественного ПО означает понимание процессов Quality Assurance (QA, Обеспечение качества). Некоторые из вас, возможно, уже знакомы с QA, и ассоциируют его с отслеживанием ошибок и тестированием программ. В целом это правильно, но эффективное QA в меньшей степени относится к сообщениям о дефектах и отслеживанию их, а в большей степени — к обеспечению качества вашего ПО. Иными словами, лучше постараться обеспечить высокое качество выпускаемого ПО, чтобы в нем вообще не было ошибок, а не отличные процессы для сообщения об обнаруженных ошибках.

Чтобы обеспечить качество, требуется эффективное тестирование кода, изгоняющее из кода как можно больше ошибок.

И эта задача намного сложнее, чем можно ожидать. С одной стороны, всегда можно протестировать разные части кода и проверить, работают ли они, как положено; но как быть в тех случаях, когда код работает в необычных условиях или с необычными данными? При этом возникают хуже всего отслеживаемые проблемы, а также гонки, когда разные ветки кода исполняются в разное время.

Чтобы добиться нашей цели, обеспечения качества, лучшим методом будет применить многогранный подход к тестированию. В идеале надо протестировать наш код поблочно, а также протестировать все функции с точки зрения пользователя. Ограничившись одной из этих задач, должный уровень качества вы не обеспечите; а выполнив обе, станете гораздо ближе к нирване стабильности ПО.

Поблочное тестирование

Самая важная форма тестирования, которую нужно встроить в ваше приложение, это блочное тестирование. Огромное количество программ написано на функциональных объектно-ориентированных языках программирования, в которых код разбивается на функции многократного использования, которые служат специальным целям. Например, у вас, возможно, есть функция, которая пишет файл на диск, или конвертирует данные в другой тип, или возвращает web-страницу с web-сервера. Поблочное тестирование разработано для того, чтобы все эти разнородные функции работали как полагается и добивались желаемого результата.

Общая философия поблочного тестирования заключается в том, что у вас есть тест для каждой функции в вашем приложении. Во многих программных проектах новую функцию не примут в кодовую базу проекта, пока не будет протестирован соответствующий блок. Аналогично, новые функции и поправки не попадут в программу, пока не пройдут блочного тестирования. Это обеспечивает надежный способ проверки, что новые куски кода нечаянно не нарушат уже отлаженные функции вашей программы.

Есть множество разных сред для написания блочных тестов, и неудивительно, что многие из них относятся именно к тому языку, на котором написан код. В этой статье я покажу вам, как использовать модуль **unittest**, являющийся частью Python; однако структура создания теста может использоваться в большинстве других тестовых сред.

Прежде чем приступить к созданию теста, предположим, что у нас есть простой класс со следующими функциями:

```
class MyClass():
    def return_true(self):
        return True
    def get_version(self):
        version = "1.0"
        return version
    def get_file(self, fileloc):
        try:
            with open(fileloc) as f: return True
```

Даетшь качество

```
except IOError as e:
    return False
```

Функций здесь три:

» **return_true()** В таком виде это самая бесполезная из всех когда-либо написанных функций; но вы представьте, что она делает нечто осмысленное и затем возвращает True.

» **get_version()** Эта функция просто возвращает текущую версию ПО в качестве строковой переменной. Она может быть полезна при написании API, чтобы клиент взял правильную версию API.

» **get_file()** Эта функция проверяет, существует ли указанный файл.

Все эти функции возвращают данные, и нам надо написать тест блока, проверяющий, что вернулись правильные данные. Если возвращаются другие данные (например, False вместо True), значит, блочный тест не проходит.

Давайте сначала добавим утверждения **import**, которые мы будем использовать в этой статье, а потом добавим приведенный выше класс:

```
import unittest
import tempfile
import os, sys, shutil
class MyClass():
def return_true(self):
    return True
def get_version(self):
    version = "1.0"
    return version
def get_file(self, fileloc):
    try:
        with open(fileloc) as f: return True
    except IOError as e:
        return False
```

Теперь создайте свой первый тест, в данном случае — для функции **return_true()**. Под кодом, который мы только что добавили (код см. выше), вставьте следующее:

```
class Tests(unittest.TestCase):
def test_return_true(self):
    myclass = MyClass()
    value = myclass.return_true()
    self.assertTrue(value)
if __name__ == '__main__':
    unittest.main()
```

Здесь мы создаем новый класс **unittest** под названием **Tests**. Внутри этого класса мы создаем блочный тест для каждой функции. Как видите, мы добавили наш первый блочный тест, под названием **test_return_true()**.

Для начала мы создаем экземпляр класса **MyClass**, который тестируется, а затем запускаем функцию и пишем результат в **value**.

Следующий наш шаг — проверка, являются ли данные в **value** тем, что мы ожидали. С этой целью мы выполняем утверждение. Утверждение проверяет, соответствует ли передаваемая ему величина тому, что оно ожидает. В этом тесте мы используем **assertTrue()**, являющееся частью библиотеки **unittest**, чтобы проверить, является ли **value** истинной (True), как мы рассчитываем.

Затем в конце исходника мы запускаем функцию **main()** модуля **unittest**, чтобы провести тесты.

Чтобы запустить свои тесты и узнать, пройдут ли они, просто выполните скрипт, и вы должны увидеть следующее:

```
jono@forge:~/Desktop$ python tests.py
-----
Ran 1 test in 0.000s
OK
jono@forge:~/Desktop$ python tests.py
F
=====
FAIL: test_return_true (__main__.Tests)
-----
Traceback (most recent call last):
File "tests.py", line 35, in test_return_true
self.assertTrue(value)
AssertionError: False is not true
-----
Ran 1 test in 0.000s
FAILED (failures=1)
```

Здесь мы запустили один тест, и он прошел отлично. '!' над линией показывает, что тест пройден. Чтобы проверить, не провалился ли тест, измените **return True** на **return False** в функции **return_true()** и снова запустите скрипт. Теперь на экране должно появиться

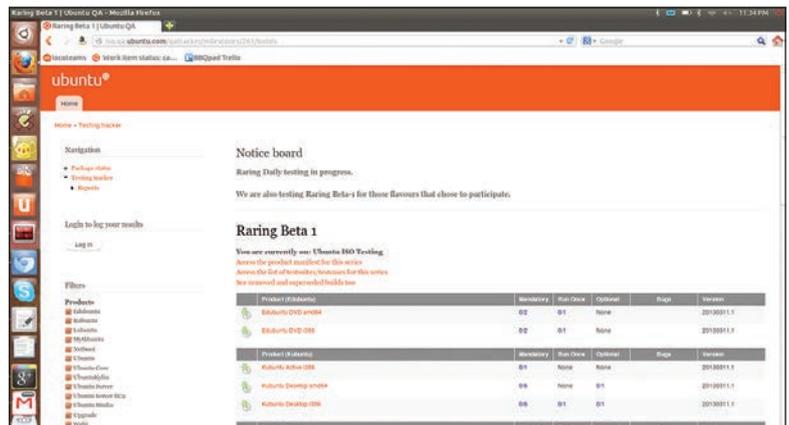
Здесь тест не пройден, и указана причина провала. Давайте напишем второй тест, чтобы проверить, возвращает ли **get_version()** правильную версию. В этом случае нам не нужно тестировать конкретно версию 1.0, поскольку версии регулярно изменяются. Вместо этого следует убедиться, что функция возвращает переменную, например, 1.0 или 1.5, и мы будем считать эту переменную соответствующим номером версии (поскольку никакой другой код не вносит данные в эту функцию).

Чтобы добавить данный тест, вставьте такую функцию в класс **Tests**:

```
def test_get_version(self):
    myclass = MyClass()
    version = myclass.get_version()
    self.assertTrue(isinstance(version, basestring))
```

Здесь мы снова создали экземпляр класса **MyClass**, запустили **get_version()** и вывели результат в **version**.

» **Ubuntu test tracker** хранит ручные тесты и результаты для Ubuntu.



Теперь нам надо протестировать, является ли **version** переменной. Для этого мы используем **isinstance()**, чтобы проверить, относится ли **version** к формату **Basestring**, и, если это так, то вернется True; затем мы проверим это в функции **assertTrue()**, чтобы вернуть результат утверждения. Снова запустив скрипт, мы увидим:

```
jono@forge:~/Desktop$ python tests.py
..
-----
Ran 2 tests in 0.000s
OK
```

Здесь видно, что проведено два теста, и над линией поставлено по точке за каждый удачно пройденный тест.

И снова, если вы хотите, чтобы тест не прошел, отредактируйте исходную функцию, на сей раз изменив "1.0" на **1.0** (удалите кавычки, превратив единицу в число вместо переменной) и перезапустите скрипт, чтобы увидеть провал.

Для нашего финального теста я хочу поговорить о важных частях написания блочных тестов — подготовке и закрытии.

Вы, возможно, заметили, что наши функции не работают с реальными данными. Например, если нужно провести блочный тест, чтобы увидеть, содержит ли определенного типа файл определенный тип данных, как провести этот тест и узнать, что данные, которые мы вводим в функцию — это данные, соответствующие тому типу, на который мы рассчитываем? Конечно, данные могут быть другими, что может послужить причиной провала теста, даже если данные были приемлемы.

Чтобы решить эту проблему, модуль **unittest** (и многие другие среды тестирования) позволяет запускать эквивалент создателя классов, применимый для создания пробных данных для теста. Точно так же есть эквивалент деструктора классов, который можно использовать для последующего удаления этой тестовой среды. Рассмотрим нашу последнюю функцию, для которой мы хотим написать тест:

```
def get_file(self, fileloc):
    try:
        with open(fileloc) as f: return True
    except IOError as e:
        return False
```

В данном случае функция проверяет, существует ли файл, и если да, то возвращает True; в противном случае появляется ошибка IOError, и функция возвращает False. Для эффективного тестирования этой функции нам нужно знать, действительно ли существует файл, который мы ей передаем.

Вот тут пригодится папка **/tmp** в вашем компьютере. Мы автоматически создадим несколько файлов в **/tmp**, чтобы точно знать те файлы, которые наш блочный тест будет использовать в качестве исходных данных.

Для этого добавьте следующую функцию в ваш класс Tests:

```
def setUp(self):
```

```
self.temp_path = '/tmp/testingtemp/'
if not os.path.exists(self.temp_path): os.mkdir(self.temp_path)
for i in range(1, 6):
    file = open(os.path.join(self.temp_path, str(i) + ".txt"), 'w')
    file.write('foo')
file.close()
```

Здесь мы создаем нашу функцию **setUp** (эквивалент создателя классов). Для этого создается функция под названием **setUp()**, определяется местоположение в **/tmp** для сохранения наших файлов, проверяется, существует ли эта директория, и затем создается пять небольших текстовых файлов под названием **1.txt**, **2.txt**, и т.д.

Когда мы запускаем наши тесты, функция **setUp()** запускается перед выполнением любого теста. По завершении этой функции у нас будет пять текстовых файлов в **/tmp/Testingtemp**, которые мы используем в тестах. Это обеспечит готовность нашей тестовой среды перед запуском тестов.

Давайте теперь создадим тест:

```
def test_get_file(self):
    myclass = MyClass()
    value = myclass.get_file("/tmp/testingtemp/1.txt")
    self.assertTrue(value)
```

Здесь мы создаем экземпляр класса **MyClass**, запускаем функцию **get_file()** и передаем ей один из файлов, созданных с помощью **setUp()**. Технически нам нужно создать один текстовый файл, но мне показалось веселее создать пять. Затем тест проверит, является ли величина, возвращаемая из **get_file()**, истинной (True). Если это так, тест пройден. Запустив скрипт, мы увидим:

```
jono@forge:~/Desktop$ python tests.py
...
-----
Ran 3 tests in 0.001s
OK
```

Как видите, все три теста пройдены успешно.

Удаление тестовых данных

Хотя **/tmp** периодически вычищается системой, и наши тестовые данные будут удалены, хорошим тоном считается предусмотреть функцию, очищающую от тестовых данных. Для этого создадим функцию **tearDown()** в классе Tests:

```
def tearDown(self):
    temp_path = '/tmp/testingtemp/'
    shutil.rmtree(self.temp_path)
```

Эта функция просто удаляет директорию из **/tmp**. Теперь, запустив тесты и заглянув в **/tmp** после их завершения, вы увидите, что тестовых данных там нет.

Конечно, есть много других функций и возможностей в модуле **unittest**, но я советую заглянуть в руководство пользователя на <http://docs.python.org/release/2.6.6/library/unittest.html>, где вы найдете более подробную информацию, или в документацию тестовой среды, которой вы пользуетесь.

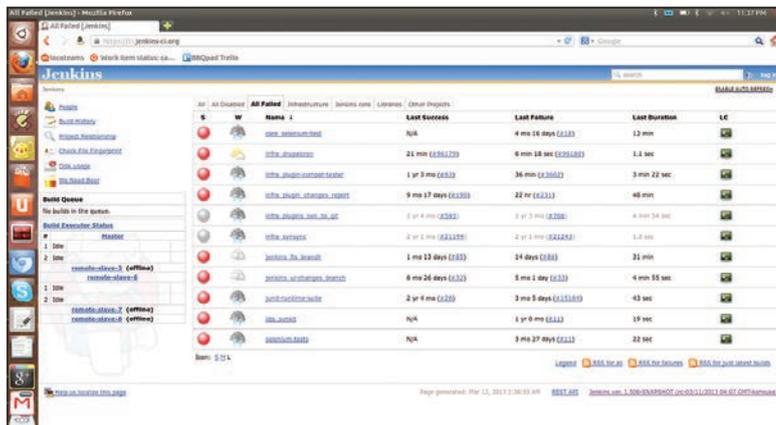
Тестирование функций

Блочные тесты являются важной частью разработки ПО, и я настоятельно рекомендую вам обзавестись пакетом блочных тестов для ваших приложений, желательно с тестом для каждой функции. Однако блочные тесты — это только проверка функциональных возможностей кода. И они совершенно не выявляют неожиданных результатов при работе приложения.

Среди примеров таких неожиданных результатов могут быть:

- » рендеринг ошибок в графических приложениях;
- » сломанные или не отвечающие графические виджеты;
- » проблема недоступности сетевого соединения;
- » текст в графическом приложении занимает слишком много места на экране.

» **Jenkins** предоставляет удобный инструмент для непрерывной интегрированной работы, например, для повторного запуска тестов.



- » Приложения работают со сбоями или с ошибками.
- » Проблемы интеграции приложения с другими компонентами системы (например, с темами).

Любой из этих сценариев может возникать при благополучно пройденном наборе блочных тестов. И это оставляет нам две возможности тестирования функций: тесты рабочего стола и ручные тесты.

Тесты рабочего стола — это автоматические тесты, имитирующие щелчки мышью в приложении и оценивающие результаты этих щелчков на предмет их соответствия ожиданиям. Есть два основных подхода к проведению этих тестов:

» **Уровень доступности** — эти тесты создаются иницирующими событиями с помощью среды доступности на рабочем столе (той же среды, которая используется инструментами доступа, например, программами для чтения с экрана).

» **Снимки с экрана** — эти тесты основаны на том, что делается серия скриншотов, и затем сравниваются функции приложения с частями скриншота этого приложения (например, панель инструментов приложения соответствует панели инструментов на скриншоте).

И хотя обе эти техники бывают полезны, они предполагают наличие неких технологий (например, среды доступности или настроенного набора скриншотов, которые соответствуют теме рабочего стола). Поэтому отсылаю вас к инструментам, используемым для опробования этих подходов (в наборе инструментов Desktop Testing Tools), а вместо этого мы сконцентрируемся на ручных тестах, приложимых ко всему.

Ручное тестирование

Ручные тесты — это подборка небольших предписывающих инструкций, которые мы просим выполнить пользователя для создания ожидаемого результата, а затем просим пользователя сравнить этот результат с должным.

Инструменты для предоставления пользователю ручных тестов (например, *Ubuntu test tracker*) в первую очередь предназначены для перечисления тестов и предоставления места для сохранения результатов тестирования. Однако в реальности вы можете использовать для этого другие инструменты, например, wiki или электронную таблицу для хранения результатов.

Создание ручных тестов может показаться не сложнее, чем написание нескольких инструкций, однако вам надо подойти к этому более методично, чтобы вы точно смогли протестировать все необходимые части вашего приложения, и чтобы каждый тест работал как надо и выдавал те результаты, которых вы и ожидали. Мы не хотим, чтобы тест провалился из-за того, что ваши инструкции неточны и пользователь их недопонял и нажал не там.

Первый шаг в написании отличного ручного теста — это определение точного списка того, что надо тестировать. Например, для текстового редактора рабочего стола нужно будет протестировать:

- » операции с файлами (загрузка/сохранение/перезапись);
- » добавление, редактирование, удаление и перемещение текста;
- » такие функции, как проверка правописания, поиск, замена, статистика по словам, и т. д.

Составив список требований, можете приступать к написанию тестов, которые будут выполнять пользователи.

Но прежде чем забарабанить по клавиатуре, обдумайте, какие основные тесты вам нужны — те, что отсутствуют в виде блочных тестов или в иной форме тестирования. Когда мы просим пользователей провести ручное тестирование, мы не рассчитываем, что они просидят за тестами четыре часа; это весьма скоро им надоест. Куда практичнее попросить их уделить 20 минут и протестировать самые проблемные или рискованные области вашего приложения, чтобы и нужное тестирование обеспечить, и не ввергнуть пользователя в тоску.

Помня о том, какие вам нужны тесты, создайте в текстовом редакторе новый документ, чтобы написать их, и присвойте каждому тесту номер и идентификатор. Например:

```
ED-001 file-loading
ED-002 file-editing
ED-003 file-saving
```

Для каждого теста добавьте описание того, что он делает. Описание должно быть высокоуровневым, но подробным настолько, чтобы быть понятным людям, незнакомым с тестом и приложением. Например:

```
ED-001 file-loading Загружает текстовый файл в редактор,
готовый к редактированию.
ED-002 file-editing Редактирует текстовый файл посредством
ввода, удаления и перемещения текста.
ED-003 file-saving Сохраняет текст в новый файл.
```

Затем задокументируйте все настройки, которые должен сделать пользователь перед запуском теста. Например, для ED-001 — должен ли файл, который он загружает, быть в определенной кодировке и должен ли он загружаться с жесткого диска или с устройства или из сетевого ресурса? Это должно быть ясно указано. Например:

Подготовка: используйте *TextEditor 1.0* и загрузите текст в формате UTF-8 с локального жесткого диска.

Теперь для каждого теста напишите набор действий, объясняющих, как проводить тест. Например, для ED-001:

- 1 Щелкните по пункту меню File.
- 2 Щелкните пункт Открыть... внутри меню File.
- 3 Используя выбор файла, выберите текстовый файл (текстовый файл показан небольшим значком с блокнотом и должен иметь расширение .txt).

Каждый тест должен включать не более 10–15 действий; если их будет больше, это просто убьет пользователя.

Теперь внятно и четко опишите ожидаемый результат теста. Например, для ED-001:

Результат: Текстовый файл загружается, и весь текст отображается в текстовом режиме со всеми переводами строки и воротами каретки.

Получившийся у вас готовый тест должен выглядеть приблизительно так:

```
ED-001 file-loading Загрузка текста в редактор, готовый к
редактированию.
```

Подготовка: используйте *TextEditor 1.0* и загрузите текст в формате UTF-8 с локального жесткого диска.

- 1 Щелкните по пункту в меню File.
- 2 Щелкните пункт Открыть... в меню File.
- 3 Используя выбор файла, выберите текстовый файл (текстовый файл показан небольшим значком с блокнотом и должен иметь расширение .txt).

Результат: Текстовый файл загружается и весь текст отображается в текстовом режиме со всеми новыми строками, и носитель возвращается.

Затем пользователь должен сообщить о том пройден ли тест (PASS) или не пройден (FAIL), когда он получит результат, следуя всем перечисленным инструкциям. Поздравляем, теперь у вас есть ручной тест! **LXF**

Инструменты тестирования

Вот несколько полезных инструментов для тестирования графических приложений:

- » Sikuli Script <http://www.sikuli.org>
- » Linux Desktop Testing Project <http://ldtp.freedesktop.org>
- » Dogtail <https://fedorahosted.org/dogtail>
- » SWTBot <http://eclipse.org/swtbot>

Inkscape: Копия-

Создайте весьма детальный рисунок без особого труда. Простыми эффектами *Inkscape* Ник Вейч добивается потрясающих результатов.



Наш эксперт

Когда начинался LXF, его держали на плаву только скрипты *Bash* от Ника Вейча. Потом их заменили «люди», и это, по мнению Ника, стало явным шагом назад...

Если вам нужно что-то нарисовать, с *Inkscape* это пара пустяков. *Inkscape* не только имеет обширный набор полезных и в общем простых для понимания инструментов, но и работает с SVG, стандартным форматом векторной графики. Талантливый художник сможет создавать в *Inkscape* удивительные вещи; и даже если у вас руки-крюки, годные разве что удерживать кусок мела, вы при известных усилиях состряпаете вполне приличное изображение.

Наш урок объяснит вам некоторые «скрытые» функции рисования *Inkscape*, посредством практического упражнения. Да не так уж эти функции и «скрыты», просто их часто не замечают или они кажутся слишком сложными, чтобы за них браться. Мы приоткроем завесу тайны, окружающей применение эффектов, расширений и фильтров, чтобы в итоге получить изображение в стиле «отсиненной» копии, которая выглядит не слишком идеально — в данном случае недостатки изображения введены намеренно.

Возможно, вы уже применяли фильтры раньше — почти наверняка, если занимались в *Inkscape* растровыми изображениями. К растру обычно обращаются, когда в изображении надо добавить текстуры, и, честно говоря, при наличии достойной готовой текстуры создание того же самого эффекта потребует десятую долю наших усилий. Почему бы нам так и не поступить?

Растровые изображения не только немало весят — они также не масштабируются. При работе с векторным изображением о масштабе волноваться незачем: изображение все равно оста-

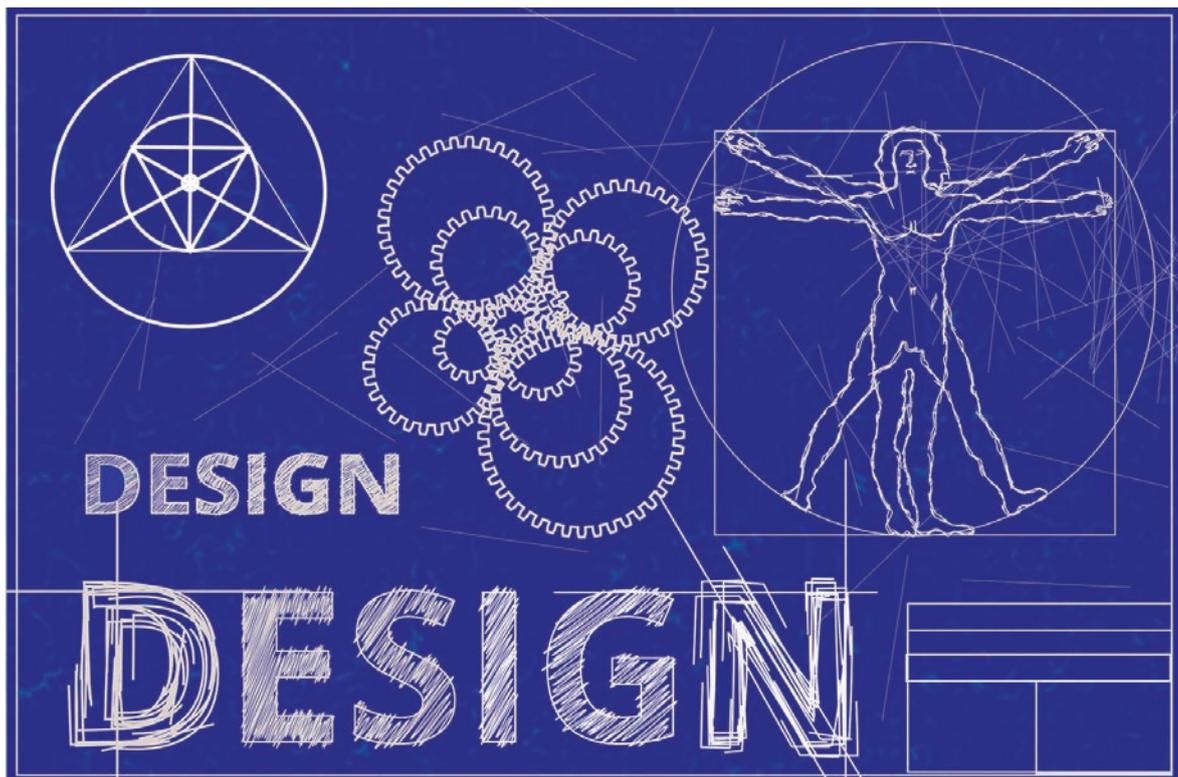
нется узнаваемым, поскольку построено исключительно из форм, а те сохраняются в любом размере. Поэтому не мешает знать, как создать эффект растрового изображения, используя всего лишь обычные формы и немного процедурной магии от *Inkscape*.

Недостаток процедурных методов в том, что некоторые из них, в зависимости от выбранного параметра, могут затребовать изрядные ресурсы CPU на рендеринг. Видя, как надсажается ваш компьютер, пытаясь воссоздать изображение на экране, вы, возможно, захотите немного снизить интенсивность эффектов.



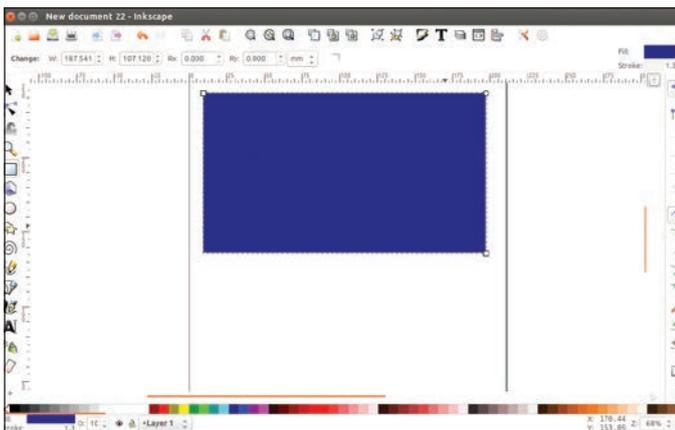
» Сочетая примененные нами эффекты, вы создадите любые рисунки-наброски!

» Наш рисунок в стиле синьки выглядит как собранный в растровой программе для рисования, но он на 100% создан из масштабируемых векторных форм!



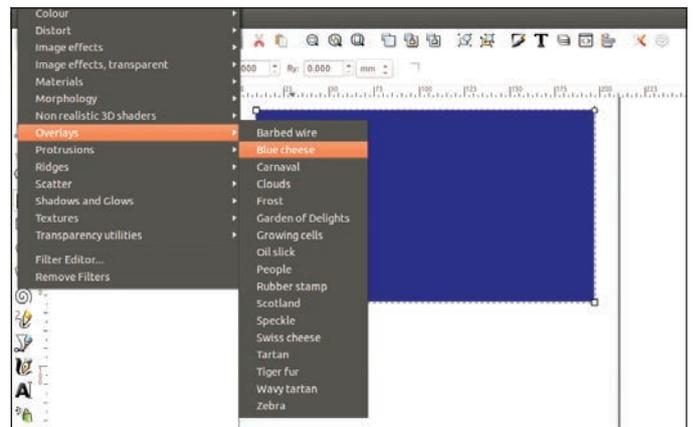
«СИНЬКА»

Шаг за шагом: Делаем «синьку» изображения



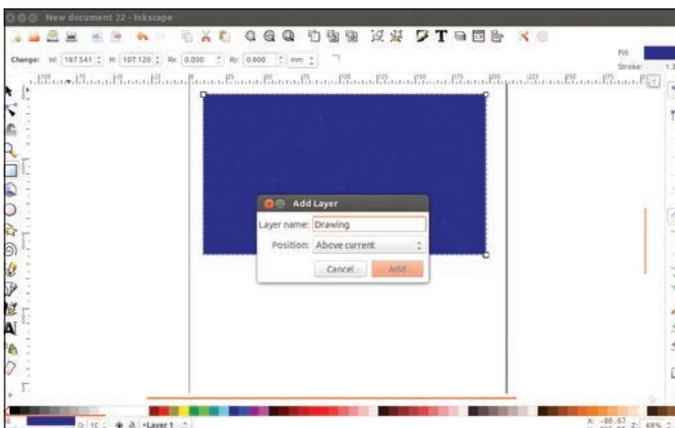
1 Сделаем синий фон

Сначала создадим синий прямоугольник — он будет фоном нашего изображения. Выберите цвет Navy [морской] в палитре внизу окна *Inkscape*. Затем выберите инструмент Прямоугольник в панели инструментов сбоку и создайте прямоугольник практически во всю ширину страницы. Изменить страницу, чтобы он лучше туда вписывался, мы еще успеем, так что не спешите в режим пейзажа. Некоторые из применяемых эффектов зависят от исходного размера фона, и у вас могут получиться другие результаты, если вы решите создать версию формата A0; или, по крайней мере, вам придется настроить некоторые параметры.



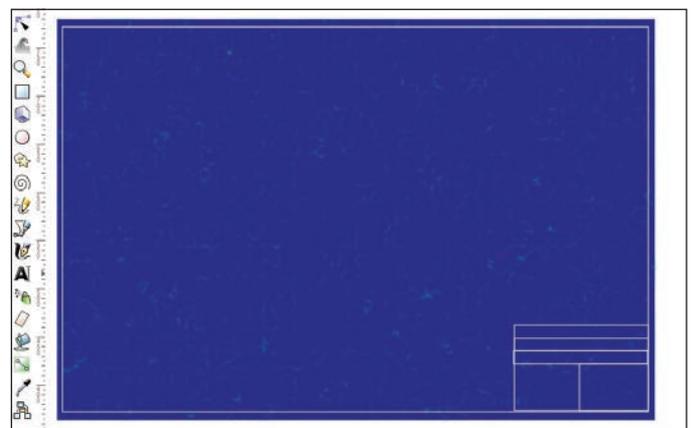
2 Добавим сыр

Да, все верно, мы добавим эффект сыра. Текстура фона придаст ему вид подлинной копии-синьки. Главное — не перебрать, потому что все же хочется распознать изображение, когда оно будет готово. Эффект, который вы найдете в *Filters > Overlays > Blue Cheese* [Фильтры > Наложения > Синий сыр] придаст вашему фону несколько памятный вид. Излишнего шума не добавится, и рисунок останется разборчивым; эффект не особо ресурсоемкий, так что вам не придется часами ждать обновления экрана.



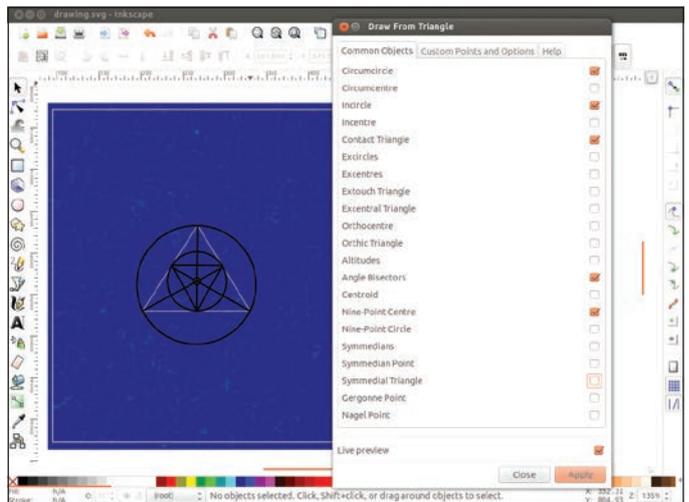
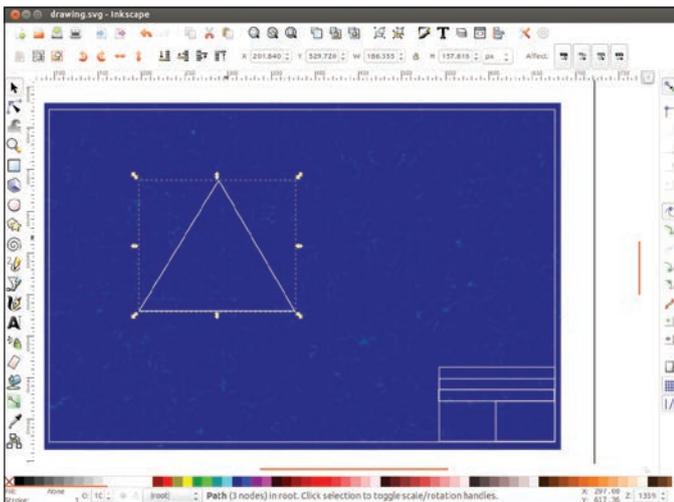
3 Создадим новый слой

Мы не хотим случайно испортить фон, поэтому наилучшим выходом будет создать поверх него новый слой и разместить рисунок на нем. Для начала надо заблокировать текущий слой (щелкните по значку с замком в середине строки состояния внизу экрана). Затем выберите *Select Layer > Add Layer...* [Выбрать слой > Добавить слой] в меню. Введите имя нового слоя и проверьте, чтобы он был настроен на *Above Current Layer* [Над текущим слоем]. Переключаться между слоями легко: выбирайте нужный слой из всплывающего меню в середине строки состояния.



4 Нарисуем что-нибудь

Пора на нашей синьке что-то изобразить. Можно начать с перетаскивания прямоугольника для обрамления области, покрытой нашим сырным фоном. Именно эта фигура часто встречается на технических чертежах и синьках. Кстати, вы можете добавить стопку прямоугольничков в нижнем правом углу, на манер штампа. Откройте свойства *Fill and Stroke* [Заливка и штрихи] (*Ctrl+Shift+F*) и выставьте их в *No Fill* [Без заливки] и *Solid stroke* [Сплошной штрих], примерно на 10% серого. Толщину настройте на свое усмотрение.

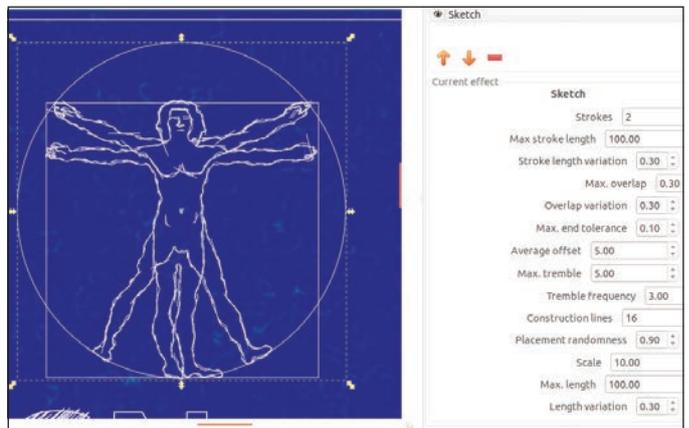
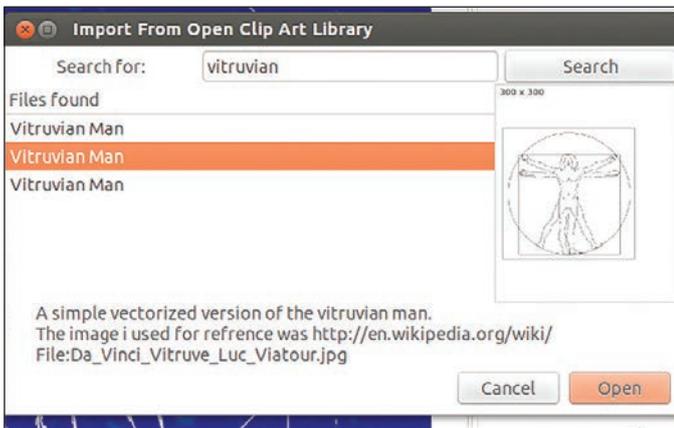


5 Что-нибудь добавим

Теперь у нас есть шаблон листа для рисунков (хорошей идеей будет сохранить его для будущего употребления), и можно начать рисовать на нем. Если у вас уже есть псевдокарандашные рисунки, которые вы бы хотели добавить, самое время сделать это, но если у вас нет ничего, достаточно похожего на технический чертеж, несколько сочетаний клавиш вам помогут. Инструментом Карандаш (Shift+F6) нарисуйте треугольник. Это станет намного проще, если удерживать клавишу Ctrl, которая поможет проводить прямые линии и делать ровные углы.

6 Трансформируем с Render

Выделив треугольник, выберите Extensions > Render > Draw From Triangle... [Расширения > Рендеринг > Рисовать из треугольника] — появится окно с массой опций по геометрическим фигурам, создаваемым из треугольника. Поэкспериментируйте с ними: щелкните по Live preview [Предпросмотр] — и увидите, на что способна каждая из них. Это демонстрирует мощь некоторых расширений, которые умеют рисовать объекты за вас или расширять то, что нарисовали вы. Внимательно ознакомьтесь с меню Render [Рендеринг]: в нем есть еще много чего интересного!



7 Импортируем что-нибудь еще

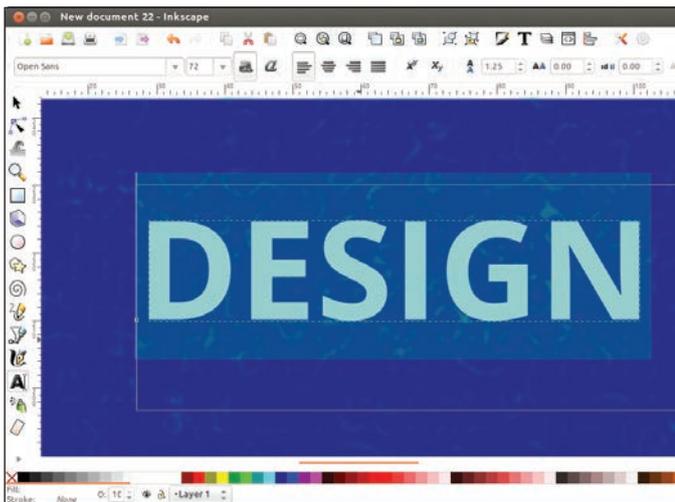
Если вы не умеете рисовать, а расширения не предлагают вам ничего подходящего, выход есть: одолжите рисунок! При выборе File > Import from Open Clip Art library [Файл > Импортировать из библиотеки клипарта] перед вами появится окно с параметрами поиска, по которым вы найдете любые изображения от Creative Commons.

Мне подумалось, что неплохо будет посмотреться витрувианский человек Леонардо да Винчи, и тут же нашлось аж три таких. Наш предоставлен пользователем YDL, и он выглядит очень подходяще для того, что мы намерены сделать далее — заставим его выглядеть довольно неряшливо! После его импорта может понадобиться изменить размер и пристроить изображение на должную позицию.

8 Придадим вид эскиза

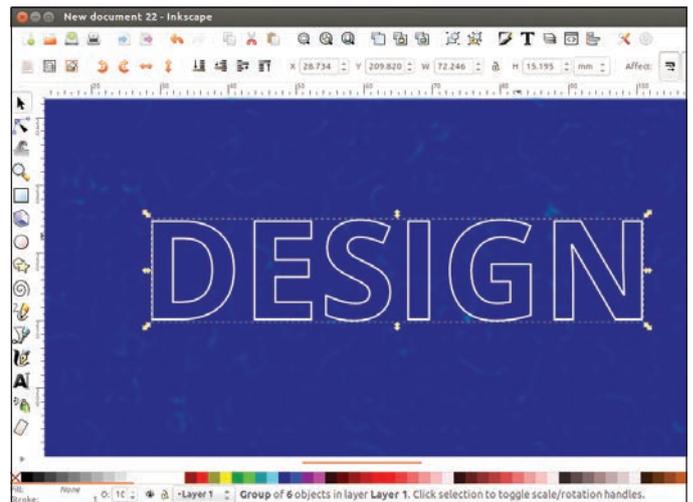
Одна из функций *Inkscape*, которой уделяется мало внимания — это эффекты Path [Контур]. Они отчасти похожи на фильтры или расширения, но изменяют контур. Здесь есть из чего выбирать, но сейчас мы хотим использовать Sketch [Набросок]. Выделив объект, нажмите на Shift+Ctrl+7, чтобы вывести диалоговое окно Path Effect [Эффект контура]. Выберите Sketch из всплывающего меню, затем нажмите на кнопку Add [Добавить]. Она изменит контур, заменив каждую кривую неким количеством линий, сходных с оригиналом. Повозитесь с настройками, пока не добьетесь нужного эффекта — но не добавляйте слишком много «дрожания» или линий, иначе рисунок превратится в полную неразбериху.

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)



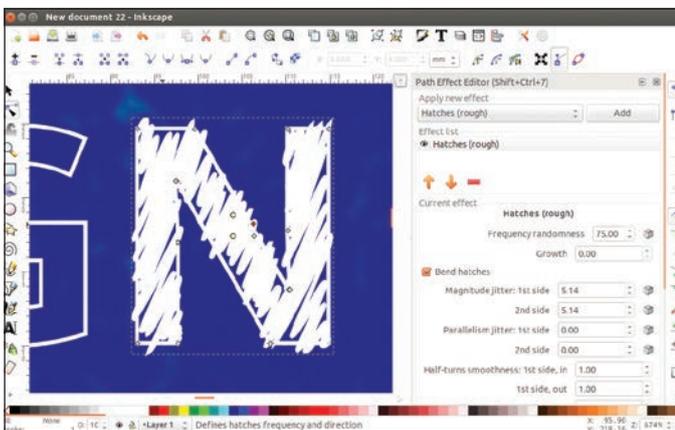
9 Добавим текст

А теперь нашему документу нужен текст, но не какой попало. Можно потратить вечность на поиск подходящего шрифта, напоминающего человеческий почерк, но вообще-то подойдет любой шрифт; мы заставим его выглядеть рукописным с помощью эффектов. Выберите шрифт и впишите текст. Я советую брусчатый шрифт вроде OpenSans, или, по крайней мере, шрифт без засечек. Добавьте его с помощью инструмента Text [Текст], и позаботьтесь выделить опцию Bold [Жирный] в панели инструментов — чем массивнее он будет, тем лучше для этого эффекта.



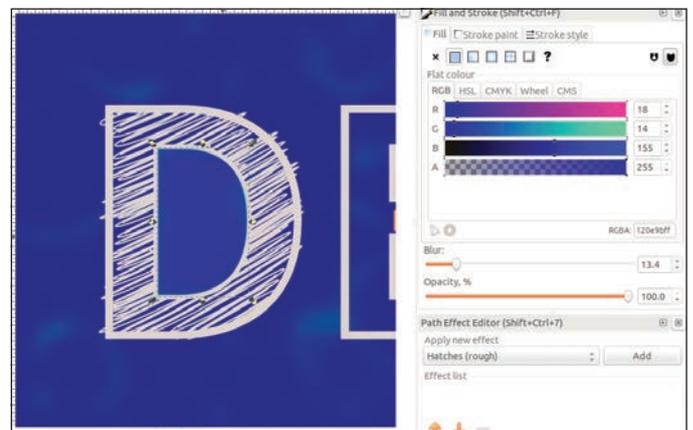
10 Преобразуем текст

Применить к тексту эффекты Path сразу нельзя, поскольку текст все-таки не контур. Зато можно превратить текст в контур. Выделите его и затем выберите Path > Object To Path [Контур > Преобразовать объект в контур]. Отныне ваш текст уже не доступен в виде текста, он стал сгруппированной подборкой контуров. Пока они сгруппированы, используйте свойства Fill and Stroke, чтобы придать им серый цвет и отсутствие заливки. Теперь выберите Edit > Duplicate [Редактировать > Дублировать] для создания точной копии. Именно копию мы превратим в каракули, а оригинал останется как был, чтобы текст оставался читаемым после того, как мы добавим цвет. Разгруппируйте контуры, выберите для затравки какой-нибудь симпатичный и несложный, затем нажмите Shift+Ctrl+7, чтобы снова открыть Path Effects Editor [Редактор эффектов контуров].



11 Превратим текст в каракули

На этот раз выберите и добавьте Hatches (rough) [Штриховка (грубо)]. Не исключено, что фигура будет выглядеть заляпанной. На панели Fill and Stroke проверьте, что Fill отключена, и уменьшайте толщину штриха, пока линии не станут различимы. Можно настроить штриховку через длинный список опций, но намного проще добиться желаемого эффекта в режиме 'live'. Переключитесь на инструмент Node [Узел] (тот, который под обычным указателем), и теперь в фигуре вы должны видеть два желтых ромба и несколько других точек. Поверните верхним ромбом, меняя густоту и наклон штриховки, чтобы добиться нужного эффекта (если тащить ромб к центру объекта, густота штриховки увеличивается).



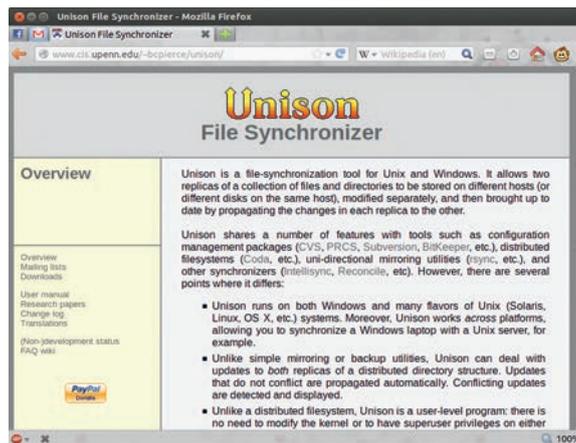
12 Работаем с хитроумными фигурами

В некоторых фигурах есть отверстия, и это проблема. Эффект Hatches не слишком хорошо с таким справляется и как бы заполняет их дважды. Несколько нелепый, но эффективный способ справиться с проблемой — снова продублировать фигуру. С помощью редактора Node удалите внешний контур, оставив только внутренний, затем заполните его и примените любые эффекты, чтобы он выглядел так же, как фон. Это довольно утомительно, но зато эффективно, и уж всяко проще, чем рисовать все эти каракули самому! **LXF**

» Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

Unison: Доступ

Хотите разделять свои данные между разными системами без проблем с приватностью и собственностью на них? Джеймс Литтон показывает, как.



Домашняя страница проекта Unison.

Предположим, что общий объем ваших данных — 500 ГБ, и вы хотите собрать их все в одном месте. Такое позволяет сделать Dropbox — за \$499 в год, и это, честно говоря, довольно разумная цена, но многим из нас не нравится то, что их данные будут находиться на серверах Amazon S3 и выйдут из-под контроля. Данные можно зашифровать, но с их конфиденциальностью и собственностью на них не все так просто. Чтобы избавиться от этих недостатков, исследуем варианты построения собственной системы.

Альтернативные решения

На первый взгляд, существуют относительно простые для реализации решения, такие как OwnCloud и Sparkleshare. Оба выглядят простейшим способом создать полноценную замену Dropbox, но если подумать о задачах этого проекта, то нам вовсе не нужно реализовывать всю функциональность Dropbox. Другими словами, нам не нужны ни web-интерфейс, ни возможность делиться файлами с другими, а нужна только возможность выборочно разделять файлы между компьютерами. То есть лучше бы взять что-нибудь полегче Sparkleshare или OwnCloud.

Учитывая простоту и возможности решений для командной строки, предоставляемые *rsync*, эта утилита приходит на ум первой. Однако проблема *rsync* в том, что она не приспособлена для двусторонней синхронизации. У нее есть несколько полезных опций, позволяющих довольно близко подобраться к двусторонней синхронизации, но достичь таковой полностью нельзя. Хорошая иллюстрация — отсутствие эффективного способа реакции на удаление файлов. Существуют способы это обойти, которые включают сохранение вывода в лог-файлы, сравнение и удаление, но это излишне усложняет дело.

Если необходим простой сервер для резервного копирования данных, находящихся на одном компьютере, *rsync* прекрасно подойдет, но нам нужно нечто более интеллектуальное, поэтому мы обратимся к *Unison*. *Unison* — утилита синхронизации



Наш эксперт

Джеймс Литтон занимал руководящие должности в нескольких организациях и является партнером и директором Identity Automation LP.

При мыслях о своих данных у многих из нас возникает внутренняя борьба: мы подыскиваем наилучший способ гарантировать их сохранность. В прежние времена все данные умещались на нескольких дискетах; потом мы перешли к лентам, CD, DVD, NAS и, наконец, к облачным хранилищам.

Рассмотрев типичную стратегию управления данными, мы обнаружим, что зачастую она фрагментарна и неэффективна. Ваши повседневные файлы, возможно, хранятся на Dropbox, что позволяет легко разделять их между разными компьютерами, которыми вы регулярно пользуетесь. Этот вариант прекрасно подходит для электронных таблиц, баз данных и документов — все вместе это довольно скромный объем данных. Но когда доходит до фотографий, музыки и фильмов, начинаются проблемы. Для файлов мультимедиа нужно гораздо больше места, и они часто оказываются на домашнем сервере NAS, и, возможно, время от времени копируются на внешний USB-диск.

Системные каталоги Unison

В Linux *Unison* создаст подкаталог `.unison` в `$HOME`. В OS X системный каталог создается в `~/Library/Application Support/Unison`. Многие пользователи Mac предпочтут системный каталог в стиле Linux.

К счастью, *Unison* будет пользоваться каталогом `~/.unison`, если он существует. Чтобы его создать, выполните следующую команду:

```
mkdir ~/.unison
```

К СВОИМ файлам

файлов, работающая в Linux, BSD, OS X и Windows. Она позволяет синхронизировать набор файлов и каталогов, которые хранятся в разных местах, путем распространения изменений между наборами.

Прежде чем рассуждать о стратегии, важно понять, как именно работает *Unison*. На самом базовом уровне *Unison* поддерживает пары реплик синхронизированными. Уловили слово «пары»? При каждом запуске *Unison* он синхронизирует две реплики, и ничего больше. Если реплик больше двух, потребуется хорошо продуманная схема, которая гарантирует, что данные синхронизируются именно так, как мы воображаем.

Реплики

Теперь уместно спросить: «А что такое реплика?» По-простому, реплика — это набор файлов и каталогов. Реплики могут соответствовать разным хостам, но это не обязательно. Чтобы лучше понять идею, представьте, что у вас есть компьютер дома и компьютер в офисе, и вы хотите, чтобы домашний каталог автоматически синхронизировался между этими компьютерами. Пусть у вас также есть внешние USB-диски для резервных копий дома и в офисе. В этот сценарий вовлечено всего два компьютера, но реплик *Unison* четыре.

Задумавшись о том, какую структуру синхронизации выбрать, вы получите различные «топологии», которые похожи на известные сетевые топологии — линия, звезда, сетка, кольцо и другие. Описанный выше сценарий соответствует линейной модели, но лучший способ организации нашей системы — с помощью хаба и модели «звезда». В такой конфигурации в центре находится сервер, и на каждом из компьютеров имеется один профиль *Unison*, который синхронизирует данные с сервером; эту схему мы и реализуем на нашем уроке.

Для нашего тестового случая рассмотрим ситуацию, при которой на регулярной основе используются четыре компьютера: офисный настольный компьютер, домашний настольный компьютер, домашний ноутбук и ноутбук для поездок. Мы заменим Dropbox на всех этих четырех компьютерах. На двух компьютерах установлена Mac OS X, на двух других — Linux.

При работе с *Unison* важно обеспечить, чтобы на всех системах, участвующих в синхронизации, была установлена одна и та же версия программы. Воспользуемся версией 2.40, это последний стабильный релиз.

На платформе Mac загрузите графическую оболочку версии 2.40.69 с сайта <http://alan.petitepomme.net/unison/index.html>. Смонтируйте **dmg** и скопируйте *Unison* в каталог приложений. Теперь откройте *Unison*, и программа попросит вас установить версию утилиты для командной строки — скажите «да». Утилита установится в **/usr/bin/**. Для проверки выполните команду

```
unison -version
```

Unison должен вывести номер версии — 2.40.69. В Ubuntu 12.04 *Unison* можно установить из репозитория Ubuntu, скомандовав

```
sudo apt-get install unison
```

Генерация пары ключей

Важно понимать, что подключение по SSH без пароля несет в себе элемент риска. Если доверенный компьютер будет скомпрометирован, злоумышленник сможет получить прямой доступ к серверу. Чтобы управлять этим риском, лучше всего создать учетную запись на сервере специально для синхронизации *Unison*, у которого будет очень ограниченный доступ к другим сервисам.

Для генерации пары ключей запустите **ssh-keygen**, и на экране появятся несколько сообщений, требующих ввода данных.

Enter file in which to save the key [Введите имя файла, где сохранится ключ]:

Можно оставить вариант по умолчанию, просто нажав Enter.

Enter passphrase [Введите парольную фразу]:

Мы хотим пользоваться SSH без пароля, поэтому парольная фраза не нужна. Просто нажмите Enter.

Enter same passphrase again [Введите парольную фразу еще раз]:

Подтвердите парольную фразу нажатием Enter.

Проверить установку можно командой

```
unison -version
```

Программа должна вывести номер версии — 2.40.65. Теперь можно создать профиль *Unison*. Профиль определяет, как *Unison* будет синхронизировать две реплики. Создадим файл профиля под названием **lxfsync.prf** в нашем системном каталоге *Unison*:

```
root = /home/lxf/Unison
```

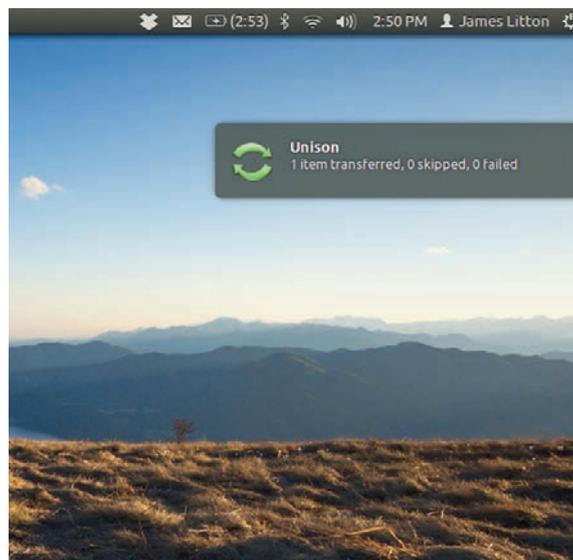
```
root = ssh://lxf@example.com//home/lxf/Unison
```

```
fastcheck = true
```

```
batch = true
```

```
force = newer
```

»



» Оповещение о результатах синхронизации на рабочем столе.

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

```
times = true
sshargs = -C
```

Строки 1 и 2 задают местоположение реплик. Строка 3 велит *Unison* применять время изменения и размер файла как «псевдономер узла» при сканировании реплик на наличие изменений, вместо считывания содержимого каждого файла целиком. Строка 4 отключает интерфейс пользователя. Строка 5 велит *Unison* брать файл с последним временем изменения. Строка 6 означает, что будет сохраняться время изменения файлов; и, наконец, в строке 7 включается сжатие SSH для увеличения производительности. Замечание об опции `times`: увы, она не поддерживает каталоги. При копировании каталога время его создания будет соответствовать созданию каталога копии, а не исходного каталога.

Теперь мы готовы к первичной синхронизации. Для начала выберите один из компьютеров, создайте копию каталога **Dropbox** и назовите ее **Unison**.

```
cp -r ~/Dropbox ~/Unison
```

Затем мы подключимся к нашему серверу *Unison* через SSH и создадим соответствующий каталог, в который будут синхронизироваться наши четыре компьютера.

```
mkdir ~/Unison
```

Синхронизируем наши данные

Теперь нужно переместить копию данных с ноутбука на сервер. Чтобы убедиться в шифровании данных при передаче между компьютерами, взгляните на параметры: с их помощью мы указали, что для подключения к удаленной реплике нужно пользоваться SSH. Синхронизация запускается следующей командой:

```
unison -ui text lxfsync
```

Первые два параметра велют *Unison* запускаться в текстовом режиме. Третий параметр — имя профиля, который мы хотим использовать.

По окончании синхронизации *Unison* должен сообщить количество переданных элементов. Если некоторые элементы были пропущены или возникли ошибки, подробную информацию о транзакции можно получить, заглянув в файл журнала — `~/unison.log`. При запуске *Unison* вы видели, что вас просят ввести пароль учетной записи на сервере. А так как мы собираемся автоматизировать этот процесс, аутентификация должна проходить без вашего вмешательства.

Чтобы достичь этой цели, воспользуемся аутентификацией SSH по ключу. Для этого на каждом компьютере, синхронизируемом с сервером, нужно сгенерировать уникальный ключ. Чтобы сделать это на ноутбуке, выполните следующую команду:

```
ssh-keygen -t rsa
```

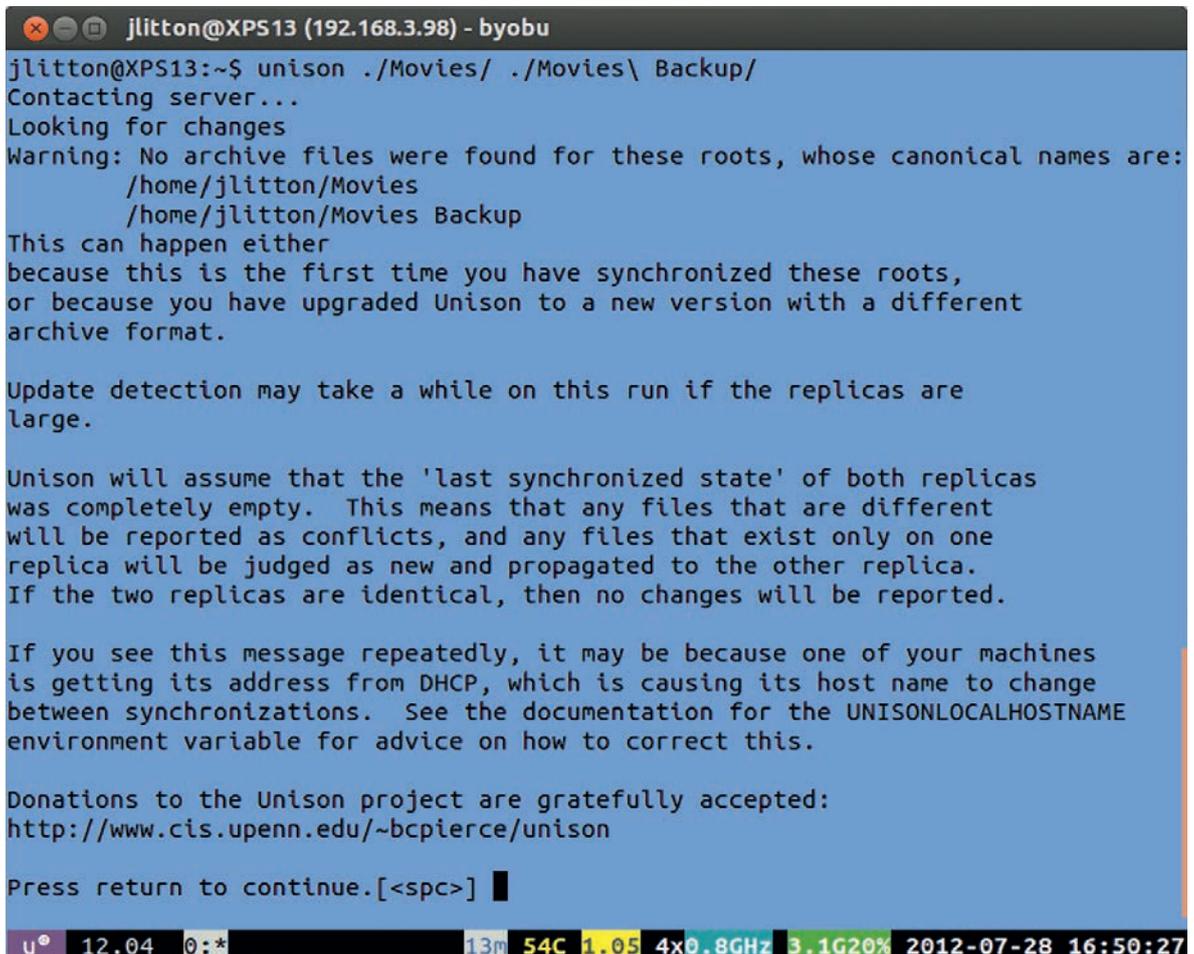
После этого в каталоге `~/ssh` должно появиться два файла: `id_rsa` и `id_rsa.pub`. Файл `id_rsa.pub` — наш публичный ключ, и его нужно скопировать на сервер, чтобы демон SSH мог доверять этому ключу. Переместим файл командой `scp`:

```
scp ~/.ssh/id_rsa.pub lxf@example.com:~/ssh/Computer1.pub
```

Теперь нужно предупредить демон SSH на сервере, чтобы он доверял этому ключу. Для этого подключимся к серверу по SSH и скомандуем:

```
cat ~/.ssh/Computer1.pub >> authorized_keys
```

Чтобы проверить нашу работу, снова запустим команду *Unison* с ноутбука. Если все в порядке, на сей раз у нас не спросят пароль:



➤ При первом запуске *Unison* на реплике будет выведена эта полезная информация.

➤ Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

```
unison -ui text lxfsync
```

Хотя это и не обязательно, многим нравится, что агент Dropbox информирует их о происшедших изменениях; перенесем эту возможность на четыре наших компьютера, синхронизируемые с сервером. Так как синхронизация производится с помощью скриптов оболочки, нужно установить утилиты, которые позволяют обратиться к системе оповещений рабочего стола. На платформе Mac можно воспользоваться *Growl* посредством **growlnotify**. На компьютерах Ubuntu воспользуемся *NotifyOSD* с помощью **notify-send**.

На платформе Mac загрузите *Growl 1.2.2* с growl.info. После загрузки смонтируйте **dmg** и установите *Growl*, дважды щелкнув на скрипте **Growl.pkg**. Откройте каталог **Extras**, затем **growlnotify**, и установите *growlnotify*, запустив скрипт **growlnotify.pkg**. Теперь можно отправлять оповещения *Growl* с командной строки. Чтобы это проверить, откройте терминал и выполните команду

```
growlnotify --message "Hello World"
```

В Ubuntu для передачи сообщений демону оповещений рабочего стола нужно установить библиотеку *libnotify*. Установить ее можно из репозитория Ubuntu, командой

```
sudo apt-get install libnotify-bin
```

Чтобы ее проверить, попробуйте выполнить следующую команду:

```
notify-send "Hello World"
```

Если все хорошо, ваши сообщения появятся на соответствующих рабочих столах.

Объединяем все вместе

Теперь мы умеем синхронизировать данные и отправлять оповещения. Давайте удобства ради объединим все эти действия в одном скрипте.

```
#!/bin/bash
/bin/ping -c 1 www.google.com 2>&1 >/dev/null
if [ "$?" == 0 ]; then
root="/home/lxf"
image="$root/unison/sync.png"
stat="/usr/bin/unison -ui text -logfile "$root"/unison.log" lxfsync 2>&1 >/dev/null | tail -1`
if [ ${PIPESTATUS[0]} != 0 ]; then
notify-send "Unison" "Sync Error" -i "$image" -t 0
exit 1
fi
if [ ${stst:0:7} == "Nothing" ]; then
exit 0
else
stat2=`echo "$stat" | grep -Po '(?<=\.).*?(?=\))'`
fi
notify-send "Unison" "$stat2" -i "$image"
else
exit 1
fi
```

Как видите, наш скрипт очень прост. В строке 1 задается тип скрипта (*Bash*). В строке 2 проверяется подключение к Интернету. В строке 3 анализируется результат проверки, и если подключения нет, мы переходим к строке 18, в которой скрипт завершается с кодом ошибки, означающим, что синхронизация завершилась неудачно.

В том случае, если подключение доступно, в строках 4 и 5 задаются несколько переменных, после чего мы переходим к строке 6. В данной строке мы запускаем *Unison*, получаем последнюю строку вывода и сохраняем ее в переменной **\$stat**, которой воспользуемся позже. В строке 7 мы проверяем, вернул ли *Unison* код 0, означающий успешную синхронизацию. Если мы получили другой код ошибки, то в строке 8 мы отправляем оповещение рабочего стола об ошибке синхронизации.

Ярлык для Unity Launcher

Иногда вам захочется запустить процесс синхронизации вручную, а не ждать, пока это сделает *cron*. Например, вы просидели над документом весь рабочий день, но потом хотите еще поработать с ним из дома. Прямо перед выключением системы можно запустить скрипт синхронизации и выполнить принудительное обновление, и дома вы сможете начать прямо с обновленного документа.

Открывать терминал каждый раз для запуска скрипта неудобно; лучше всего создать иконку и добавить ее в Unity

Launcher. Для этого создайте файл **sync.desktop** в **/usr/share/applications/**:

```
[Desktop Entry]
Version 1.0
Name=Unison
Type=Application
Categories=Application
Exec=/home/lxf/sync.sh
Terminal=true
Icon=/home/lxf/unison/sync.png
```

Теперь наше приложение появится в Unity, и его легко подключить к Unity Launcher.

При успешном завершении синхронизации в строке 11 мы анализируем переменную **\$stat**, чтобы понять, выполнил ли *Unison* какие-либо действия. Если первое слово в переменной — "Nothing [Ничего]", то мы выходим из скрипта, потому что обновлений не было; в противном случае переходим к строке 14, где получаем строку в круглых скобках и сохраняем ее в переменной **\$stat2**, после чего переходим в строку 16 и отображаем результаты пользователю.

Чтобы все работало без нашего участия, скрипт должен запускаться регулярно и синхронизироваться с сервером. С этой задачей прекрасно справится *cron*. Давайте сделаем интервал синхронизации в пять минут следующей командой:

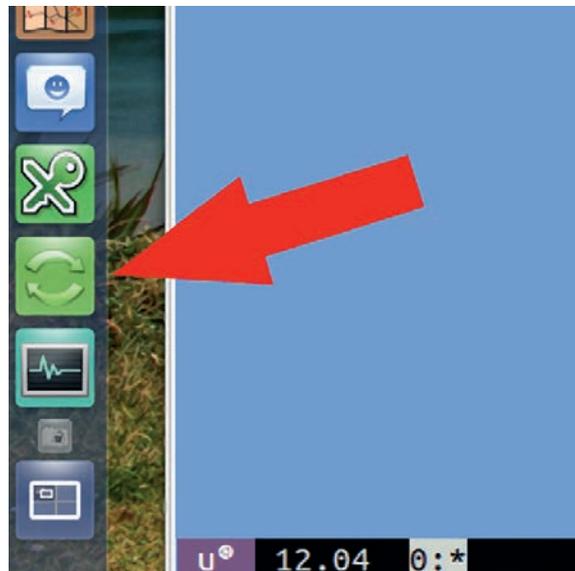
```
crontab -e
```

Теперь добавим следующее:

```
*/5 * * * * DISPLAY=:0 /home/lxf/sync.sh
```

***/5** велит *cron* запускаться каждые пять минут. Следующие четыре звездочки велют *cron* запускаться каждые пять минут каждого часа каждого дня каждого месяца. **DISPLAY=:0** направляет вывод скрипта на текущий монитор.

Вот и все! Теперь вы обзавелись бесплатной заменой Dropbox, которая находится под вашим контролем. Некоторые из вас, несомненно, заинтересуются: а как же лучше всего делиться файлами с членами семьи и друзьями? Здесь лучше Dropbox вам не найти. Продолжайте пользоваться бесплатной учетной записью и делиться файлами по мере необходимости, не беспокоясь об остальных данных. **LXF**



► При создании этого файла **.desktop** появится иконка, которую можно подключить к Unity Launcher.

Turnkey Linux:

Маянк Шарма хочет ухитриться и web-приложения настроить, и пальчики при этом не обжечь.



Наш эксперт

Фанат открытого ПО **Маянк Шарма** был пишущим редактором [Linux.com](#) и печатался в [LinuxToday](#), [Digg](#) и [PC Plus](#).



Разворачивать web-приложения с сетевым доступом на web-сервере — вариант не для каждого. Во-первых, нужно отлично знать конфигурацию своей сети. Вдобавок web-приложениям нужна куча инфраструктурных программ, от серверов баз данных до библиотек, и настраивать их совместную работу можно часами. А если пользователи приложения будут обращаться к нему извне, придется тщательно исследовать конфигурацию web-сервера на предмет дыр в безопасности.

И здесь на помощь приходит Turnkey Linux. Из его виртуальных модулей новый сервер разворачивается почти мгновенно. Виртуальный модуль Turnkey — самодостаточная система, содержащая полнофункциональный экземпляр web-приложения с урезанным набором компонентов операционной системы. Система работает из коробки, и ее можно развернуть на «голом железе» или виртуальном устройстве.

Turnkey Linux использует JeOS (Just Enough Operating System — минимально необходимую ОС), это позволяет уменьшить размер образа. На данный момент все модули Turnkey Linux базируются на Debian 6.0.5 Squeeze. Эта база, с некоторыми дополнительными приложениями, которых нет в официальных репозиториях Debian, образует основной виртуальный модуль Turnkey Linux — Core (см. врезку), на котором строятся все остальные модули.

Например, в виртуальном модуле *WordPress* используется последняя версия *WordPress* в минимальной системе Debian. Чтобы управлять CMS было проще, в виртуальной системе также есть механизм загрузки последних обновлений.

Для удобства в виртуальный модуль также введено больше десятка популярных плагинов *WordPress*, и, разумеется, можно установить еще, как и в любой другой установке *WordPress*. Разработчики также позволяют настроить автоматическое обновление плагинов в виртуальном модуле. Удобнее всего то, что управлять виртуальным модулем можно через SSL-подключение.

В четвертую годовщину Turnkey Linux вышел релиз 12.0 библиотеки виртуальных модулей, и теперь он включает более 100 web-приложений. В дополнение к *WordPress* поддерживаются другие популярные и не очень популярные системы управления контентом, такие как *Joomla*, *Drupal*, *Concrete5*, *Elgg*, *Clipbucket* и т.д.

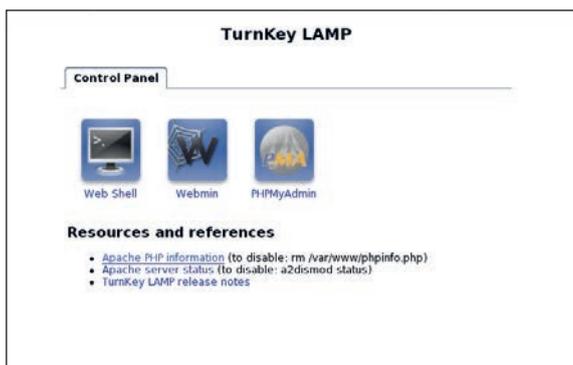
Есть также модули, являющиеся прекрасными платформами для разработки приложений; например, стек LAMP, модуль .NET для развертывания web-приложений .NET в Linux и различные версии SDK Google AppEngine. Если вам интересно, все модули Turnkey Linux на 100 % состоят из открытого кода и бесплатны для загрузки.

Средства ремесла

Модули Turnkey Linux доступны в различных форматах в зависимости от устройств, на которых вы хотите их разворачивать. Однако важно то, что после настройки и запуска они предоставляют вам одинаковый интерфейс для развертывания и управления web-приложением. На данный момент каждый виртуальный модуль Turnkey Linux доступен в шести форматах. Из модуля в формате ISO можно установить виртуальный модуль на физический компьютер, а также на виртуальные машины. Это образ Live CD, в котором для установки системы используется модифицированный установщик Debian.

Хотя ISO-образ пригоден для установки на виртуальную машину, он не оптимизирован для виртуальных устройств. Лучше будет взять образ либо VMDK, либо OVF. Первый предназначен для устройств с расширениями виртуализации, например, Intel VT-x, а второй поладит и с более старыми устройствами. И образы OVF, и образы VMDK будут работать в бесплатных программах виртуализации *VirtualBox* и *VMware Player*. Сборки OpenStack, как следует из названия, оптимизированы для развертывания на облачной платформе OpenStack. Аналогично, сборки OpenVZ и Xen — для соответствующих виртуализационных сред.

Размеры модулей различны, но большинство из них менее 300 МБ, а ISO-образы меньше, чем образы соответствующих виртуальных решений.



» В некоторых модулях есть панель управления со ссылками для проверки конфигурации сервера Apache и его статуса.

Web-модули

Шаг за шагом: Установим виртуальный модуль



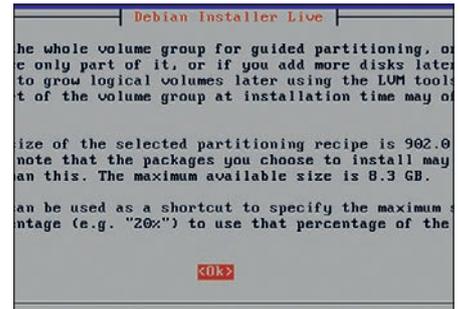
1 Загрузка с Live CD

Если у вас ISO-образ, модуль можно либо установить сразу, либо сперва опробовать, загрузив в Live-режиме. Установщик представляет собой подправленный установщик Debian.



2 Разбиение диска на разделы

Единственное решение, которое нужно принять во время установки — как разбивать диск для установки модуля. Если вы не знаете, что делать, оставьте эту задачу установщику.



3 Настройка LVM

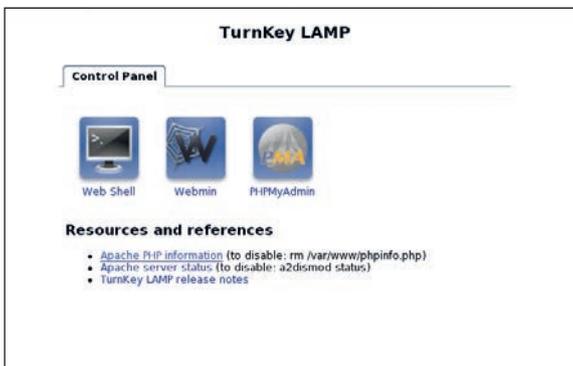
По умолчанию установщик Turnkey Linux предлагает создать на диске раздел LVM, преимущество которого в том, что его размер можно потом увеличить, добавив новые диски.

Гаджет, вперед!

Первый шаг после загрузки модуля из репозитория Turnkey Linux — запустить его. Этот шаг зависит от типа виртуального модуля.

Если вы загрузили ISO-образ, запишите его на диск и загрузитесь с него. После этого следуйте инструкциям из врезки «Установим виртуальный модуль» для установки и настройки модуля.

Образ OVF можно импортировать в VirtualBox, зайдя в File > Import Appliance [Файл > Импортировать модуль]; при этом создается виртуальная машина с настройками из модуля. А если у вас образ VMDK, сначала нужно создать виртуальную машину в VirtualBox, как обычно, но вместо создания нового виртуального диска выбрать имеющийся образ и указать загруженный файл образа VMDK.



» Хотя аппаратные требования модулей Turnkey Linux минимальны, нужно выделить достаточное количество ресурсов.

После этого запускайте виртуальную машину. Неважно, устанавливали ли вы модуль из файла ISO или воспользовались готовым: процесс настройки один. Он описан во врезке «Настроим виртуальный модуль».

Настроенный модуль автоматически загрузится, и появится консоль настройки со ссылками на утилиты и сервисы, запущенные в виртуальном модуле.

Во всех модулях есть как минимум две утилиты. Web-оболочка вызывает эмулятор терминала *Shell-in-a-Box* на основе AJAX, с помощью которого осуществляется вход на удаленный сервер из браузера.

Управление через Webmin

Кроме того, есть web-утилита настройки *Webmin* — с ее помощью можно менять все настройки на удаленном сервере: например, настраивать задания *cron*, просматривать лог-файлы и управлять выполняющимися процессами. Если в вашем модуле есть приложение, работающее с *MySQL*, в нем также будет *PHPMyAdmin* для администрирования баз данных *MySQL*.

Консоль также выводит информацию по подключению к модулю через SSH/SFTP и адрес, по которому запущено приложение, построенное на базе этого виртуального модуля. По этому адресу к приложению можно обратиться из любой точки сети и настроить его точно так же, как если бы вы установили его вручную.

Turnkey Linux не модифицирует приложения, и при необходимости можно обратиться к официальной документации любого приложения.

Если надо изменить файл при настройке web-приложения, можно подключиться к модулю по SSH или воспользоваться web-оболочкой через HTTPS. Чтобы скопировать файл в модуль, »



В канале Turnkey Linux на YouTube есть много роликов с инструкциями: www.youtube.com/turnkeylinux.

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

TurnKey Core

Все виртуальные модули Turnkey Linux строятся поверх модуля Turnkey Core. Кроме дистрибутива Debian 6.0.5, в нем есть все компоненты, необходимые всем прочим модулям.

В модуле Core также настроена автоматическая загрузка обновлений безопасности для установленных компонентов из официальных репозитория Debian. По умолчанию Turnkey попросит обновиться во время установки, а также каждую ночь в 4 часа утра. Также в модуле Core есть все специализированные ути-

литы Turnkey Linux для резервного копирования и миграции серверов, web-оболочка AJAX и интерфейс удаленного управления Webmin.

Хотя модуль Core доступен для загрузки, большинству пользователей он будет не слишком интересен. Если вы хотите развернуть собственное web-приложение, лучше воспользоваться готовым модулем, таким как LAMP или LAPP. К модулю Core лучше всего применять набор скриптов TKLPatch для настройки модуля.

просто подключитесь к нему любым FTP-клиентом, используя пароль администратора, заданный при настройке модуля.



Скорая помощь
Убедитесь, что в виртуальной машине настроен адаптер «мост», чтобы можно было обращаться к ней с любого компьютера сети.

У хаба

Наряду с самими модулями в Turnkey Linux есть несколько специальных утилит, которые помогут управлять модулями. Главная — Turnkey Hub. С ее помощью можно запускать модули Turnkey на облаке Amazon EC2 и управлять ими. В модулях, созданных на EC2 с помощью Hub, используется оптимизированная, но редактируемая политика брандмауэра. По умолчанию разрешен трафик только через порты, используемые приложением.

С помощью Hub также можно наблюдать за удаленными серверами и отслеживать использование процессора, операции ввода/вывода и сетевой трафик. Hub также упрощает более сложные операции, например, подключение дисков NAS к удаленным облачным серверам.

Есть еще утилита резервного копирования и миграции — Turnkey Linux Backup and Migration (TKLBAM). По умолчанию она предназначена для работы с облачным сервисом Amazon S3. Настроить поведение утилиты можно в модуле *Webmin Backup and Migration* [Резервное копирование и миграция], а просматривать резервные копии и управлять ими — из Hub.

И Hub, и TKLBAM бесплатны, но для резервного копирования на Amazon S3 нужно зарегистрироваться на облачном сервисе. Однако сервис довольно дешев: расходы составят не более £0,07 за ГБ в месяц, а новые пользователи получают 5 ГБ бесплатно.

Кроме того, с помощью TKLBAM можно вручную создавать и восстанавливать резервные копии файлов локально. Пусть вы подключили локальный диск к виртуальному модулю в точке `/mnt/backupdrive`; запустите оболочку и введите следующие команды:

```
$ tklbam-backup --address file:///mnt/backupdrive/tklbam/backup
$ tklbam-escrow /mnt/backupdrive/tklbam/key
```

Ключ *escrow* [залоговый] можно использовать для аутентификации и восстановления резервной копии даже в другом модуле TurnKey таким образом:

```
$ tklbam-restore --address file:///mnt/backupdrive/tklbam/backup
--keyfile=/mnt/backupdrive/tklbam/key
```

Не теряйте ключ *escrow*. С его помощью можно восстановить резервную копию, даже если вы забыли парольную фразу зашифрованных резервных копий.

Еще один туз в рукаве Turnkey Linux — его бесплатный сервис динамического DNS tklapp.com. Он позволяет обращаться к модулю извне сети и дает ему удобное для запоминания имя.

Для настройки сервиса используется утилита HubDNS, которая установлена во всех модулях Turnkey Linux. Перед настройкой зайдите в свою учетную запись Turnkey Hub и перепишите свой ключ Hub API в вкладке User Profile [Профиль пользователя]. Затем откройте терминал и введите следующие команды:

```
$ hubdns-init HUB_APIKEY foo.tklapp.com
$ hubdns-update
```

Готово. Замените HUB_APIKEY на свой ключ, а foo — на желаемое имя сервера. Если оно доступно, HubDNS зарезервирует его для вас. Вторая команда свяжет ваш IP-адрес с созданным именем домена. Давайте воспользуемся модулем Torrent Server и вернем старый ненужный компьютер к жизни, превратив его в торрент-сервер без монитора.

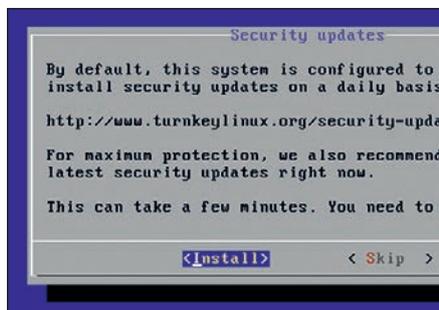
Загрузите ISO-образ, запишите его на компакт-диск и установите его на компьютере с помощью установщика. Теперь запустите браузер на другом компьютере той же сети и введите в адресную строку IP-адрес сервера. Откроется панель управления торрент-сервера.

Шаг за шагом: Настроим виртуальный модуль



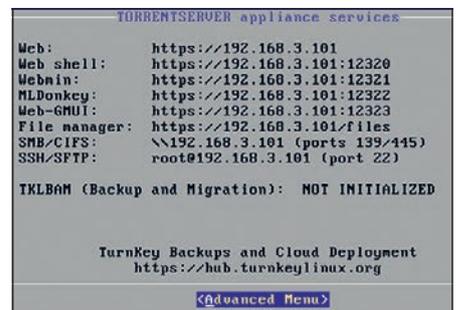
1 Установка паролей

При настройке модуля вас попросят задать пароль пользователя root модуля и пароли администратора для отдельных компонентов, таких как MySQL.



2 Установка обновлений безопасности

Этап с запросом ключа API Hub можно опустить. Установку обновлений безопасности тоже можно опустить, но если у вас есть подключение к Интернету, делать этого не стоит.

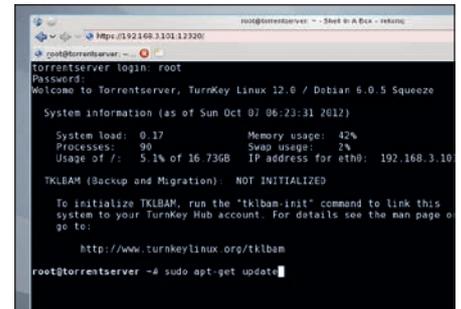
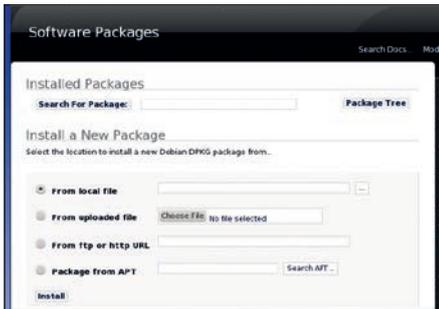


3 Сервер запущен

Вот и все. После настройки модуля откроется консоль настройки со списком адресов различных утилит и сервисов, запущенных в модуле.

» Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

Шаг за шагом: Добавим программы



1 Обновление списка пакетов

Откройте интерфейс *Webmin* и зайдите в System > Software packages [Система > Пакеты программ], выберите Re-synchronise package list [Пересинхронизировать список пакетов] внизу и нажмите кнопку Upgrade now [Обновить сейчас].

2 Установка пакетов

После обновления списка пакетов поищите в нем пакеты, которые хотите установить. В списке появятся пакеты из репозитория Debian, соответствующие поисковому запросу.

3 Установка командами

Опытные пользователи могут устанавливать пакеты и обновлять список пакетов командами `apt-get update` и затем `apt-get install <packagename>` в web-оболочке — как в обычном дистрибутиве Debian.

Изюминка этого модуля — браузерный файловый менеджер AjaXplorer, в котором можно работать с файлами и каталогами. У приложения удобный интерфейс, и есть мобильные версии для iOS и Android.

Трудолюбивый ослик

Чтобы загружать торренты, нужно подключить модуль к своему экземпляру *MlDonkey* [donkey — англ. ослик]. Для этого зайдите в интерфейс Basic [Основной], перейдите на вкладку Clients [Клиенты] и нажмите кнопку Connect [Подключить] рядом с сервером, запущенном на 127.0.0.1:4001.

Теперь откройте свой любимый торрент-трекер и скопируйте ссылку на файл .torrent, который хотите загрузить. В интерфейсе Basic перейдите на вкладку Links [Ссылки], вставьте URL в текстовое поле и нажмите кнопку Load Link [Загрузить ссылку] для загрузки торрента.

Загруженный торрент сканируется встроенным антивирусом *ClamAV* и помещается в каталог `/var/lib/mldonkey/incoming`.

С помощью *MlDonkey* вы сможете еще и раздавать свои торренты. Поскольку все уже настроено, перейдите на интерфейс Advanced [Продвинутый] в панели управления и перейдите на Help+ > Sysinfo [Справка > Информация о системе] для просмотра параметров. Каталоги по умолчанию приведены внизу страницы.

MlDonkey позволяет раздавать как отдельные файлы (например, `my-fav-distro.iso`), так и группы файлов в каталогах (напри-

мер, `the_top_distros/`). Чтобы раздавать индивидуальные файлы, поместите их в каталог `/var/lib/mldonkey/incoming/files`. Для раздачи каталогов поместите их в каталог `/var/lib/mldonkey/incoming/directories`.

Теперь надо создать файл .torrent — для этого нужен трекер, который оповестит о наличии этого файла других клиентов. По умолчанию *MlDonkey* отслеживает торренты в локальной сети с помощью IP-адреса сервера, на котором запущен.

Чтобы воспользоваться внешним трекером, придется изменить настройки *MlDonkey*. Самый простой способ это сделать — перейти к настройкам клиента [Client] в основном интерфейсе и взглянуть на поле `BT-default_tracker`. По умолчанию оно пусто — это означает, что используется локальный трекер.

Чтобы указать внешний трекер, введите его адрес в поле, например, `http://bit.ly/zJ4xF6`, это трекер для LinuxTracker.org. Теперь перейдите в интерфейс Advanced и введите команду `compute_torrent` в большое текстовое поле, чтобы указать файлы, которые нужно раздавать.

Например, `compute_torrent/srv/storage/incoming/files/my-fav-distro.iso` сгенерирует файл .torrent в каталоге `/var/lib/mldonkey/torrents/seeded`. Теперь перейдите в Transfers > Uploads [Передача > Загрузки], чтобы убедиться, что торренты благополучно раздаются.

Вот и все. Теперь можно оставить загруженные файлы и зарабатывать очки от своих одноранговых партнеров-пиров [peer] за раздачу торрентов. **LXF**

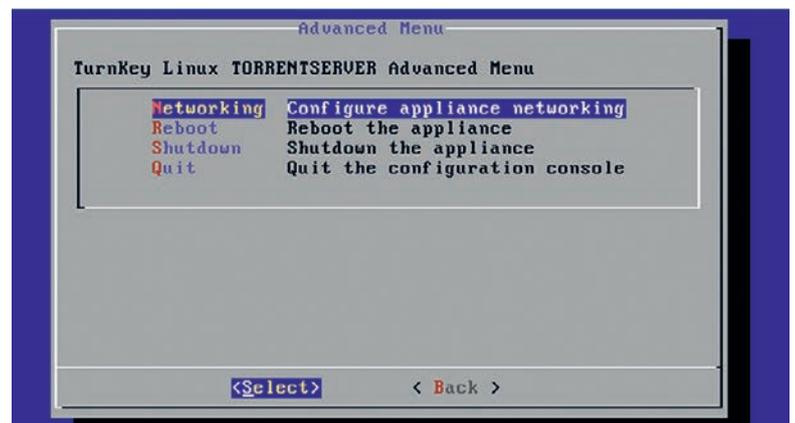


В Turnkey Linux в общей сложности 606 загружаемых модулей суммарным объемом 120 ГБ.

➤ Для повторной настройки сети, перезагрузки или выключения модуля воспользуйтесь меню Advanced.



➤ Иконка Basic переводит нас в интерфейс P2P-GUI, который гораздо проще интерфейса *MlDonkey* по умолчанию.



OpenCL: Стандарт

Михаил Остапкевич и Евгений Балдин отыскали способ призвать несколько разнотипных вычислителей под знамена одной программы.



Наш эксперт

Михаил

Остапкевич

Романтик, очарованный компьютерами и создаваемыми в них идеальными мирами; верит, что сложнейшие новые технологии могут и должны служить во благо человечеству.

Вы планируете загрузить нагло простаивающие процессоры, но еще не определились с любимой архитектурой? Вы не против загрузить видеокарту, причем не обязательно видеокарту фирмы N, а в перспективе перенести программу на энтерпрайзный Intel Xeon Phi и инди-процессор Eriphany, или прошить ее код в программируемую логическую интегральную схему — ПЛИС? Тогда OpenCL — ваш выбор, а кроме того, светлое будущее параллельного программирования!

Откуда и зачем все это?

OpenCL (Open Computing Language) — это стандарт, предложенный Apple в 2008 году. Как и MPI, он позволяет создавать платформенно-независимые параллельные программы. Но если MPI ориентирован на мультикомпьютеры, то OpenCL рассчитан на мультипроцессоры (например, SMP-системы или системы с многоядерными процессорами), GPU и всевозможные ускорители. Идея Apple заключалась в том, чтобы разогнать свой графический интерфейс с помощью простаивающих специализированных мощностей во время неигровой деятельности. Чуть погодя к Apple решила примкнуть фирма AMD, у которой на тот момент уже был свой собственный велосипед Close To Metal, который она убила в пользу открытого стандарта. Теперь OpenCL официально поддерживают фактически все основные производители вычислительной техники, включая IBM, Intel, AMD и NVIDIA. Разработкой OpenCL занимается некоммерческая независимая организация Khronos Group (<http://www.khronos.org/>).

Есть несколько реализаций OpenCL для различных вычислителей: CPU (IA-32, AMD64, ARM, STL Cell, Intel Xeon Phi), GPU (NVIDIA, AMD, S3 Imagination Technologies PowerVR) и даже ПЛИС (Xilinx, Altera). Народная суперкомпьютерная архитектура Eriphany от Adapteva также имеет SDK OpenCL. В перспективе появятся реализации для процессоров ZiiLabs ZMS-40 и цифровых сигнальных процессоров TI. OpenCL позволяет работать как с параллелизмом по данным, когда одна и та же операция одновременно применяется к разным комплектам данных, так и с параллелизмом на уровне задач, когда независимые потоки команд работают одновременно и выполняют разные функции.

В мире современных вычислительных систем есть весьма разнообразный и неоднородный набор вычислителей. Во-первых, есть мощные центральные процессоры, способные выполнять одновременно небольшое число потоков команд (нитей, threads). Также легко доступны графические процессоры, которые можно использовать для счета. Счет на число нитей в современных графических процессорах идет уже на тысячи, хоть они и не такие «толстые», как нити центрального процессора. Есть и другие разнообразные специализированные вычислительные ускорители (акселераторы) — например, ПЛИС'ы (FPGA) или сигнальные процессоры (DSP).

Даже в совершенно обычном компьютере уже довольно давно сожительствуют несколько разнотипных вычислителей — как минимум один процессор и графическая карта. А если посмотреть на самые высокопроизводительные кластеры, то размещение в их узлах нескольких многоядерных централь-

ных процессоров и нескольких GPU стало теперь уже практически стандартным. Но на этом эволюция не закончилась. Следующий шаг на пути ускорения видится в добавлении в вычислительный узел ПЛИС'ов, логика работы которых может многократно перепрограммироваться, а достижимая производительность вычислений в ряде полезных случаев на порядок выше даже чем у высокопроизводительных GPU.

Ранее в рамках данной серии было рассказано о технологии CUDA. Эта технология позволяет эффективно использовать для параллельного счета широкий спектр устройств GPU от NVIDIA прямо здесь и сейчас. Но как быть, если у нас есть графический процессор от другого производителя? Или что делать, если мы хотим, чтобы наша программа без переделки могла считать не только на GPU, но и на ядрах центрального процессора?

Как-то совершенно не хочется переписывать программу заново каждый раз, когда объявляется новый вид вычислителей. OpenCL в какой-то мере позволяет избежать этого. Цель создания OpenCL — дать единую технологию построения программ на разнородных вычислителях. Написав только одну программу, можно запустить ее фактически на всех параллельных устройствах. А если сейчас этого сделать нельзя — то это не надолго. Более того, OpenCL позволит запустить программу, написанную сегодня, на вычислителе, который создадут в будущем, даже без необходимости перекомпилировать ее заново! Следует, однако, осознать, что хоть правильно «твикнутая» программа под соответствующую архитектуру не уступает по производительности той же CUDA, но при переносе на принципиально другой вычислитель с другой производительностью безусловно могут возникнуть серьезные проблемы.

Словарик

В OpenCL аппаратное обеспечение подразделяется на две части: одна «хост-машина» и одно или несколько устройств OpenCL — примерно так, как это показано на рисунке. Хост-машина — это обычный компьютер, на центральном процессоре которого выполняется программа, координирующая ход вычислений. Устройство OpenCL — это устройство, используемое для выполнения вычислений. В качестве устройства OpenCL могут, например, использоваться GPU или сам процессор хост-машины.

Устройство OpenCL состоит из вычислительных модулей [compute unit], каждый из которых может выполнять свой поток команд. Вычислительный модуль содержит обрабатывающие элементы (ОЭ), каждый из которых одновременно с остальными обрабатывающим элементом обрабатывает свой комплект данных.

Если в качестве OpenCL-устройства используется GPU, то в нем может быть 8–16 вычислительных модулей, каждый из которых содержит до нескольких десятков обрабатывающих элементов. Если в качестве OpenCL-устройства используется CPU, то число вычислительных модулей в нем совпадает с числом ядер, или в два раза больше него, если в ядре поддерживается технология HyperThreading, а в каждом таком вычислительном модуле только один обрабатывающий элемент.



Наш эксперт

Евгений Балдин

Физик, который действительно знает, что такое нехватка вычислительных ресурсов.

для будущего

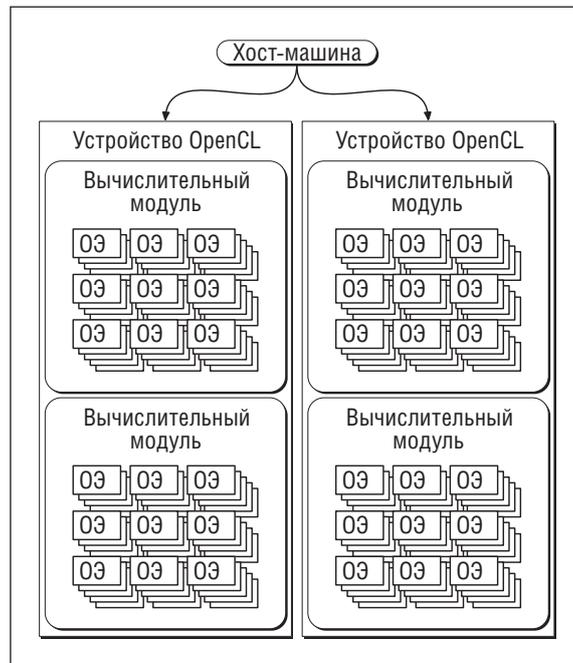
OpenCL-приложение состоит из двух частей: хост-программы и счетной программы. Хост-программа играет центральную роль. Она создает очереди для выполнения вычислений счетной программой, инициирует перемещения данных между хост-машиной и устройствами OpenCL, а также запускает на исполнение функции счетной программы. Счетная программа играет подчиненную роль. Она состоит из набора функций, производящих вычисления. Те ее функции, которые можно вызывать из хост-программы, называются ядрами. Инициатором запуска ядра на исполнение обычно выступает хост-программа. При этом запрос на его запуск размещается в очереди. Типичный порядок исполнения запросов в очереди — в порядке поступления запросов в нее, хотя OpenCL позволяет отойти от этого естественного правила. Запуск ядра заключается в создании большого числа процессов для его исполнения. Все эти процессы будут выполнять одну и ту же функцию, которая запрограммирована в ядре, но каждый из них будет обрабатывать свой фрагмент данных. Русскоязычный термин для этих процессов еще не устоялся, поэтому используем термин «рабочий элемент» (от англ. Work-item).

Определить, какой фрагмент данных нужно обрабатывать в данном рабочем элементе, можно по его уникальному идентификатору. В зависимости от характера обрабатываемых данных в качестве идентификатора можно использовать один, два или три целочисленных индекса.

Рабочий элемент может исполняться на одном или нескольких обрабатывающих элементах. Рабочие элементы, исполняемые на одном вычислительном модуле, образуют рабочую группу [work-group]. Если число обрабатывающих элементов больше числа рабочих элементов, то возможно их одновременное исполнение. Однако гораздо чаще возникает ситуация, когда рабочих элементов больше, чем обрабатывающих. Тогда исполнить сразу все рабочие элементы невозможно, и организуется их последовательное исполнение на обрабатывающих элементах.

Реальна ситуация, когда в одном рабочем элементе требуются данные, вычисляемые в другом, который, может быть, еще не начинал исполняться. В этом случае для корректной обработки требуется использование синхронизации, при которой рабочий элемент приостанавливается до того момента, когда требуемые ему данные будут посчитаны. Расстановка точек синхронизации — это просто неисчерпаемый источник разложенных на дороге разного рода «граблей». Без хорошего алгоритма вычислений легко загубить производительность параллельной программы, поэтому кроме программистских навыков, специалистам по параллельным вычислениям необходима фундаментальная математическая подготовка.

Разные реализации OpenCL могут сосуществовать на одной хост-машине. Более того, их можно одновременно использовать в одной хост-программе. Формально они представляются как отдельные «OpenCL-платформы», каждая из которых содержит набор поддерживаемых конкретно ими OpenCL-устройств. Одно и то же устройство может присутствовать в более чем одной платформе. Например, центральный процессор Intel Core2 Quad Q8300 может быть перечислен как в реализации Intel OpenCL, так



» Модель OpenCL-платформы.

и в AMD OpenCL. При этом на хост-машине может быть установлен пакет CUDA, который, кроме всего прочего, также содержит реализацию OpenCL. В платформе, соответствующей ему, будет перечислено одно устройство GPU, например, NVIDIA GTS250. Со всем абстрагироваться, естественно, не удастся, так как необходимо хотя бы примерно представлять, где программа будет считаться, но стандартизация правильная.

Память OpenCL-устройства отделена от памяти хост-машины. Перед началом вычислений необходимо скопировать исходные данные из основной памяти хост-машины в глобальную память OpenCL-устройства, а после завершения вычислений переместить результаты обратно. Глобальная память доступна всем рабочим элементам счетной программы. Кроме нее, OpenCL-устройство имеет константную, локальную и собственную память [private memory]. Константная память доступна всем рабочим элементам по чтению. Записывать в нее можно из хост-программы. Локальная память может использоваться только рабочими элементами одной рабочей группы. Собственная память доступна только одному рабочему элементу. Такое разделение на типы памяти позволяет более полно использовать особенности графических ускорителей. В частности, в случае GPU локальная память проецируется на разделяемую, а собственная — на регистровую; обе они существенно быстрее глобальной. При использовании же центрального процессора в качестве OpenCL-устройства глобальная, локальная и собственная памяти проецируются на основную память, и скорость их работы соответственно одинакова.

В стандарте OpenCL описан функциональный интерфейс для хост-машины, язык «OpenCL C» для реализации счетной программы и функциональный интерфейс для OpenCL-устройства.

»

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

Интерфейс для хост-машины — это тот набор функций, который можно вызывать из хост-программы, написанной, например, на C. Эти функции позволяют получать список OpenCL-устройств, определять их параметры (clGetPlatformIDs, clGetPlatformInfo), компилировать счетную программу (clCreateProgramWithSource), выделять и освобождать память на OpenCL-устройстве (clCreateBuffer, clReleaseMemObject), пересылать данные (clEnqueueReadBuffer, clEnqueueWriteBuffer), запускать вычисления (clEnqueueNDRangeKernel) и осуществлять синхронизацию на OpenCL-устройствах.

Язык OpenCL C основан на стандартном C, однако в нем нет стандартной библиотеки функций ANSI C и ее заголовочных файлов, указателей на функции, массивов переменной длины и битовых полей, а также в нем запрещена рекурсия. Но, с другой стороны, в него добавлены рабочие элементы, рабочие группы, векторные типы, синхронизация и квалификаторы адресного пространства, которые определяют, в глобальной, локальной или собственной памяти располагаются данные.

Сборка программы

OpenCL C включен в библиотеку для хост-программы и вызывается через интерфейс для хост-машины. Компиляция счетной программы производится в процессе работы хост-программы. Это позволяет настраиваться на использование того или иного устройства OpenCL в процессе выполнения программы. Также это даст возможность исполнять OpenCL-программу без перекомпиляции хост-программы даже на тех устройствах, которые еще не разработаны. В отличие от CUDA или MPI, компиляция OpenCL-программы не требует запуска препроцессора, хотя и не исключает его:

```
gcc helloworld.c -o helloworld -lOpenCL
```

Возможно, при компиляции придется указать местоположение заголовочных файлов и библиотеки *libOpenCL*. Предполагается, что соответствующие драйвера и SDK для имеющихся вычислительных устройств уже установлены. Если вы не в курсе, о чем тут говорится, то для окукивания GPU от Nvidia имеет смысл заглянуть по адресу <https://developer.nvidia.com/opencl>, для Intel есть соответствующий ресурс <http://software.intel.com/en-us/vcsources/tools/opencl>, а в случае видеокарты от AMD можно порыться на сайте <http://developer.amd.com>.

Интерфейс для вычислительного устройства позволяет работать с рабочими элементами и векторными данными и производить синхронизацию потоков, и содержит большое число математических и геометрических функций, функций печати, отладки, асинхронного копирования между разными видами памяти.

Пример хост-программы

В типичной хост-программе можно выделить следующие этапы работы:

- 1 Определение доступных устройств OpenCL и их характеристик.
- 2 Формирование контекста, который включает выбранные для использования устройства и настройки.
- 3 Создание очереди команд.
- 4 Создание двоичного исполняемого кода счетной программы по ее исходному тексту.
- 5 Декларация ядра в двоичном коде счетной программы.
- 6 Резервирование памяти в OpenCL-устройствах и копирование туда данных для счетной программы.
- 7 Инициализация запуска ядер на счет.
- 8 Копирование результатов вычисления из памяти OpenCL-устройства в основную память хост-машины и освобождение блоков памяти, зарезервированных ранее на OpenCL-устройствах.

9 Освобождение ресурсов и удаление созданные ранее объектов в хост-программе по работе с OpenCL.

Рассмотрим пример простейшей хост-программы поэлементного сложения двух векторов.

Для начала нужно подключить OpenCL-интерфейс для хост-программы (возможно, для вашей реализации OpenCL путь до заголовочного файла будет немного другой):

```
#include <CL/opencl.h>
#include <stdio.h>
#include <stdlib.h>
```

```
Для удобства определяем константы:
#define TEST1_OPENCL1_BUF1_SIZE1 4096
#define SIZE 5 // Задание размера векторов
```

Добавляем исходник счетной программы, содержащей ядро vadd для сложения двух чисел:

```
const char *vaddsrc =
    "__kernel void vadd(
    \" __global float *d_A,\"
    \" __global float *d_B,\"
    \" __global float *d_C\"
    ){
    \" unsigned int n;\"
    \"\"
    // Получение уникального идентификатора рабочего элемента
    \" n = get_global_id(0);\"
    \" d_C[n] = d_A[n] + d_B[n];\"
    }\";
};
```

Здесь d_A и d_B — указатели на складываемые векторы в глобальной памяти, d_A — указатель на результирующий вектор.

Определяем векторы в основной памяти хост-компьютера:

```
float hostdata_A[SIZE] = {10., 20., 30., 40., 50.};
float hostdata_B[SIZE] = {1., 2., 3., 4., 5.};
float hostdata_C[SIZE];
char clcompileflags[TEST1_OPENCL1_BUF1_SIZE1];
char buf[TEST1_OPENCL1_BUF1_SIZE1];
// Одномерное пространство уникальных идентификаторов рабочих элементов
size_t sizeWork[1] = {SIZE};
```

Начинаем программу с определения вспомогательных переменных:

```
int main(){
// Код ошибки, который возвращается OpenCL-функциями
cl_int clerr;
// Количество доступных OpenCL-платформ
cl_uint qty_platforms = 0;
// Список идентификаторов OpenCL-платформ
cl_platform_id* platforms;
cl_uint ui;
// Количество доступных OpenCL-устройств для каждой платформы
cl_uint *qty_devices;
cl_device_id **devices;
cl_uint i;
// Дескриптор контекста
cl_context context1;
size_t parmsz;
// Дескриптор очереди команд
cl_command_queue queue1;
cl_program program1;
// Дескриптор объекта ядра
cl_kernel kernel1;
```

» Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

Определяем указатели на векторы A и B с исходными данными и вектор C с результирующими данными в глобальной памяти:

```
cl_mem remotedata_A;
cl_mem remotedata_B;
cl_mem remotedata_C;
```

Этап 1 Определение доступных OpenCL-устройств.

```
// Определение числа OpenCL-платформ
clerr = clGetPlatformIDs(0, NULL, &qty_platforms);
// Обработка ошибок
if(clerr != CL_SUCCESS){
    fprintf(stderr, "Ошибка, код = %d.\n", clerr);
    // Далее аналогичные блоки опущены
    return 1;
}
// Выделение памяти для хранения информации
// о конфигурации и параметров устройств
platforms = (cl_platform_id*)
    malloc(sizeof(cl_platform_id)*qty_platforms);
devices = (cl_device_id**)
    malloc(sizeof(cl_device_id**)*qty_platforms);
qty_devices = (cl_uint*)malloc(sizeof(cl_uint)*qty_platforms);
// Обработка ошибок выделения памяти здесь и далее
// опущена
// Получение списка идентификаторов платформ
clerr = clGetPlatformIDs(qty_platforms, platforms, NULL);
for (ui=0; ui < qty_platforms; ui++){
    // Получение количества имеющихся OpenCL-устройств
    // для каждой платформы
    clerr = clGetDeviceIDs(platforms[ui], CL_DEVICE_TYPE_ALL,
        0, NULL, &qty_devices[ui]);
    // Получение списка идентификаторов OpenCL-устройств
    if(qty_devices[ui]){
        devices[ui] = (cl_device_id*)
            malloc(qty_devices[ui] * sizeof(cl_device_id));
        clerr = clGetDeviceIDs(platforms[ui], CL_DEVICE_TYPE_
            ALL, qty_devices[ui], devices[ui], NULL);
    }
}
```

Этап 2 Формирование контекста. Создается контекст, в который включены OpenCL-устройства платформы 0 (упрощение):

```
clerr = CL_SUCCESS;
context1 = clCreateContext(0, qty_devices[0], devices[0], NULL,
    NULL, &clerr);
```

Этап 3 Создание очереди команд. Очередь создается для нулевого устройства нулевой платформы:

```
queue1 = clCreateCommandQueue(context1, devices[0][0], 0,
    &clerr);
```

Этап 4 Компиляция счетной программы.

```
program1 = clCreateProgramWithSource(context1, 1, &vaddsrc,
    NULL, &clerr);
snprintf(clcompileflags, TEST1_OPENCL1_BUF1_SIZE1,
    "-cl-mad-enable");
clerr = clBuildProgram(program1, 0, NULL, clcompileflags, NULL,
    NULL);
```

Этап 5 Декларация ядра в счетной программе.

```
kernel1 = clCreateKernel(program1, "vadd", &clerr);
```

Этап 6 Резервирование памяти в OpenCL-устройстве. При этом векторы A и B с исходными данными копируются из основной памяти хост-машины в память OpenCL-устройства.

```
remotedata_A = clCreateBuffer(context1, CL_MEM_READ_ONLY |
    CL_MEM_COPY_HOST_PTR, sizeof(int) * SIZE, hostdata_A, NULL);
remotedata_B = clCreateBuffer(context1, CL_MEM_READ_ONLY |
    CL_MEM_COPY_HOST_PTR, sizeof(int) * SIZE, hostdata_B, NULL);
remotedata_C = clCreateBuffer(context1, CL_MEM_WRITE_ONLY |
    CL_MEM_COPY_HOST_PTR, sizeof(int) * SIZE, hostdata_C, NULL);
```

Этап 7 Запуск ядра на вычисления.

```
clSetKernelArg(kernel1, 0, sizeof(cl_mem),
    (void*)&remotedata_A);
clSetKernelArg(kernel1, 1, sizeof(cl_mem),
    (void*)&remotedata_B);
clSetKernelArg(kernel1, 2, sizeof(cl_mem),
    (void*)&remotedata_C);
clerr = clEnqueueNDRangeKernel(queue1, kernel1, 1, NULL,
    sizeWork, NULL, 0, NULL, NULL);
```

Этап 8 Копирование результатов в основную память хост-машины и освобождение памяти в OpenCL-устройстве.

```
// Копирование результирующего вектора
clEnqueueReadBuffer(queue1, remotedata_C, CL_TRUE, 0, SIZE *
    sizeof(int), hostdata_C, 0, NULL, NULL);
// Освобождение памяти на OpenCL-устройстве
clReleaseMemObject(remotedata_A);
clReleaseMemObject(remotedata_B);
clReleaseMemObject(remotedata_C);
```

Этап 9 Освобождение ресурсов.

```
clReleaseKernel(kernel1);
clReleaseProgram(program1);
clReleaseCommandQueue(queue1);
clReleaseContext(context1);
free(qty_devices);
free(devices);
free(platforms);
```

А теперь наконец-то можно выдохнуть и распечатать результирующий вектор:

```
for(i = 0; i < SIZE; i++)
    printf("%f\n", hostdata_C[i]);
}
```

Несмотря на видимую сложность, рассмотренный пример максимально упрощен. В частности, мы производили счет на фиксированном нулевом устройстве нулевой платформы. В реальности мы предпочли бы более осознанно выбрать устройство для счета — например, возможно, захотели бы поискать сначала GPU от NVIDIA. Для этого в функции `clGetDeviceIDs` необходимо заменить `CL_DEVICE_TYPE_ALL` на `CL_DEVICE_TYPE_GPU`.

В более сложных реализациях счет может производиться на нескольких разнотипных OpenCL-устройствах. Типичная схема организации такого счета заключается в создании нескольких контекстов или нескольких очередей в одном контексте, каждая из которых ассоциирована с некоторым устройством.

Другое упрощение связано с тем, что текст счетной программы с ядром мы храним как константу в хост-программе. Счетных программ и их ядер может быть достаточно много, и в полноценной реализации имеет смысл хранить их в отдельных файлах и загружать при необходимости.

В заключение необходимо отметить, что использование OpenCL не исключает возможность применения других технологий построения параллельных программ в одной программе. Например, на кластерах с многопроцессорными узлами или узлами с GPU целесообразно строить гибридные программы на базе OpenCL и MPI. OpenCL в них будет отвечать за организацию параллельного счета внутри одного узла, а MPI — за параллельное исполнение на уровне кластера, между несколькими узлами. **LXF**

Обратная связь

Приглашаем высказаться потенциальных авторов статей по параллельным вычислениям – ценные предложения, критику и советы присылайте по электронной почте: ostap@ssd.sgcc.ru, E.M.Baldin@inp.nsk.su.

Vacula: Шанс для

Игорь Спирин устанавливает сервер системы резервного копирования.



Наш эксперт

Игорь Спирин — ведущий инженер ООО «Газинформсервис». Его увлечение GNU/Linux со временем переросло в интересную работу. В свободное время любит что-нибудь «свеженькое» поставить из ветки Debian unstable...



- » *Vacula* является кросс-платформенной сетевой клиент-серверной программой резервного копирования, использующей для передачи данных протокол TCP/IP.
- » Архитектура ПО имеет модульную структуру, в ней изначально заложена возможность масштабирования.
- » Проект *Vacula* существует достаточно давно — «детские болезни» изжиты.

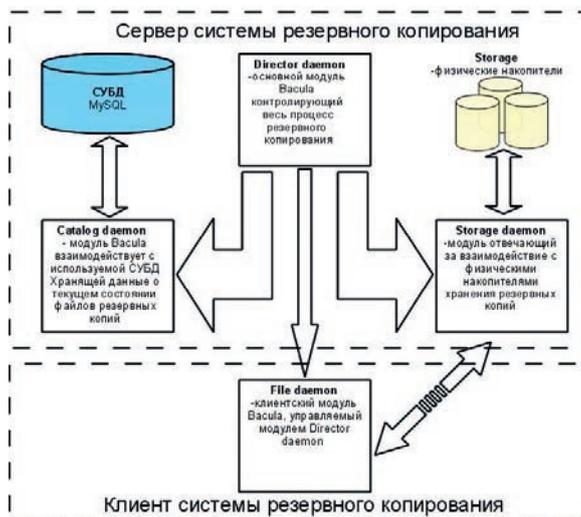
Предлагаемая структура системы резервного копирования на основе СПО *Vacula* приводится на рис. 1.

Ознакомимся с компонентами:

- » Система управления базы данных (СУБД) — необходимый компонент для работы *Vacula*.
- » *Catalog daemon* — задаваемые параметры работы находятся в секции *Catalog {...}* файла */etc/bacula/bacula-dir.conf*.
- » *Director daemon (DD)* — говоря музыкальным языком, это «дирижер», который руководит слаженной работой компонентов согласно настройкам в */etc/bacula/bacula-dir.conf*.
- » *Storage* — физические устройства хранения информации, как дисковые, так и ленточные. Их параметры и перечень прописываются в секции *Device {...}* файла */etc/bacula/bacula-sd.conf*.
- » *Storage daemon (SD)* — задаются в файле */etc/bacula/bacula-sd.conf*.
- » *File daemon (FD)* — взаимодействует с сервером с соответствии с файлом */bacula-fd.conf*.

Настроенная IT-инфраструктура предприятия надежно работает? У вас появилось свободное время? Есть недозагруженный Linux-сервер? Самое время создать систему резервного копирования. Возьмем за основу одного из представителей свободного программного обеспечения — *Vacula*.

Почему, собственно, *Vacula*, а не другое ПО? Вот аргументы в защиту *Vacula*:



```
Storage {
    # definition of myself
    Name = bserver1-sd
    SDPort = 9103 # Director's port
    WorkingDirectory = "/var/lib/bacula"
    Pid Directory = "/var/run/bacula"
    Maximum Concurrent Jobs = 40
    # SDAddress = 127.0.0.1
}

Director {
    Name = bserver1-dir
    Password = "Gu9nCa2YqkNl82PmD3tHEjK3TSjvmzdWl"
}

Director {
    Name = bserver1-mon
    Password = "3iUtRy8BafmEuLlmtZRLR0hlcPeunDwMf"
    Monitor = yes
}

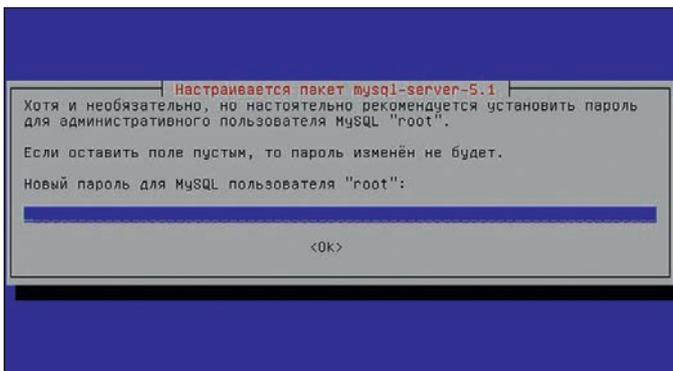
Device {
    # Уникальное имя используемого устройства
    Name = FileStorage
    # тип используемого устройства
    Media Type = File
    # путь к устройству хранения
    Archive Device = /mnt/tank1
    # yes - автоматического маркирования носителей
    LabelMedia = yes;
    # параметры доступа к данным на устройстве
    # для ленточных устройств no
    Random Access = Yes;
    # автоматическое монтирование
    AutomaticMount = yes;
    # возможность извлечения устройства хранения
    RemovableMedia = no;
    # no - открывать доступ к устройству только во время обращения
    AlwaysOpen = no;
}
```

» Рис. 2. Секции *Storage* и *Device* файла параметров *SD*.

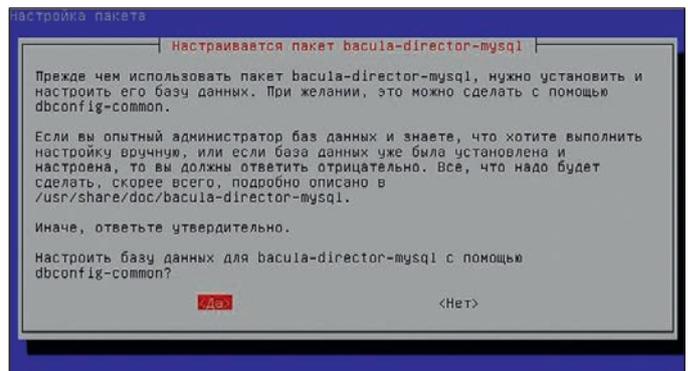
» Рис. 1. Структура системы резервного копирования.

СИСОДМИНО

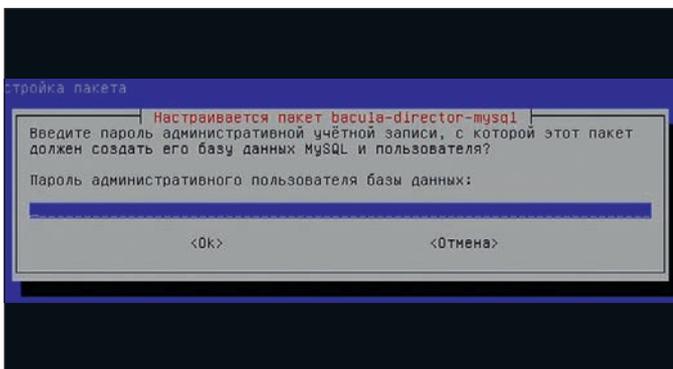
Шаг за шагом: Устанавливаем сервер Bacula



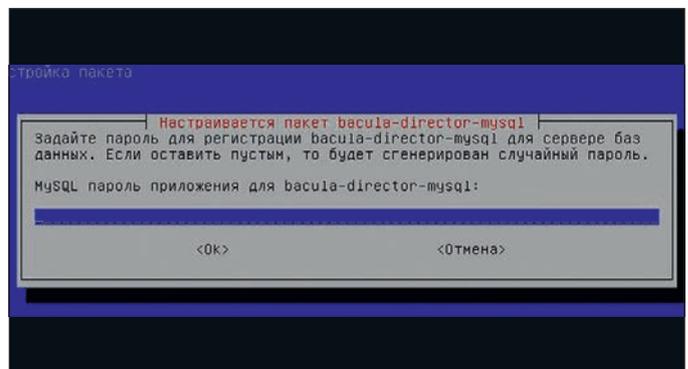
1 Установим пароль СУБД MySQL



2 Выберем способ настройки СУБД



3 Создадим экземпляр СУБД



4 Установим пароль DD для работы с СУБД

Бегло изучив архитектуру системы, приступим к главному: установке и базовой настройке компонентов *Bacula*. Устанавливать будем *Bacula* на сервер с ОС Debian (выпуск stable). От имени root выполним в команду

```
apt-get update && apt-get -y install bacula bacula-director-mysql bacula-sd-mysql bacula-fd bacula-doc
```

В процессе установки инсталлятор установит СУБД *MySQL*, если ранее она не была установлена, и попросит задать пароль администратора СУБД. Далее идет процесс настройки пакета *bacula-director-mysql*; указываем пароль для DD, используемый для входа в СУБД. Вот, в общем-то, и все — инсталляция завершена.

Проверим состояние компонентов ПО *Bacula*, выполнив команды **service bacula-director status** (состояние DD — “bacula-dir is running”); **service bacula-sd status** (состояние SD — “bacula-sd is running”); **service bacula-fd status** (состояние File daemon — “bacula-fd is running”).

Получив на выходе заветное “... is running”, приступим к настройке нашего сервера системы резервного копирования.

Настройку начнем с Storage daemon'a — файла **/etc/bacula/bacula-sd.conf**. Поправим вначале секцию Storage {...}, сделав параметр SDAddress неактивным подстановкой в начало строки символа #, а затем секцию Device {...} — задав значение

» Не хотите пропустить номер? Подпишитесь на [www.linuxformat.ru/subscribe/!](http://www.linuxformat.ru/subscribe/)

```

Director {
    # define myself
    Name = debian-dir
    DIRport = 9101 # where we listen for UA connections
    QueryFile = "/etc/bacula/scripts/query.sql"
    WorkingDirectory = "/var/lib/bacula"
    PidDirectory = "/var/run/bacula"
    # Максимальное кол-во параллельных заданий
    Maximum Concurrent Jobs = 40
    Password = "Cv70F6pfltpBopT4vQ0nigDrR0v3L" # Console password
    Messages = Daemon
    # DirAddress = 127.0.0.1
}

# Профиль задания используемый по умолчанию
JobDefs {
    Name = "DefaultJob"
    Type = Backup
    Level = Incremental
    Client = debian-fd
    # Перечень сохраняемых файлов
    FileSet = "Full Set"
    # План выполнения
    Schedule = "WeeklyCycle"
    Storage = File
    Messages = Standard
    Pool = File
    # Приоритет задания (максимальный -1, минимальный -1000)
    Priority = 10
    # Файл хранящий данные о том где найти резервную копию
    # восстанавливаемых данных
    Write Bootstrap = "/var/lib/bacula/%c.bsr"
}

# Definition of file storage device
Storage {
    Name = File # Do not use "localhost" here
    Address = 10.10.8.251 # N.B. Use a fully qualified name here
    SDPort = 9103
    Password = "_iBuLFqI1jxgE0Lc0Hx9Z6AatgJweeRLL"
    Device = FileStorage
    Media Type = File
}

# Default pool definition
Pool {
    Name = Default
    Pool Type = Backup
    Recycle = yes # Bacula can automatically recycle Volumes
    AutoPrune = yes # Prune expired volumes
    Volume Retention = 365 days # one year
    Label Format = "volExample_Default-${Year}_${Month}_${Day}"
}
    
```

Рис. 3. Основной фрагмент файла настроек DD.

параметра Archive Device, указав путь к файлу устройства (/dev/...) либо каталог, в котором, собственно, и будут размещаться наши резервные копии. На рис. 2 показан пример. После внесения правок в файл параметров SD сохраняем изменения и проверяем их корректность командой

```
bacula-sd -c /etc/bacula/bacula-sd.conf -t
```

Если в результате вышеуказанной команды на экране не появились ошибки, значит, базовая настройка SD благополучно

Рис. 5. Вид консоли BAT, страница Jobs).

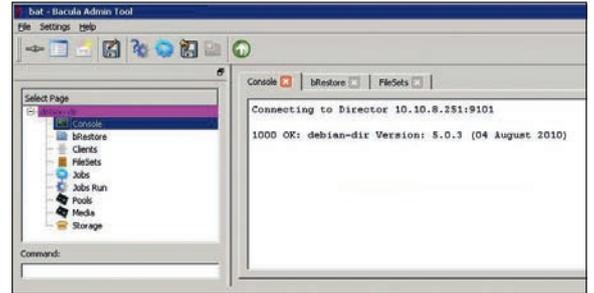
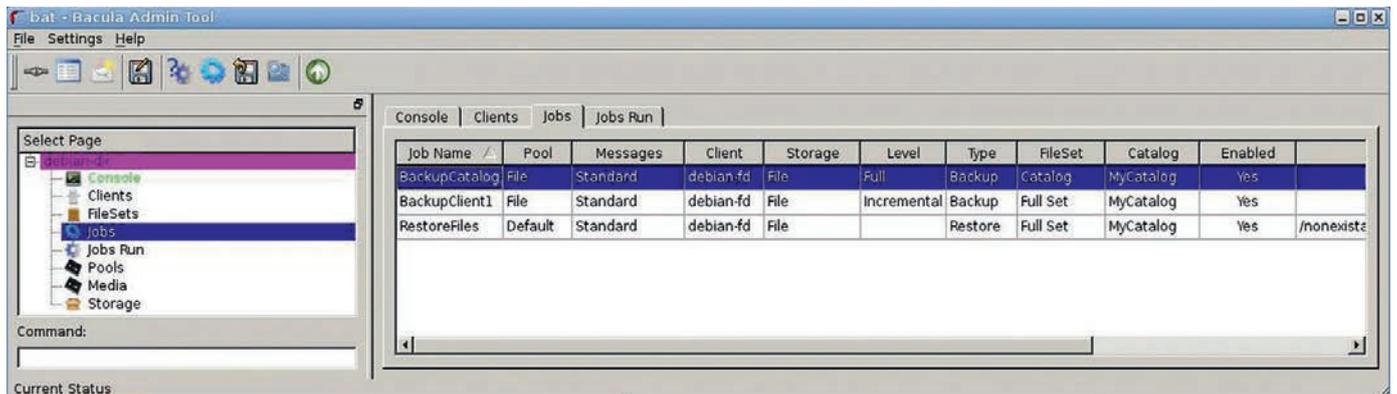


Рис. 4. Вид консоли управления Bacula Administration Tool.

завершена, и самое время настроить другой компонент Bacula — DD. Так что займемся правкой конфигурационного файла /etc/bacula/bacula-dir.conf нашего сервера резервного копирования.

Начнем с секции Director {...}, где присвоим новое, явно большее значение Maximum Concurrent Jobs; в нашем случае для всех компонентов выставим значение, равное 40. Далее параметр DirAddress сделаем неактивным, поставив в начале строки символ #, и тем самым разрешим взаимодействовать DD с инструментами управления bconsole и BAT (Bacula Administration Tool), подключаемыми как локально, так и удаленно.

Да, и еще, не позабудем в каждой секции Pool {...} добавлять переменную Label Format = «LabelName-\${Year}_\${Month}_\${Day}».

После внесенных правок в файл параметров DD, сохраним изменения и проверим их корректность командой

```
bacula-dir -c /etc/bacula/bacula-dir.conf -t
```

Далее выполним перезапуск компонентов Bacula командой `service bacula-sd restart` и `service bacula-director restart`

Основной фрагмент файла настроек DD приведен на рис. 3. Цветом выделены настраиваемые параметры.

Далее следует установить и настроить инструмент управления Bacula — Bacula Administration Tool (BAT).

Установка BAT

Установку BAT выполняется на том же компьютере, с которого мы в дальнейшем собираемся управлять системой резервного копирования.

Установку BAT под Linux делает команда `apt-get install bacula-console-qt`; чтобы установить BAT под ОС семейства Windows, сперва скачаем последнюю версию инсталлятора по ссылке http://sourceforge.net/projects/bacula/files/Win32_64/ и запустим ее. По ходу установки BAT инсталлятор запросит параметры DD, находящиеся в секции Director {...} файла /etc/bacula/bacula-dir.conf. Установка под обеими ОС проста и дополнительных пояснений обычно не требует.

» Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

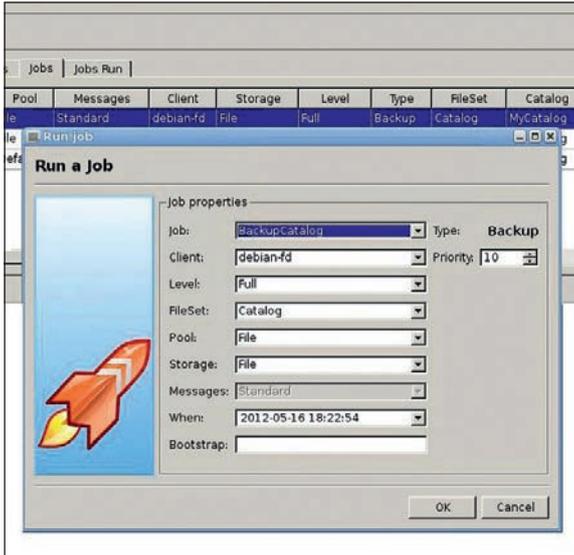


Рис. 6. Вид консоли BAT, подпункт Run Job.

После инсталляции следует настроить BAT — под ОС семейства Linux правим файл `/etc/bacula/bat.conf`. Покончив с настройкой, запустим BAT; и если в процессе настройки были заданы верные значения параметров, то на экране монитора появится окно консоли управления *Bacula*, с сообщением об удачном подключении к DD. Вид консоли BAT показан на рис. 4.

Далее, используя BAT, проверим выполнение резервного копирования файлов СУБД нашего сервера резервного копирования, с помощью установленного ранее пакета *bacula-fd*.

В открытой консоли управления BAT выберите и откройте страницу Jobs [Задания], в поле открывшейся страницы выберите задание BackupCatalog, правой клавишей мыши выберите подпункт Run Job [Запустить задание] и в появившемся окне выберите нужные параметры (по умолчанию) и нажать ОК. Для контроля выполнения ранее выбранного задания перейдем на страницу Jobs Run [Запущенные задания] и посмотрим состояние задания BackupCatalog. Описанные действия пояснены на рис. 5–7.

Если действия по установке и базовой настройке выполнялись правильно, то результат не заставит себя ждать — “Completed successfully [Успешное завершение]”.

Успешно создав резервную копию сервера системы резервного копирования, приступим к установке и настройке дополнительного инструмента мониторинга выполнения заданий резервного копирования — *Bacula demon status monitor* или просто *traymonitor*.

«Установка — это еще не все: надо будет настроить параметры.»

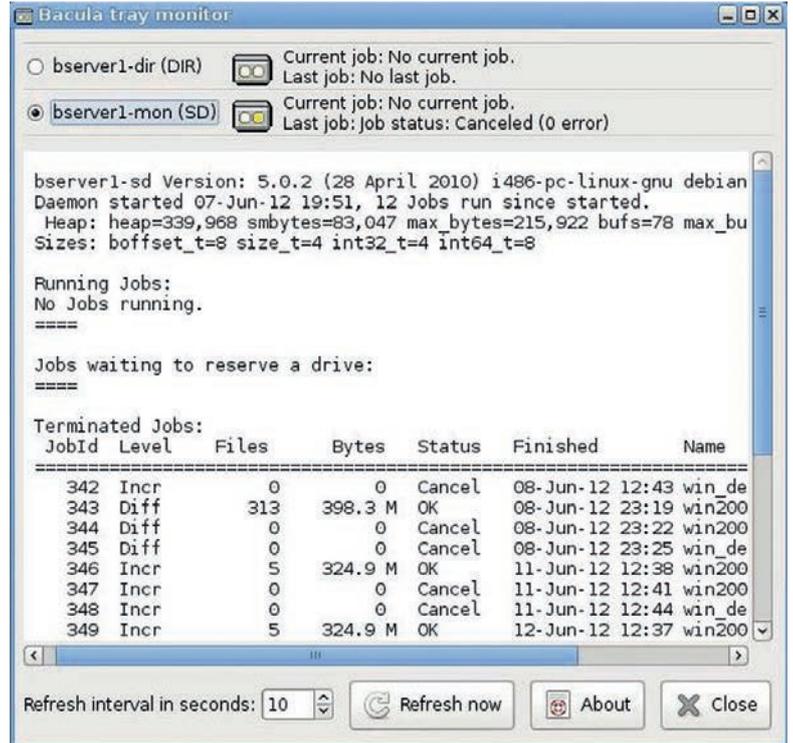


Рис. 8. Вид развернутой консоли мониторинга traymonitor.

Установка *traymonitor*:

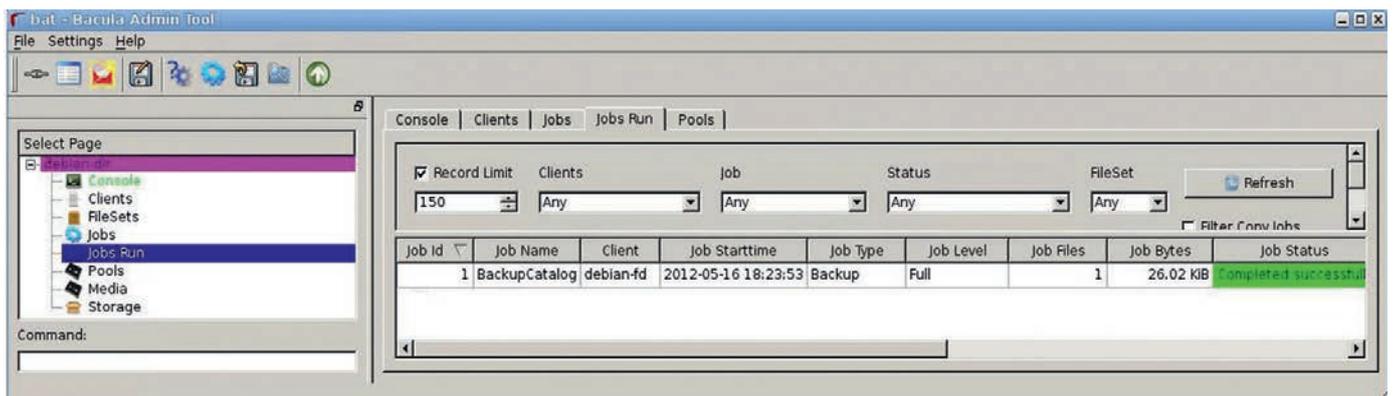
» Под ОС семейства Linux устанавливаем пакет *bacula-traymonitor* командой `apt-get install bacula-traymonitor`;

» Под ОС семейства Windows установка выполняется аналогично установке BAT, только в меню инсталлятора отмечается компонент *traymonitor*.

Но установка — это еще не все; надо будет настроить параметры подключения к DD и SD. Параметры подключений к DD берем из секции `Console {...}` файла `/etc/bacula/bacula-dir.conf`, а для подключения к SD из секции `Director {...}` с активным параметром `Monitor` файла `/etc/bacula/bacula-sd.conf`. После выполнения вышеуказанных действий запускаем *traymonitor* и, в случае успеха, наблюдаем значок дискеты, повисший в системном лотке на рабочем столе, и при необходимости разворачиваем окно консоли. Вид развернутой консоли *traymonitor*'а показан на рис. 8.

Продолжение следует... далее нам предстоит установка FD на ОС Windows, Solaris и AIX и создание правил резервного копирования, а также выработка практических рекомендаций по разбивке пулов, томов. LXF

Рис. 7. Вид консоли BAT, страница Jobs Run.



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 Пропажу командной строки
- 2 Телевидение на Raspberry Pi
- 3 Брандмауэр Netgear
- 4 Беспроводные сети
- 5 Нераспознаваемый формат файла
- 6 Запуск Mint на iBook

1 А где мой терминал?..

В Я прекрасно работал в Ubuntu на своем стареньком ноутбуке, а друг дал мне свой Sony Desktop Vaio с Windows Vista, который тормозил из-за вируса. Я установил на него Ubuntu 12.10 с вашего DVD, и в ней я и набираю этот текст. Проблем нет, за исключением того, что я не могу найти чертов терминал. Меня не беспокоит, что они так продвигают Amazon. Мне не нравятся иконки, но уж так заведено. Но нельзя же пользоваться компьютером без командной строки!

Джон Вагнер [John Wagner]

О А, радости Unity... Там больше нет меню со всеми программами, где можно что-нибудь выбрать. Вместо этого щелкните по символу «Домой» в Dash, в верхнем левом углу экрана, затем щелкните в строке поиска и начинайте набирать «terminal». Вскоре появится иконка. При работающем терминале Вы увидите иконку для него в строке запуска (Launcher). Щелкните по ней правой кнопкой и выберите «Привязать к строке запуска [Lock to Launcher]», чтобы закрепить ее на постоянном месте. Затем вы сможете ее перетащить в более удобное место в строке запуска.

Как активному пользователю терминала, Вам можно предложить несколько вариантов. Первый — запускать терминал автоматически при загрузке рабочего стола. Вернитесь к иконке «Домой» в Dash и найдите Автозагрузку [Startup Applications]. Нажмите Добавить [New] и укажите gnome-terminal в качестве запускаемой команды. Можно указать дополнительные параметры — например, команда

```
gnome-terminal --geometry 80x25
```

откроет терминал размером 80×25. Для просмотра всех доступных опций введите команду

```
gnome-terminal --help
```

Другой вариант — выпадающий терминал в стиле Quake. В KDE я пользуюсь *Yakuake*, которая открывает терминал *Konsole* при нажатии горячей клавиши (KDE также запоминает открытые мной терминалы и восстанавливает их при следующем

входе в систему, но это другая история); аналогичная программа есть и для Gnome — *Guake*. Попробуйте ее: свободный терминал под рукой по-настоящему повышает производительность.

2 ТелеPi

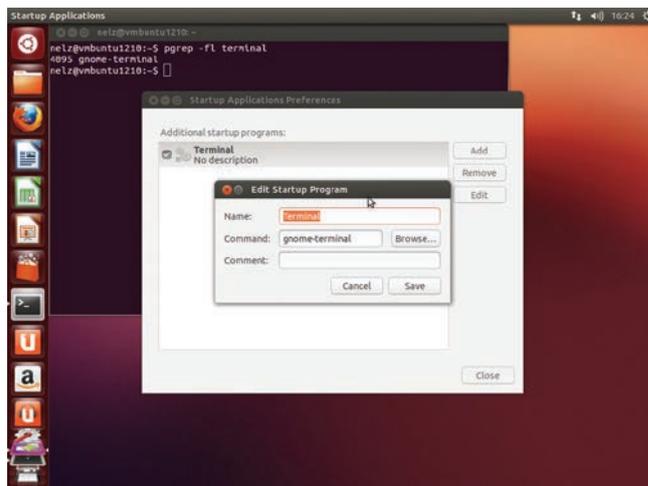
В Я не один день пытался заставить Raspberry Pi (версия 2, на данный момент с Raspbmc) работать со своим ТВ-тюнером. Я приобрел KW-UB499-2T T09, потому что если верить http://linuxtv.org/wiki/index.php/DVB-T_USB_Devices, он совместим с RPi со свежим ядром, к тому же я нашел его на eBay по довольно доступной цене. Однако я не могу заставить его работать: когда я открываю web-интерфейс, адаптер просто не появляется в списке адаптеров *tvheadend*. Я подробно читал руководства и форумы, большинство которых посвящено системам x86 или x64. Я сделал все что мог, потому что не разбираюсь в модулях ядра, драйверах и тому подобном (до сих пор мне везло, и все «просто работало»). Не могли бы вы помочь заставить его работать, так как сейчас я не могу позволить себе купить еще один адаптер — и я почти отчаялся, пока статья Грэма в LXF169 не заставила меня предпринять еще одну попытку.

Марк Скиннер [Mark Skinner]

О Да, это устройство действительно поддерживается текущими ядрами Raspbmc, но чтобы заставить его работать, нужны не только модули ядра. Как и для многих других подобных устройств, нужен еще файл прошивки. Это небольшой фрагмент кода, который должен находиться в микросхемах устройства, но изготовители предпочли включить его в драйвер Windows.

Это упрощает обновления, а также позволяет исправлять ошибки, не меняя устройство (что позволяет разработчикам раньше выпустить устройство, потратив меньше времени на тестирование). Чтобы устройство работало в Linux, Вам нужна эта прошивка в файле в каталоге `/lib/firmware`. Если Вы посмотрите на вывод *dmesg* командой

```
sudo dmesg | grep -C 5 -i dvb
```



Unity хоть и прячет программы, но их легко запустить автоматически.

то скорее всего увидите сообщения об отсутствующей прошивке. Установить ее можно двумя способами.

Если у Вас установлены исходники ядра, выполните следующие команды для автоматической загрузки, распаковки и установки прошивки:

```
cd /usr/src/linux/Documentation/dvb
sudo ./get_dvb_firmware it9135
```

Они загрузят и установят прошивку для чипсета it9135, который используется в Вашем устройстве.

В противном случае, если исходники ядра не установлены, сделать это можно вручную, запустив следующие команды в терминале или в сеансе SSH:

```
wget http://www.ite.com.tw/uploads/firmware/v3.6.0.0/dvb-usb-it9135.zip
unzip dvb-usb-it9135.zip
sudo dd if=dvb-usb-it9135.fw ibs=1 skip=64 count=8128 of=/lib/firmware/dvb-usb-it9135-01.fw
sudo dd if=dvb-usb-it9135.fw ibs=1 skip=12866 count=5817 of=/lib/firmware/dvbusb-it9135-02.fw
```

Команда **unzip** в Raspbmc по умолчанию не установлена. Ее можно установить либо командой

```
sudo apt-get install unzip
```

либо загрузив и распаковав файл на настольном компьютере и скопировав содержимое архива, файл **dvb-usb-it9135.fw**, на Raspberry Pi, после чего выполнить две команды **dd**. Теперь подключите USB-тюнер, и он должен появиться в каталоге `/dev/dvb` и в *tvheadend*.

3 Двойной брандмауэр

В Я пользуюсь Netgear DG834G v5, и несколько лет полагался на него в качестве брандмауэра. Он настроен на блокировку всего

входящего трафика и разрешение всего исходящего трафика. В Linux брандмауэры я не настраивал. Стоило ли полагаться на брандмауэр Netgear? Была бы защита брандмауэра Linux более надежной с настройками по умолчанию?

Моя беспроводная сеть настроена так, что подключение возможно только с заданных MAC-адресов, и для всех MAC-адресов (проводных и беспроводных) заранее выделены конкретные IP-адреса, поэтому мои дети и внуки жалуются, когда являются с последними i-Игрушками: они не могут пользоваться ими, пока не сообщат мне все чудовищно многочисленные подробности.

Майкл Дайер (Michael Dyer)

В принципе, предпочтительнее отдельное устройство, такое как Ваш маршрутизатор (роутер). Если брандмауэр запущен на том же компьютере, который он защищает, то несанкционированный доступ к нему позволит атакующему отключить брандмауэр. Брандмауэр на роутере защищает все устройства, а не только компьютер. С другой стороны, если Вы позволяете подключаться к сети устройствам, которыми не управляете, то брандмауэр на компьютере поможет защитить компьютер, если одно из этих устройств будет инфицировано вредоносным ПО. Брандмауэры прекрасно работают в компании, и осторожность никогда не бывает лишней, так как два брандмауэра предоставляют защиту на различных уровнях.

Также нужно убедиться, что у Вас не запущено ненужных сервисов, открывающих сетевые порты, и через роутер не перенаправляются ненужные порты. Существуют сайты, с помощью которых можно просканировать сеть извне и эффективно проверить настройки роутера, такие как Shields Up! (<http://www.grc.com>), и сайты для сканирования портов, например, <http://www.whatsmyip.org>. То же относится и к компьютерной сети — не запускайте ненужных сервисов и проверяйте сеть сканером портов, запустив его на дру-

гом компьютере локальной сети. В Linux можно воспользоваться `nmcli` или запустить сканеры портов с телефона или планшета с Android. Чтобы просканировать сеть с `nmcli`, скопировать

```
nmcli -v -sT 192.168.1.0/24
```

где 192.168.1.0/24 представляет Вашу сеть. Этот пример соответствует всем адресам, начиная с 192.168.1 — измените его для своей сети. Если хотите воспользоваться Android — лично у меня хорошо работали и *Net Scan*, и *Network Discovery*.

Ограничение доступа по MAC-адресам — плохая идея, так как MAC-адрес очень легко подделать. Это, конечно, еще один уровень защиты, но полагаться на него не стоит. Обеспечьте шифрование WPA или WPA2 и задайте длинный пароль, это защищает гораздо лучше.

4 Беспроводной диалог

Мне нужно настроить беспроводную сеть. У меня есть пара беспроводных USB-адаптеров и старый ноутбук Acer. Я хочу подключить один адаптер к публичной сети, а другой использовать как точку входа локальной сети для клуба, который раз в неделю собирается в гостиной. Локальная сеть будет доступна только нашей группе с помощью обычного ключа беспроводной сети, например, WEP или другого. Я хотел бы воспользоваться одной из версий Linux, чтобы закрыть доступ от любопытных глаз!

Один USB-адаптер нужен потому, что в здании нет сигнала, а на окнах остались решетки времен короля Георга, и для нормального приема приходится высовывать адаптер за окно! Подойдет ли обычный дистрибутив? Если да, то какой лучше? Графический интерфейс не нужен, так как компьютер будет только точкой доступа. Можно ли воспользоваться IPCop или чем-то еще? Вопрос в том, как настроить две беспроводных (USB) карты для различных целей — IP-адресацию публичной сети сохранять не нужно. Последний вопрос — как сделать, чтобы вход в систему в публичной сети был автоматическим, так как имя пользователя

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`.

понадобится вводить на web-странице, а не в настройках беспроводной сети. Буду рад вашим предложениям.

Тони Райдер (Tony Rider)

Вам должны подойти некоторые дистрибутивы для роутеров, хотя в нескольких из них для поддержки беспроводных адаптеров может потребоваться дополнительная настройка. Обычно подключение к Интернету в них



Коротко про...

Без паники

Иногда ядро Linux сталкивается с ситуацией, которую не в состоянии обработать, и, игнорируя совет на обложке величайшей из когда-либо написанных книг, паникует. Это проявляется в появлении в консоли загадочного сообщения (а если Вам по-настоящему не везет, оно будет скрыто экраном заставки) и мигающем светодиоде Caps Lock.

Если к этому моменту ядро смонтировало корневую файловую систему в режиме чтения и записи, оно также может записать на диск диагностическую информацию, «дамп ядра».

С ядром, предоставляемым дистрибутивом, вы теоретически никогда не увидите таких со-

общений. А если у вас собственное ядро, это явление обычно означает, что вы забыли включить поддержку контроллера дисков и файловой системы, используемой корневой файловой системой, и диски нельзя смонтировать.

Вам простительно будет решить, что все поггло, но ядро все еще способно на последний подвиг, даже если вы его убьете.

Ядро можно попросить перезагрузиться после возникновения паники. Если добавить `kernel.panic=10` в `/etc/sysctl.conf`, то ядро перезагрузится через 10 секунд; но чтобы прочесть этот файл, система должна добраться до монтирования корневой файловой системы. Более надеж-

ный способ — добавить `panic=10` к параметрам ядра в меню `Grub`.

Вы можете спросить, а в чем же смысл перезагрузки, если ядро будет снова паниковать и перезагружаться до бесконечности. Ответ лежит в возможности настройки альтернативной загрузки в `Grub`. Если она настроена, то паника ядра приведет к перезагрузке в альтернативное ядро, которое гарантированно работает. Это настоящая находка при проверке нового ядра на удаленном компьютере — и не такое позорище, как необходимость звонить кому-то по телефону и просить перезагрузить компьютер, потому что вы испортили конфигурацию ядра.

проводное, но так бывает не всегда — например, SmoothWall был изначально создан для разделения ISDN-подключения. Потенциальная проблема с двумя USB-адаптерами в том, что нужно обеспечить постоянно имен (обычно об этом заботится *udev*), поскольку ПО роутера использует имена интерфейсов для определения того, какой интерфейс подключен к Интернету, а какой к локальной сети. Однако в зависимости от беспроводных адаптеров могут понадобиться дополнительные действия, чтобы они работали с ядром SmoothWall. IPSec, который Вы упомянули, не работает с беспроводными сетями. Если Вы хотите управлять с его помощью беспроводной сетью, Вам также понадобится аппаратная точка доступа.

Для беспроводной сети можно воспользоваться готовым дистрибутивом с *hostapd*, но они работают только с определенными беспроводными адаптерами. На самом деле, по крайней мере один из двух беспроводных адаптеров должен поддерживать точку доступа — тот, который покрывает локальную сеть. Проверить это можно командой `iw` (она доступна по умолчанию не во всех дистрибутивах — возможно, ее придется установить).

```
iw list
```

Учтите, что это не то же самое, что команда `iwlist`. Поищите точку доступа под «Поддерживаемыми режимами интерфейса [Supported interface modes]», чтобы найти подходящий дистрибутив.

Есть и другая альтернатива — компактная, удобная и недорогая: любимый всеми компьютер с Linux — Raspberry Pi. Pi-Point (www.pi-point.co.uk) — производная Raspbian, которая превращает Pi в точку доступа беспроводной сети. Она устанавливается обычным образом: загрузите образ, запишите его на SD-карту командой `dd` и загрузите с нее Pi. Вам понадобится изменить файл `/etc/network/interfaces`, задав статический IP-адрес интерфейса локальной сети из другой подсети по отношению к подсети внешнего интерфейса. Также нужно изменить файл `/etc/hostapd/hostapd.conf`, указав в нем, какой

интерфейс используется как точка доступа локальной сети.

Pi-Point не хватает приятного web-интерфейса, но его можно настроить по SSH. Единственная штука, с которой могут возникнуть проблемы — аутентификация во внешней сети. Если это надо делать через web-интерфейс, можно воспользоваться текстовым браузером при подключении по SSH (это относится и к другим решениям). Если Вам повезет, то Wi-Fi будет использовать cookie для входа в систему, и Вы сможете задать подходящую страницу в качестве домашней страницы браузера и просто добавить браузер в автозагрузку; в противном случае каждый раз придется входить в систему вручную.

Какой бы вариант Вы ни выбрали, не пользуйтесь WEP — его невероятно легко взломать. Пользуйтесь WPA или WPA2. Еще нужно проверить, что у Вас есть право разделения подключения к публичной сети таким образом.

5 Файл ИНКОГНИТО

В 2009 году я создал файл, который затем сохранил. А сейчас не могу его открыть. Я не представляю, в какой программе создавал его. Я также не помню, было ли это в XP, Vista или Mint. Попробовал открыть его в *Scribus* и *LibreOffice*, но безуспешно. Расширение файла — `.dtp`, и он довольно большой, поэтому в нем, видимо, есть картинки и он был сделан в одном из офисных приложений.

Джошуа [Joshua]

О, эти прелести идентификации файлов по трем последним буквам имени! Разумно предположить, что файл с расширением `.dtp` был создан в пакете DTP, но в котором? Форматы файлов у большинства из них несовместимы.

К счастью, в Linux есть более удобный способ идентификации файлов. Команда `file` —

```
file somefile.dtp
```

заглянет внутрь файла, посмотрит, какой программой он был создан, и выведет информацию

о содержимом файла и об этой программе. Это позволит как минимум определить программу, в которой создан файл. Определив ее, Вы сможете либо открыть свой файл в этой программе, если она у Вас уцелела, либо поискать в Сети утилиту для конвертации. Возможно, Вы также сможете импортировать его в Google Docs, но для файла в проприетарном формате DTP это не слишком вероятно.

В качестве последнего прибежища можно извлечь из файла текст командой `strings`, если Вам нужен именно текст. ASCII-строки в двоичном файле ищет команда

```
strings --encoding={S,b,l} -n 2 somefile.dtp >somefile.txt
```

Если она что-то найдет, Вы сможете найти еще больше, добавив к команде параметры — в частности, параметры кодировки. Подробную информацию об использовании `strings` можно узнать, запустив команды

```
man strings
strings --help
```

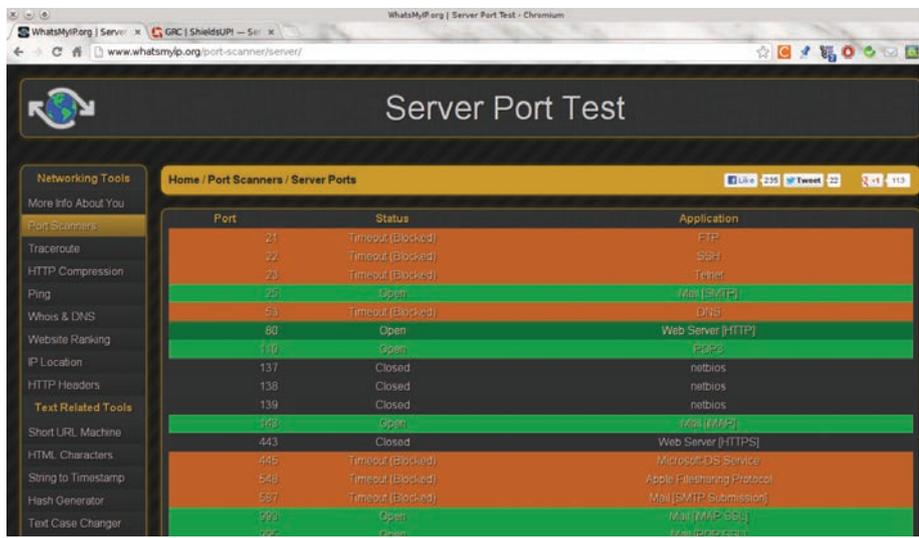
6 Linux на iBook

В у меня есть старый ноутбук Blueberry PPC iBook, на котором я собирался только писать тексты. Ни для чего другого он не годен, ну я и попробовал установить на него Mint. Мне удалось установить ОС, но при загрузке она доходит до запроса пароля и перестает реагировать на клавиатуру. Я попробовал переустановить систему, но это не помогло. Диска с оригиналом ОС ноутбука у меня нет, и ее восстановить тоже нельзя. Есть ли варианты? Или его просто можно выбросить?

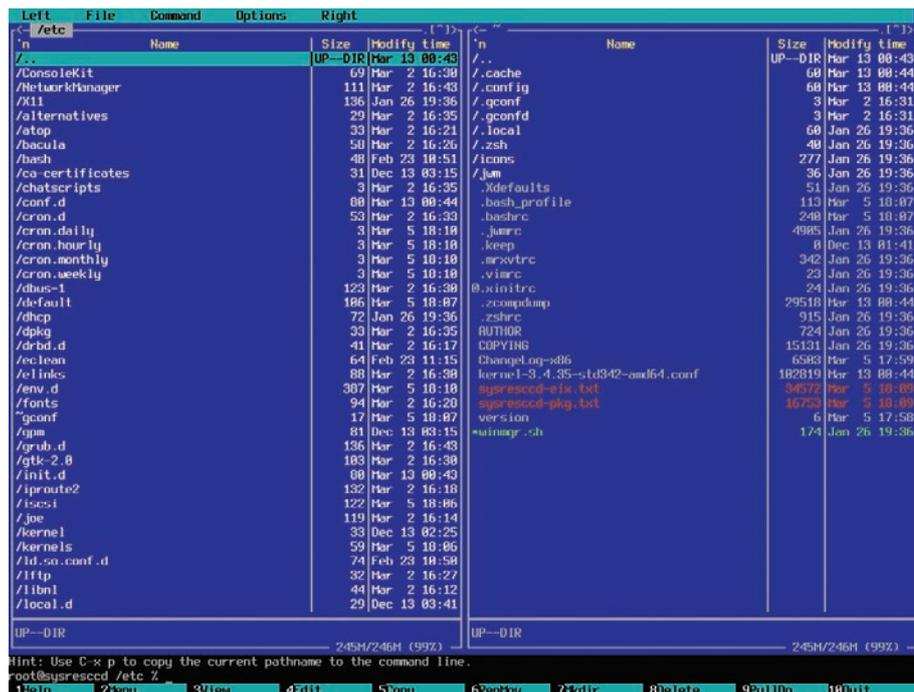
Алан Турпин [Alan Turpin]

О Пароль запрашивается в графическом режиме или в консоли? Если в графическом, попробуйте включить опцию восстановления и загрузиться в консольном режиме. Если все получится, то у Вас, вероятно, не хватает оперативной памяти для загрузки всего рабочего стола. У iBook очень небольшой объем памяти — 32 или 64 МБ; поменяв «железо», его можно обновить до 320 МБ — поэтому вполне вероятно, что Вам просто не хватает ресурсов. Сами устройства, похоже, в порядке, раз Вы могли пользоваться клавиатурой во время установки.

Linux, наверное, можно установить на этот ноутбук, но Mint слишком тяжеловесен. Для машинки с ограниченными ресурсами нужно взять дистрибутив полегче. Можете попробовать Lubuntu — это Ubuntu с рабочим столом LXDE, доступный как для x86, так и для PowerPC. Либо можно установить PPC-версию Debian с диска сетевой установки. Это диск с минимальным содержанием, достаточным для перевода системы в состояние, в котором она может запустить установщик, а все остальное будет загружаться во время установки. С установщиком Debian можно точно указать, что нужно установить, и у Вас может получиться легкая рабочая среда, которая лучше подходит для этого ноутбука, чем Cinnamon от Mint.



➤ Сайт для сканирования портов позволит посмотреть, хорошо ли работает брандмауэр.



» Ну зачем вам графический интерфейс? Можно прекрасно обойтись *Midnight Commander* и текстовыми утилитами.

Рабочих столов несколько, но *LXDE* — хороший компромисс между низкими требованиями к ресурсам и знакомой рабочей средой.

Если Вы хотите писать только текстовые заметки, нужен ли Вам вообще рабочий стол?

Существует огромное количество консольных текстовых редакторов и файловых менеджеров. Вооружившись только *nano* и *mc* (*Midnight Commander*), Вы сможете сделать многое и без графических приложений. **lxf**

Помогите нам помочь вам

Ежемесячно мы получаем несколько писем, на которые не в состоянии ответить, поскольку проблема описана в них недостаточно полно. Чтобы дать вам наилучший ответ, нам 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
```



Часто задаваемые вопросы

Разделы и файловые системы

» Зачем нам столько разделов и файловых систем в Linux?

Жесткий диск разбивается на разделы — отдельные области, каждая из которых ведет себя как отдельный диск. Для хранения данных на каждом разделе используется файловая система (ФС).

» А почему в Linux несколько разделов? В Windows хватает одного.

Хранение данных разных типов на разных разделах повышает производительность и безопасность. Эффективнее хранить данные подкачки в отдельной файловой системе, чем в файле или каталоге файловой системы корневого раздела. Некорректно работающий процесс может испортить раздел подкачки, но другого не тронет.

» Мне достаточно раздела подкачки и корневого раздела, как в Ubuntu?

Да, но это не оптимальный вариант. В настольных системах (включая ноутбуки) должен быть по меньшей мере еще один отдельный раздел для домашнего каталога. Тогда ваши персональные настройки и данные хранятся отдельно от файлов операционной системы.

» А зачем так делать?

Когда-нибудь вы захотите попробовать другой дистрибутив или установить более новую версию текущего дистрибутива. Если все файлы находятся в корневом разделе, они будут уничтожены во время установки — и перед установкой придется делать резервную копию всех файлов и настроек. Если домашний каталог расположен на отдельном разделе, нормальный установщик оставит ваши данные нетронутыми. После установки вы сможете загрузить компьютер и продолжить работу

с ним точно так же, как в прежней системе.

» Нужны ли мне еще какие-нибудь разделы?

Для серверных систем неплохо создать отдельную файловую систему для `/var`. Это также полезно для настольных систем, так как процесс, заполняющий системные логи, которые находятся в `/var/log`, не сможет занять все место на диске, на котором находятся другие файлы операционной системы.

» Так много типов файловых систем... Зачем нужны ext2, ext3, reiserfs, xfs, jfs?

У каждой файловой системы есть свои сильные и слабые стороны. Ext2 — оригинальная файловая система Linux, и она очень быстрая, но не слишком хорошо справляется с внезапными сбоями системы.

Ext3 — журналируемая файловая система, что ускоряет и повышает надежность восстановления после падения системы или отключения питания, но немного снижает производительность. Reiserfs особенно хорошо работает с маленькими файлами, как с точки зрения эффективного использования пространства, так и скорости. XFS очень быстро работает с большими файлами, но плохо справляется со сбоями питания, поэтому ею лучше пользоваться на устройствах, поддерживаемых батареями. Ext4 сочетает в себе преимущества многих файловых систем, а BTRFS — новая, но обещает гораздо больше.

» Какая файловая система лучшая?

Вы бы еще спросили, что лучше — *Vi* или *Emacs*, *KDE* или *Gnome*? С вашего разрешения, на это я не стану отвечать, спасибо.



LXF Hot Picks



Майк Сондерс

Издавив самые недостижимые и укромные уголки Интернета, Майк точно знает, где таятся главные сокровища открытого кода.

Tiled » Mosh » Mp3splt » i7z » Terminator » JSesh » The Silver Searcher » Irrlamb » QNetWalk » Halttimer » Torsocks

Редактор карт

Tiled

Версия: 0.9.0 Сайт: www.mapeditor.org

Разве этой программе место не в разделе HotGames? Именно этот вопрос мы задавали себе, но после долгих размышлений, чесания в затылке и подергивания своих солидных Unix-бородич, мы решили вместо этого выделить ей первое место в данном материале. *Tiled* явно будет в большей мере использоваться разработчиками игр, однако очень пригодится

и в других, неигровых программах — например, образовательных.

Tiled — это редактор карт, но не из тех, что позволяют создавать документы для военно-топографической съемки. Нет, вместо этого он поможет вам создавать целые миры из мозаики. Вам не придется ничего рисовать пиксель за пикселем — здесь действует принцип использования и перегруппировки малых изображений, чтобы получить большие. Почти все платформенные и ролевые игры на старых 8- и 16-битных консолях использовали этот принцип, и даже в лучших



» *Mana World* — популярная ролевка с открытым кодом, созданная в *Tiled*.

играх с «реалистичной» графикой можно было заметить раскладку на основе сетки, скрепляющей элементы.

Чтобы создать карту, щелкните File > New и выберите размер. Здесь вы можете указать размеры пикселей для отдельных плиток, которые будете использовать, и количество плиток в готовой карте по вертикали и горизонтали. Помимо обычной 2D-раскладки, *Tiled* поддерживает изометрические карты для псевдо-3D-эффекта. Открыв новую карту, вы должны загрузить набор плиток через Map > New Tileset. Это большое изображение, разделенное на отдельные плитки (обычно 32×32 или 64×64 пикселя) — в Интернете полно бесплатных примеров.

Теперь начинайте создавать карту. Справа располагаются плитки из предлагаемого набора, а в панели инструментов есть кнопка Stamp для размещения отдельной плитки и кнопка Fill для заполнения всей области одинаковыми плитками. Здесь не особо много функций и опций, и полученные в результате файлы XML могут использоваться многими игровыми движками (либо напишите собственный парсер). На самом деле, масса игр были созданы с помощью *Tiled* — просмотрите список на <http://tinyurl.com/d4fouor> и подумайте, не хотите ли вы его обогатить.

«Tiled помогает создавать миры из плиток мозаики.»

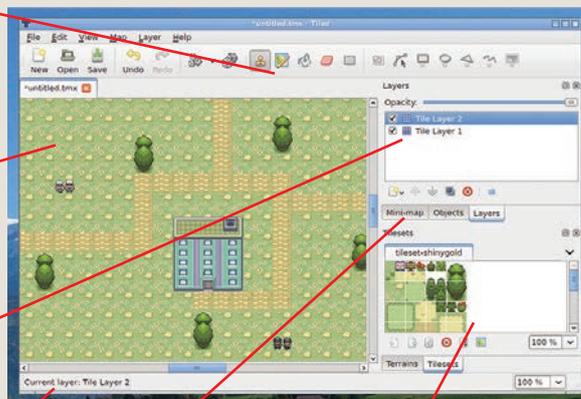
Исследуем интерфейс Tiled

Панель инструментов
Вы найдете инструменты для размещения отдельных плиток, заполнения областей плитками или удаления выделенных областей.

Редактор карт
Здесь, собственно, все и происходит. Прокручивайте с помощью мыши и используйте Ctrl + колесо мыши для изменения масштаба.

Слой
Применяйте слои для работы с объектами на переднем плане и на фоне. Прозрачность слоев можно менять.

Состояние
Отображает положение плитки по осям X и Y, когда мышь на карте, а также используемый слой.



Мини-карта
Щелкните по этой вкладке, чтобы перейти на мини-карту (быстрый способ навигации по всей карте).

Набор плиток
Здесь появятся загруженные плитки. Щелкайте и перетаскивайте для выделения нескольких плиток; затем их можно перенести в панель редактирования.

Мобильная оболочка

Mosh

Версия: 1.2.3 Сайт: <http://mosh.mit.edu>

Сидите вы себе в поезде, стучите по ноутбуку, у вас открыта пара сессий SSH. И вдруг до вас доходит, что поезд стоит, причем на вашей станции. Вы спешно захлопываете ноутбук, хватаете свои вещи и выскакиваете вон. Потом вы заходите в кафе с бесплатным Wi-Fi, берете кофе-латте, усаживаетесь, открываете свой ноутбук... И видите, что ваши сессии SSH повисли, потому что сейчас вы в другой сети и с другим IP-адресом.

SSH, конечно, крут, но для путешествий не очень удобен. В пути вы перемещаетесь из одной сети в другую, и придется постоянно входить и выходить (или использовать нечто вроде экрана с функциями соединения и разъединения). Наилучшее решение предлагает *Mosh*, мобильная оболочка. *Mosh* можно использовать так же, как SSH, но у вас должен быть установлен исполняемый сервер *mosh-server* на удаленной машине. И вы соединяетесь с помощью команды

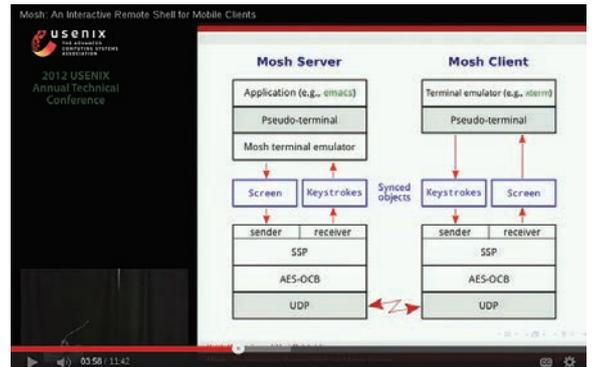
```
mosh username@example.org
```

Это позволит вам зайти через SSH, а потом *Mosh* запустит *mosh-server* на удаленном компьютере, который затем установит соединение UDP между двумя компьютерами.

Тогда сессия сохранится в рабочем состоянии даже при смене сети, хотя основной SSH в обычном состоянии просто испустит дух. Есть здесь и другие преимущества: при вводе чего-то через обычное соединение SSH, SSH ждет ответа сервера и только потом показывает результат. *Mosh* дает ответ немедленно, и это создает ощущение очень резкого отклика в медленных или ненадежных соединениях.

Больше всего в *Mosh* нам понравилась простота интеграции. Не нужно установ-

«Лучше всего в Mosh простота интеграции.»



➤ Загляните на <http://youtu.be/XsIxNYI0oyU> — там есть презентация, которая показывает работу *Mosh* изнутри.

ливать кучу бинарников только для root, фоновых демонов или скриптов настройки. Ее зависимости очень обычные, и компиляция из исходника не должна превратиться для вас в кошмар. А далее *Mosh* не очень отличается от SSH в использовании, и для процесса входа можно задействовать опции SSH с помощью флага `--ssh`.

Разработчики планируют в будущих релизах пророс X11 и родное приложение для Android; а для желающих заглянуть внутрь программы есть обильная техническая документация.

Сплиттер аудиофайлов

Mp3splt

Версия: 0.7.3 Сайт: <http://mp3splt.sourceforge.net>

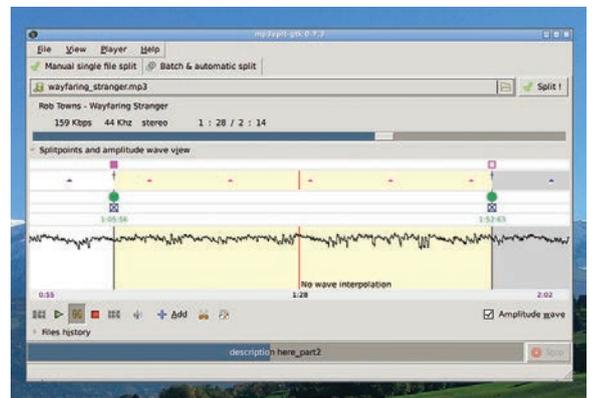
Вероятно, вы уже знаете, что стандартный набор инструментов командной строки Linux включает инструменты для нарезки файлов. Например, `split` отлично работает, если вам надо порубить огромный файл, чтобы перенести его на USB-брелки, и вы даже можете использовать `dd`, но он выполняет только самую примитивную работу. Если же вам надо нарезать медиа-файл и сшить куски, чтобы плеер не спотыкался при воспроизведении, то нужен специализированный инструмент.

Обычный способ нарезки файла MP3 — импортировать его в аудиоредактор типа *Audacity*, нарезать и заново закодировать в формат MP3. Однако перекодирование уже кодированного файла — не лучшая идея для сохранения качества аудио, ведь каждый раз вы что-то теряете. *Mp3splt* особенно интересен, поскольку он ничего не перекодирует и оставляет исходное аудио нетронутым.

Есть инструмент командной строки в виде `mp3splt`, но его графический эквивалент (`mp3splt-gtk`) позволяет справиться с работой, не вбивая вручную сотни параметров. Чтобы использовать это приложение, загрузите файл MP3 и щелкните по кнопке Play. Когда решите вставить точку нарезки — то есть место, где программа должна разрезать аудиозапись на два файла — щелкните по кнопке Pause и затем Add. Когда закончите, щелкните Split вверху.

Это отлично для простых операций, но *Mp3splt* способен на большее: если у вас есть аудиофайл целого альбома и вы хотите порезать его на треки, можете взять временной режим на TrackType.org

«Перекодирование уже кодированного файла — не лучшая идея.»



➤ Щелкните по флажку **Amplitude wave**, чтобы упростить поиск тишины в аудио.

(там огромная база данных по альбомам) и настроить точки нарезки автоматически. Можно также велеть *Mp3splt* искать куски тишины в аудиофайле и размещать точки нарезки на паузах.

Вы даже можете выполнять пакетную обработку данных с несколькими файлами, менять тэги ID3 на лету и выбирать имя файла-результата на основе этих тэгов. *Mp3splt* — замечательно умная программа, и хотя ее основная работа весьма проста, нам нравится, что она предоставляет почти все мыслимые функции для ее выполнения.

Инструмент отчета о работе CPU

i7z

Версия: 0.27.2 Сайт: <http://code.google.com/p/i7z>

Мы тоскуем по дням, когда на ПК были кнопки Turbo и жидкокристаллические счетчики частоты. Если вы никогда не видели кнопки Turbo или в те времена были слишком поглощены восхищением несравненной Amiga и вам было не до глупостей ПК, то объясняем специально для вас: многие ПК в начале 1990-х имели кнопку для снижения производительности CPU, чтобы старые игры (жестко запрограммированные на конкретные МГц) работали нормально. Название Turbo было глупым, потому что это был режим ПК по умолчанию — кнопку следовало назвать «Замедление для старых игр». Но с позиций маркетинга это же не круто...

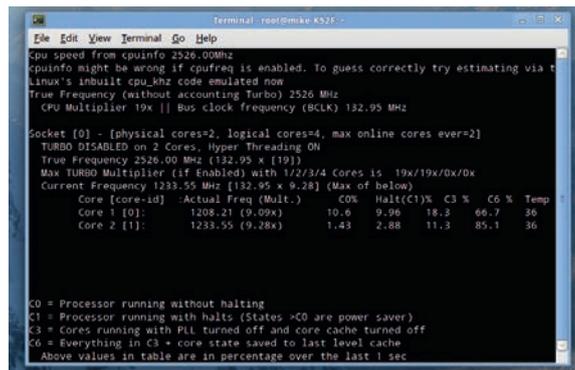
Ну да ладно. Хотя сегодня нас не так уж волнуют МГц (ну хорошо, ГГц), все же мы хотим нагружать наши CPU по максимуму. На любом стандартном компьютере с Linux вы можете ввести

```
cat /proc/cpuinfo
```

Эта команда берет в ядре основные статистические данные по процессору, на

пример, название модели, размер кэша, и т.д. Она не выдаст вам особо детальную информацию по процессорам Intel Core iX — Core i3, i5 и i7; здесь и вступает в дело *i7z*. *i7z* — это инструмент командной строки для предоставления подробной информации по этой линии CPU. Вы можете взять на сайте проекта прекомпилированные бинарники со статическими ссылками, которые должны работать практически на любом дистрибутиве Linux; однако помните, что вы должны запускать их от имени root.

По умолчанию *i7z* отображает список CPU и их физических ядер, а также список (ежесекундно обновляемый), который показывает, сколько времени ядра провели в разных состояниях C. Эти состояния C



► Батарея нашего ноутбука счастлива, потому что CPU в режиме экономии энергопотребления C6 по большей части отдыхает.

представляют, с каким энергопотреблением работает CPU: например, в C0 процессор работает на полную мощность. C1 немного менее энергозатратно, и чем выше номер состояния C, тем больше функций CPU отключено, чтобы продлить заряд батареи. Весьма удобно, что есть небольшая таблица, которая объясняет, что делают разные состояния C.

Если ваше окно терминала больше стандартного размера 80×24 символа, вы также увидите столбец отображения температуры для каждого ядра. И если вы предпочитаете выводить результат в файл, а не на монитор, то для этого тоже имеется опция.

«По умолчанию *i7z* отображает список CPU и их физических ядер.»

Усовершенствованный терминал

Terminator

Версия: 0.96 Сайт: www.tenshu.net/p/terminator.html

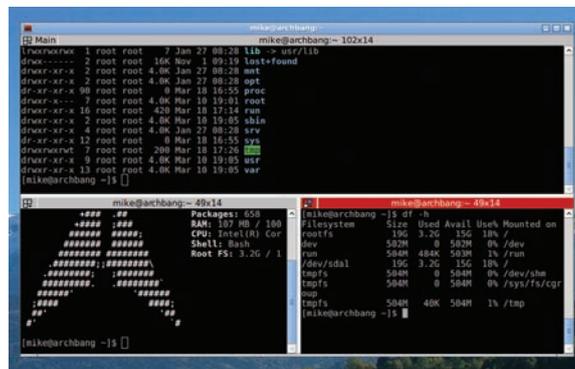
Сейчас, в 2013 году, уже трудно представить себе, что скромное окно терминала по-прежнему оставляет простор для новых функций. Оригинальный *xterm* находился в разработке с 1984 года, и с тех пор появилось множество разных эмуляторов терминала. Одна из популярных недавних разработок — терминал со вкладками: так же, как и в браузере со вкладками, вы можете создавать новые сессии внутри основного окна программы, не засоряя панель задач.

В *Terminator* эта функция есть, но она уже стала рядовой, и не стоит водить вокруг нее хороводы. Зато в нем есть две другие функции, которые стоит исследовать: первая — это разделение терминала. Нажмите на Ctrl+Shift+O внутри окна *Terminator*, и текущая сессия разделится на две панели. Нажмите Ctrl+Shift+E — и получите вертикальное разделение. Если вы озачочены местом на экране, эта функция вас здорово выручит, создавая весьма

креативные раскладки, где самым важным панелям терминала будет отведено достаточно места; однако можно оставить незанятыми небольшие участки, чтобы просматривать логи, каналы IRC и т.п.

Еще одна замечательная функция, которую поддерживает *Terminator* — это группировка. В верхней левой части панели терминала вы увидите небольшой значок, похожий на блок из *Тетриса*. Щелкните по нему и выберите New group. Теперь можете ввести имя, и выделенная панель терминала принадлежит к этой группе. А теперь самое лучшее: в меню под *Тетрис*-подобным блоком вы увидите разные опции Broadcast. Используя их, вы сможете отправить то, что ввели с клавиатуры, на активную панель, на другие панели группы или на все панели. Вряд ли вы будете пользоваться этим ежедневно, но в некоторых обстоятельствах, например, при тестировании нескольких связанных друг с другом программ с одинаковым вводом, эта опция творит чудеса.

«Terminator легко настраивается, и можно сохранять раскладки.»



► Красный цвет заголовка отмечает активную на данный момент панель, но при необходимости вы можете направить то, что вводите, на все панели.

туры, на активную панель, на другие панели группы или на все панели. Вряд ли вы будете пользоваться этим ежедневно, но в некоторых обстоятельствах, например, при тестировании нескольких связанных друг с другом программ с одинаковым вводом, эта опция творит чудеса.

Terminator легко настраивается (щелкните правой кнопкой внутри окна и выберите Preferences), и можно сохранять свои раскладки для повторного использования в будущем. Также он позволяет изменять программные привязки к клавишам, если вас не устроит набор по умолчанию.

Редактор иероглифов

JSesh

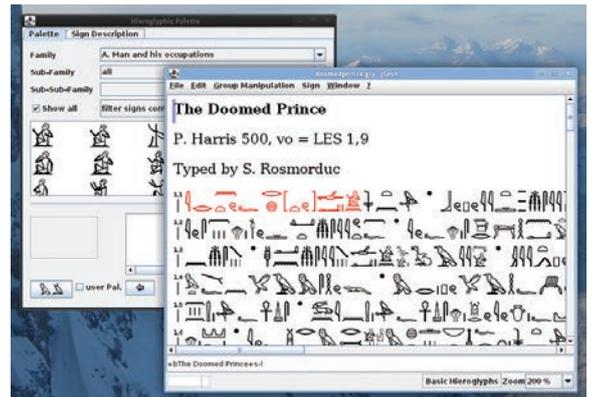
Версия: 6.3.2 Сайт: <http://jsesh.qenherkhopeshef.org>

Слышали шутку про древнеегипетского фараона? Звучит примерно так: птица, птица, глаз, колесо, человек в поклоне, птица, глаз, змея, птица, рука. Мы бы включили эту белиберду в настоящих иероглифах, да поопасались насчет того, что в этом случае может произойти, когда журнал дойдет до печати — уж точно тогда какой-нибудь расвирепешивший бедняга позвонил бы Эффи, решив, что наши PDF скамкались.

И все же, оказывается, есть программы для редактирования иероглифов, и они являют собой доказательство изобретательности и трудолюбия сообщества открытого кода. *JSesh* написан на Java, а значит, вам не придется гоняться за кучей зависимостей перед началом работы — просто скачайте файл `.jar` и запустите на нем `java -jar` для его установки в домашней директории. Обратная сторона использования им виджетов Java и Swing — старомодный GUI, однако для подобного рода приложений вряд ли это большая проблема.

Слева *JSesh* представляет набор иероглифов, разделенный на категории (например, мужчины, женщины, животные, строения, еда). Вы можете вывести описание любого из них, однако важно отметить, что *JSesh* — это редактор, и его целью не является перевод или обучение вас принципам письма. Двойной щелчок по иероглифу устанавливает его в панель редактирования справа; помимо иероглифов, вы также можете вводить обычные латинские буквы.

Можно экспортировать документы во многих форматах, в том числе PNG, PDF и HTML. *JSesh* прекрасно работает с буфером обмена: ни одна другая программа не поймет отдельно взятый иероглиф, поэтому когда вы копируете их набор в *JSesh*



» Эта легенда датируется временем правления XVIII династии — примерно 1400 год до н.э. Вот деньки были, да?

в буфер обмена, он сохраняет строки как отдельные изображения. Затем вы можете вставить их, например, в *LibreOffice*, и вам не придется возиться с экспортом и загрузкой изображений вручную.

Программа содержит массу документации с примерами (загляните в папку Египетские тексты [EgyptianTexts] в директории установки), в том числе классические истории из папирусов и надписей на саркофагах. Файл `demo_mdc.gly` даст вам подсказки по пользованию программой. *JSesh* — прекрасный способ изучения системы письменности и истории, да и работа с ним — истинное удовольствие; попробуйте использовать *JSesh*, чтобы составить записку своему сантехнику.

«JSesh — прекрасный способ изучения истории, и удовольствие.»

Программа поиска по исходному коду

The Silver Searcher

Версия: 0.14 Сайт: <http://bit.ly/XGtiR0>

Ах, *grep*. Старый добрый *grep*. Где бы мы были без этого классического, разумного, безотказного (он существует в Unix с 1973!) инструмента поиска по тексту? Он просто превосходен, если вам надо перелопатить кучу текстовых файлов и найти некое слово; но он не лишен ограничений. Автор *Ack* (www.betterthangrep.com) осознал это — и создал программу поиска, в которой были дополнительные прелести для программистов: например, лучший поиск по исходному коду и способность игнорировать определенные директории.

Но хоть *Ack* и отличается «невероятной быстротой», Джеффу Гриру [Geoff Greer] этого было мало, и он решил создать еще более быструю версию — и вот теперь у нас есть *The Silver Searcher*. Его команда — `ag`, и запускать его надо так:

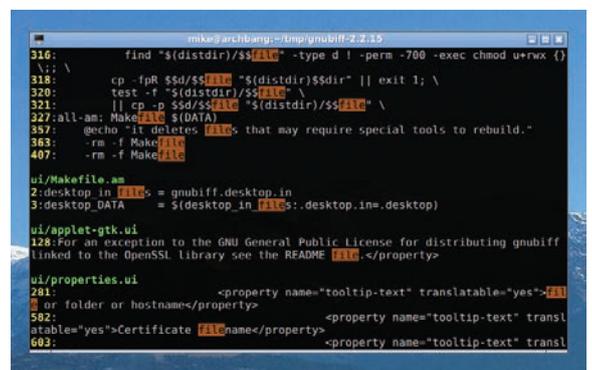
```
ag pattern filenames
```

Итак, чтобы, например, найти примеры «ошибки [error]» в файлах исходного ко-

да в текущей директории, вы используете `ag error *`. Можно осуществлять поиск с помощью регулярных выражений: `ag [a-z]ile *` отображает строки кода, содержащие файл, части, группы, и т.д. Очень удобно, что *The Silver Searcher* делает результат цветным, что позволяет его сразу увидеть — особенно если последние несколько часов вы неотрывно смотрели на окно терминала. Имена файлов, номера строк и результаты совпадений все окрашены в разные цвета, поэтому результаты отлично видны.

По умолчанию *The Silver Searcher* игнорирует файлы, указанные в `.gitignore` и `.hgignore`. Если вы не используете систему контроля версий и хотите вручную на-

«Ценный компонент инструментария любого программиста.»



» Быстрый, как молния, красочный, дружелюбный к исходному коду поиск — это просто *grep* королей.

строить путь к игнорируемым файлам, можете ввести их в `.agignore` в своей домашней директории. А если хотите искать абсолютно везде, дайте команду `ag` игнорировать игнорируемые файлы (если вы следите за тем, что мы рассказываем, используя опцию `-c`).

Разработчик *The Silver Searcher* стремился сделать `ag` как можно более быстрой, и дошел даже до того, что написал собственную реализацию `scandir()`, и оно того стоило — она намного быстрее, чем *Ack*. Но что куда важнее, она намного приятнее, чем *grep*, когда дело касается поиска по исходному коду, и станет ценным компонентом инструментария любого программиста.

HotGames Развлекательные приложения

Физическая игра

Irrlamb

Версия: 0.1.1 Сайт: <http://bit.ly/4U6iLh>

Мы любим физические игры: уж очень эффективно они задействуют энергию CPU. Игра легко может заставить систему использовать свои ресурсы по максимуму, отображая десятки миллиардов полигонов в секунду или добавляя слой за слоем забавные световые эффекты, но все это лишь внешняя красота. Мы любим те игры, где физические законы действительно работают, где есть песочница для экспериментов и где вам некого обвинить в своих ошибках, кроме себя.

Irrlamb соответствует всем этим требованиям — хотя находится пока еще на очень ранней стадии разработки, так что в ней не хватает большей части контента. Вкратце, вы управляете голубой сферой, используя клавиши E, S, D и F, чтобы передвигаться, и пробел для прыжка. Внутри 3D-арены вам нужно заставить вашу сферу столкнуться

с золотыми сферами, обходя различные препятствия на вашем пути. Чтобы передвинуть камеру, просто покрутите мышью.

Эта концепция кажется обманчиво простой, но некоторые испытания достаточно трудны и интересны. У вас есть множество ящиков, которые могут перевернуться, вращающиеся бревна, которые пытаются сбить вашу сферу с пути, и прочие прелести, сильно усложняющие жизнь. Но даже и на самых сложных уровнях вам ясно: своими промахами вы обязаны только себе, а не недочетам игры. Этому способствует отлично проработанная физика — попробуйте под-

«И на сложных уровнях ясно: промахами вы обязаны себе.»



» Чтобы прыгать по этим шатким ящикам, нужны отличные рефлекссы.

прыгнуть в воздухе и нажать E, чтобы перевернуться, а затем увеличить масштаб, когда вы приземлитесь, и в игру вступит сила трения.

Жаль, что в *Irrlamb* маловато уровней. Остается ждать новых, потому что пока игра еще находится на слишком раннем этапе разработки; но когда вы пройдете по базовым и продвинутым уровням, а также по двум уровням *Skate*, вам явно захочется еще. Если вы набили руку в *Blender* и *Lua*, можете создавать свои собственные уровни — загляните в [wiki](#) на сайте игры, там имеются руководства.

Имитатор системного администратора

QNetWalk

Версия: 1.4 Сайт: <http://bit.ly/POG400>

В наш паб (где, честно говоря, мы большую часть своего времени и сидим) иногда заходят коллеги-компьютерщики, работающие сисадминами. Эта должность подразумевает очень многое, и объем работы варьируется в зависимости от ОС. Например, админ Windows живет в постоянном кошмаре на тему дыр в системе безопасности, шпионского ПО и безбашенных пользователей, портящих все подряд. А вот Linux-админ, наоборот, уверен в подкованности пользователей и периодически пишет какой-нибудь скрипт Perl — пусть, мол, выполняет 99 % его задач.

Так или иначе, быть системным администратором — вряд ли самое увлекательное занятие на планете; поэтому у нас глаза полезли на лоб от удивления, когда мы наткнулись на *QNetWalk*. Это Qt-версия головоломки, уже не первый год

существующей на разных платформах, и даже можно найти Flash-версии для игры в браузере. Основные зависимости — только Qt 4.3 и SDL; скачав [qnetwalk-1.4.tar.gz](#) с сайта игры, вы найдете внутри прекомпилированный бинарник.

В *QNetWalk* вам дается сетка из сетевых компонентов с компьютерами, кабелем и центральным сервером. Ваша цель — вращать компоненты до тех пор, пока они не соединятся воедино, то есть пока они все не соединятся с сервером. Компьютерный экран загорится синим, показывая, что компьютер может достичь сервера, и щелкая средней

«Она заняла нас на пару часов. Затем мы вернулись в паб.»



» Черт возьми, надо с этим разбираться. Ну что бы компьютерам просто не использовать Wi-Fi...

клавишей мыши по компоненту, вы можете заблокировать его.

Здесь есть множество различных уровней сложности, причем последние стадии предлагают невероятно сложные головоломки в более крупных сетках. На самом деле, эта игра не имеет ничего общего с системным администрированием (если только вы не являетесь сетевым администратором с весьма своеобразным парком оборудования), но это хорошая разминка для мозгов, и она заняла нас на пару часов. Затем мы снова вернулись в паб.

АССИСТЕНТ ВЫКЛЮЧЕНИЯ

Halttimer

Версия: 0.3.1 Сайт: <http://bit.ly/YE8AEv>

Очевидно, что Linux активно прокладывает себе дорогу на рынок мобильных телефонов благодаря Android, но он также становится все популярнее на домашних компьютерных кинотеатрах (HTPC). Здесь широкий выбор медиа-плееров, и плюс к тому — настраиваемость, на которую способна только открытая платформа. А главное, вы сами можете подобрать для себя нужные решения.

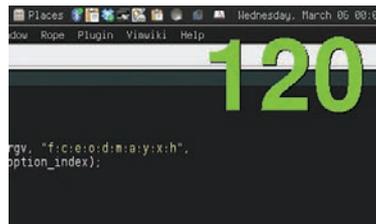
Например, если у вас есть компьютер с Linux, подключенный к телевизору для воспроизведения фильмов и прочего, вам захочется, чтобы он вел себя, как обычный телевизор — т.е. выключался нажатием всего одной кнопки на пульте дистанционного управления, без возни с мышью и всякими кнопками в менеджере окон.

Здесь поможет *Halttimer*: если на вашем компьютере Linux есть устройство инфракрасного ввода, можете использовать его вместе с пакетом **LIRC**, чтобы запустить последовательность выключения, когда будет нажата определенная кнопка

на вашем пульте. Вы можете определить кнопку в вашем файле `~/lirc`, и когда она будет нажата, появится таймер, отсчитывающий время до выключения ПК.

Можно настроить размер, цвет и размещение шрифта таймера. И даже если у вас нет инфракрасного устройства ввода и вы используете стандартную клавиатуру ПК для управления HTPC, вы все равно можете задать кнопку для запуска *Halttimer*.

В любом случае, возможность выключения одним нажатием вкупе с симпатичным таймером сделают ваш компьютер Linux намного более похожим на обычный телевизор.



➤ *Halttimer* может ожидать в фоновом режиме и приступить к действию, получив SIGUSR1, так что его легко запускать из скриптов.

АССИСТЕНТ TOR

Torsocks

Версия: 1.3 Сайт: <http://code.google.com/p/torsocks>

Анонимность в Интернет — горячая тема в наши дни. Правила постоянно изыскивают новые способы проникнуть в наши данные, рекламные сети хотят следовать за каждым нашим щелчком мыши, а есть еще и всякие придурки, жаждущие кому-нибудь нагадить просто из вредности характера. Tor (www.torproject.org) по мере сил стремится облегчить эту проблему и позволяет посещать сайты, не открывая местонахождение посетителя; однако его использование по большей части привязывалось к просмотру web-ресурсов.

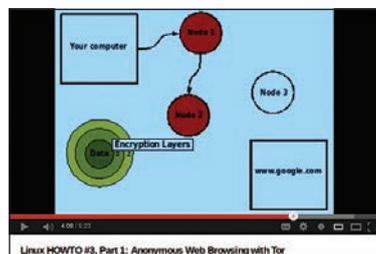
Сейчас стало возможно использовать Tor с другими программами, не являющимися браузерами, однако, как это часто случается, «возможно» — не то же самое, что «легко». Загляните в web-справочники по использованию Tor с SSH, например — даже простейшие из них подразумевают необходимость лезть в файл настройки. Если мы намерены поддержать тот

постулат, что анонимность онлайн — дело важное, настройку нужно упростить.

Torsocks делает это, позволяя вам запускать многие программы через сеть Tor без необходимости настраивать их. Например, если на вашей машине установлен Tor, вы можете ввести

```
torsocks ssh username@example.org
```

Здесь соединение SSH будет направлено через сеть Tor — не требуется никакой специальной настройки SSH. Есть поддержка многих других программ; список вы найдете на сайте *Torsocks*. **LXF**



➤ Незнакомы с Tor? Дружеское его представление — на <http://youtu.be/rVilDmCPonk>.

ТАКЖЕ ВЫШЛИ

Новые и обновленные программы, тоже достойные внимания...

➤ **openAviToGif 0.4**

Конвертирует видеофайлы AVI в анимированные GIF.

<http://sf.net/projects/openavitogif>

➤ **4M Linux 6.0**

Миниатюрный дистрибутив для мультимедиа, мини-сервера и игр в текстовом режиме.

<http://4mlinux.com>

➤ **Monitorix 3.1**

Легковесный инструмент мониторинга системы, поддерживающий множество сервисов.

www.monitorix.org



➤ Отслеживайте использование файловой системы, нагрузки на систему и нагрева CPU.

➤ **Pynguin 0.3**

Основанное на Python приложение с черепахой графикой, редактором и интерпретатором.

<https://code.google.com/p/pynguin>

➤ **Fisicalab 0.3.2**

Образовательное приложение для решения физических задач.

www.nongnu.org/fisicalab

➤ **Beets 1.13b**

Система управления библиотекой мультимедиа, разработанная для «одержимых музыкальной техникой».

<http://beets.radbox.org>

➤ **Bochs 2.6**

Мощный эмулятор ПК, особенно полезный любителям-разработчикам операционных систем.

<http://bochs.sourceforge.net>



➤ Хотите посмотреть, как выглядела Windows? *Bochs* вам поможет.

На диске

Лучшие развлечения с открытым кодом



Мы взяли Fedora 18 XFCE и наштапиговали ее таким количеством игр, которое только можно вместить в блестящий пластик-металлический диск. Результат — игровой дистрибутив LXF. Здесь более 100 игр — развлекайтесь! А мы расскажем вам о 10 играх, которые нас особо впечатлили. Однако, учитывая, что их на диске еще 90, у вас есть чем поживиться, если те, о которых мы рассказали, не утолили ваш голод.

Нельзя сказать, что все они совершенны в плане графики, но нам показалось, что это-то и неплохо. Бывают случаи, когда отличная графика делает игру выдающейся, но чаще дело все же не в графике, а в процессе.

Итак, найдите окно в своем расписании, прогоните детей спать, прихватите какую-нибудь нездоровую пищу, установите наш игровой дистрибутив — и готовьтесь к потехе с открытым кодом.

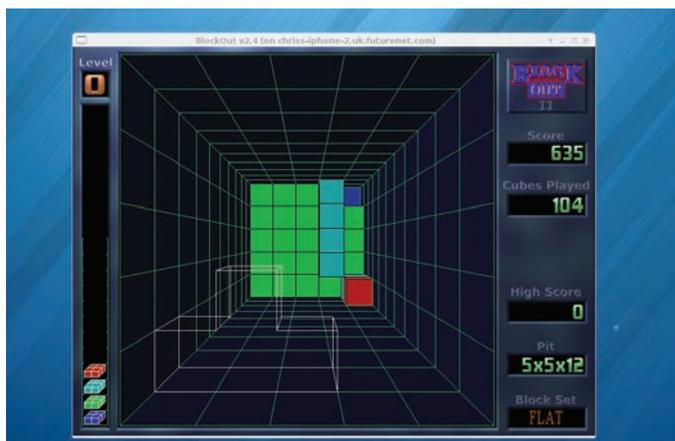
Frozen Bubble

Идея проста: палите одними цветными шариками в другие. Если цвета совпадают, шарики вылетают. Вы должны выбить все шарики до того, как они упадут на пол, тогда вы выигрываете. Однако часы тикают, а шарики все падают и падают, и вам остается только ждать голубого шарика.



Blockout II

Думаете, *Tetris* для вас простоват? А вот попробуйте 3D-версию. Вам не нужны специальные очки — только хорошее пространственное чутье и несколько часов. Нажимайте на клавиши ASDW, чтобы перемещать блоки, а затем клавиши со стрелками, чтобы опускать их.



Lincity

Похожа на *SimCity* до того, как DRM сделал ее неиграбельной. Здесь даже нет серверов, чтобы перегружаться. Более того, она настолько захватывающая, что мы потеряли Гэри почти на неделю, и за малым не сорвали срок публикации. Вперед — на свой страх и риск.



Emilia Pinball

Более или менее то, чего и стоит ожидать от игры в пинбол, но только с лапами (оценили?), металлическими шариками и всякой всячиной, которую нужно этими шариками выбивать. Вам дано два стола, чтобы попрактиковаться. Какой лучше? Конечно, с Туксом!



Pipe Night Dreams

Есть вход, выход и некая ядовито-зеленая субстанция между ними. Все, что вам надо сделать — соединить входящую трубку с выходной трубкой, пока вся эта воничая грязь не заполнила все вокруг. В чем подвох? А нельзя подбирать куски труб, чтобы их соединить.



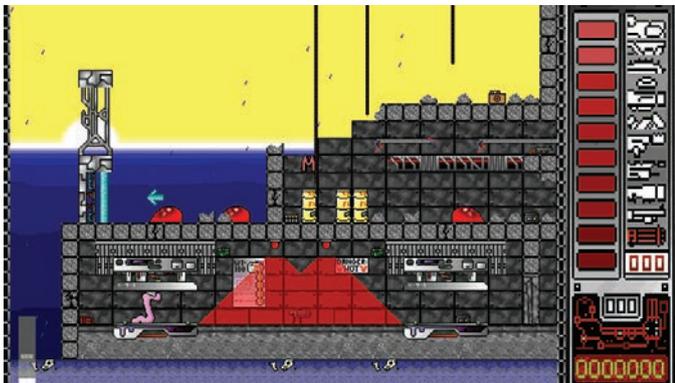
Pingus

Пингвины, в отличие от леммингов, не прыгают вниз со скал. Кстати, лемминги тоже не прыгают: это миф. Подозреваю, что эта игра не более реалистична, чем вдохновившая ее легенда, но тем большее удовольствие в нее играть. Давай, пингвинчик, давай!



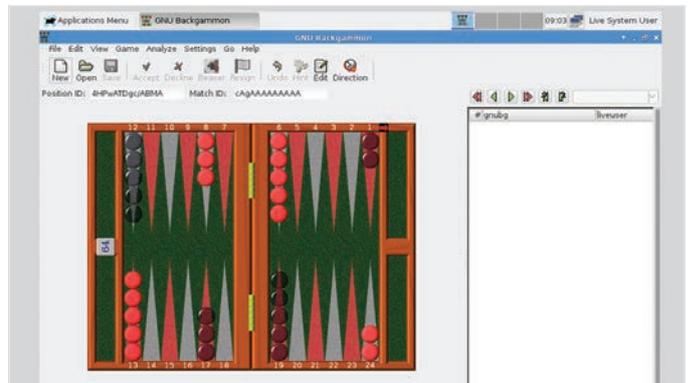
Worminator

Вас только что создали на восставшей фабрике, и ваша задача — сбегать, а потом найти и убить лидера повстанцев. Вы — это червяк с автоматом. *Worminator* — платформенная игра с ретро-графикой, где вы бегае (или ползете, извиваясь) среди стреляющих красных пузырей.



GNU Backgammon

Отличная игра вовсе не обязательно подразумевает зубодробительный экшен, который заставляет игрока выпрыгивать из штанов. Классическая настольная игра в нарды недаром существует вот уже 5000 лет, и реализации этой игры в Гну не подкачала.



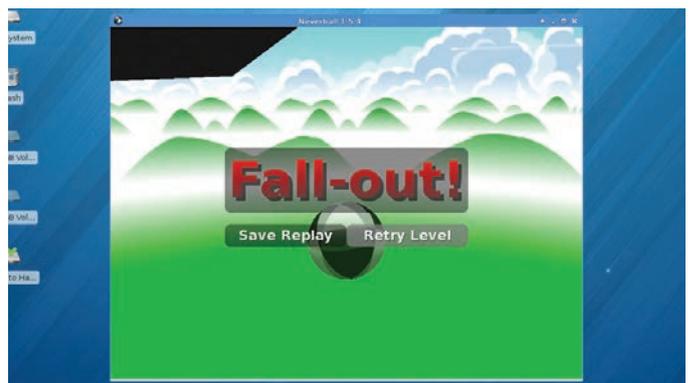
Xmoto

Вы управляете парнем на мотоцикле, тщательно контролируя равновесие и ускорение, чтобы провести его по предательскому внедорожному маршруту. Это сложнее, чем кажется, и иногда создается впечатление, что он сам жаждет угодить мордой в грязь.



Neverball

Раскачивайте мышью доску, управляйте мячом и собирайте монетки. Думаете, просто? А вы попробуйте. Эта небольшая игра чертовски сложна, и в то же время невероятно захватывает. Понадобятся сверхчеловеческие навыки управления мышью, чтобы гонять блуждающие мячи.



На диске

Пробуйте новую операционную систему уже сегодня!



В этом месяце мы пошли вразнос и львиную долю DVD отделили на игровой дистрибутив, который нами сделан, чтобы продемонстрировать с его помощью лучшие достижения среди игр с открытым кодом. Нельзя недооценивать вклад в создание этого дистрибутива команды Fedora и их превосходной системы kickstarts, которая позволяет без проблем модифицировать дистрибутив по своему желанию.

Однако не стоит думать, что в этом месяце у нас на повестке дня только развлечения и игры (вообще-то так и есть, только малость другим способом), и на диске вы найдете также и отличные дистрибутивы для вашей работы. У нас есть самый свежий OpenSUSE, которого мы так долго ждали; Kali, новый дистрибутив от создателей BackTrack; и Trisquel — дистрибутив, свободный как слово и одобренный GNU.

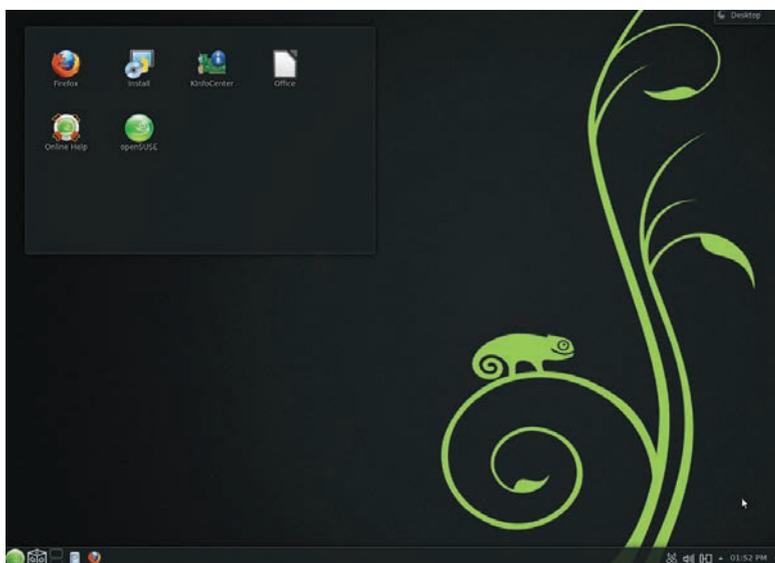
Кармический хамелеон

openSUSE

OpenSUSE относится к классике Linux, и это, вероятно, самый популярный из дистрибутивов с KDE по умолчанию. Эта версия идет с последним KDE (версия 4.10), известным как «самый быстрый из всех KDE». Мы не претендуем на то, что проверили абсолютно все версии KDE, но нам она показалась довольно быстрой. Подробнее о ней — в нашем обзоре на стр. 14.

Основная функция openSUSE — и объект равно любви и ненависти — YaST2. Это сборный пункт, где можно настроить большинство аспектов работы компьютера. Сторонники заявляют, что в нем есть большинство нужных вам функций в едином интерфейсе, и не надо возиться с файлами настройки или невнятными инструментами командной строки, а противники считают, что он уменьшает степень контроля. Но все дело в том, для чего вам компьютер. Если всего-то и надо отладить пару настроек, то YaST, вероятно, облегчит вам жизнь. А если вы мастер на все руки и знаете, где искать настройки, то обойдетесь и без него.

Нас также немало впечатлил процесс установки openSUSE одним щелчком — он позволяет устанавливать программы прямо в браузер. Его аналог из Ubuntu явно уступает в мощности.



➤ Зелень остается, правда, оттенок потемнее, чем обычно.

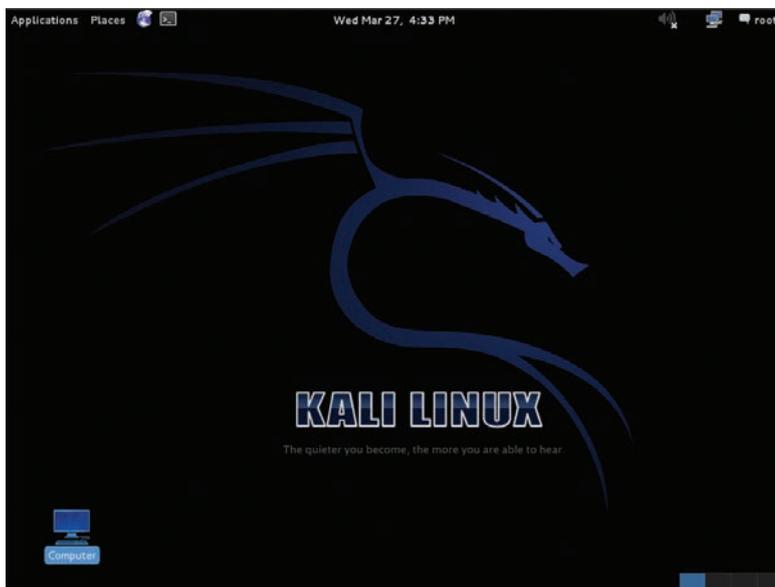
Чем тише, тем слышнее

Kali

BackTrack был дистрибутивом, особенно любимым как хакерами, так и теми, кто проводил тесты на возможность взлома — за то, что в него входили практически все инструменты безопасности, которыми стоит пользоваться в Linux. Однако как таковой, этот дистрибутив был не лишен проблем, и его разработчики решили, что пора с ним покончить.

Вместо него они создали Kali. На первый взгляд, он очень похож на BackTrack. Тот же логотип с выгнувшимся драконом, тот же интерфейс и похожие (хоть и обновленные) инструменты безопасности. Однако Kali гораздо ближе к полнофункциональному дистрибутиву, чем когда-то был BackTrack.

Если вы читали основную статью этого месяца и вам не терпится отточить свои навыки, загрузите Kali и ознакомьтесь с инструментами. Большинство из тех, которыми пользуются профессионалы в сфере безопасности и ее нарушители, здесь имеется, так что это — идеальный стартовый пункт.



➤ Kali сохраняет тот же стиль, что и BackTrack.

Вам не отнять у нас свободы

Trisquel

Большинство дистрибутивов включают несвободный код — обычно это драйверы устройств. За них вам не придется платить (они бесплатны), но у разработчиков нет доступа к исходному коду (то есть они не свободны). Для одних это неизбежное зло, поскольку драйверы все же позволяют вашему ком-

пьютеру работать немного лучше, а для других их использование означает пересечение невидимой черты.

Однако Trisquel не занимается подобной ерундой. Он свободен на 100 %, как свободны слово и открытый код. Вы можете получить доступ к открытому коду в каждой части системы и делать с ним все, что пожелаете.

Конечно, это означает, что некоторые игры будут работать не так гладко, как в других дистрибутивах, но разве это не мелочь по сравнению со свободой?

► Помимо свободы, нас очень впечатлила настройка рабочего стола Trisquel, которая происходит без всякой лишней суеты.



Столлмен о свободном цифровом сообществе

Stallman

Что хуже всего в Ричарде Столлмене [Richard Stallman]? То, что он только один, а значит, не может находиться сразу в нескольких местах, чтобы вести свою проповедь свободного ПО. Чтобы хоть чуть-чуть исправить эту ситуацию, мы включили видео его январского выступления в Университете Мичигана. Называется оно «Свободное цифровое сообщество». Мы не присуждаем призов тем, кто догадался, какую именно свободу он имеет в виду.

Речь длится час сорок пять минут — больше, чем некоторые художественные

фильмы — и поможет вам немного лучше понять этого великого человека.



► Отец-основатель GNU и обладатель лучшей бороды во FLOSS.

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.

Загрузка DVD

Если вы читаете журнал в цифровом виде, то можете загрузить DVD с нашего сайта (www.linuxformat.com/content/downloads) и записать его на чистый DVD.

Информацию о том, как это сделать в разных операционных системах, вы найдете здесь: <http://tinyurl.com/burndvdiso>.

На диске

Помимо этих замечательных дистрибутивов и игр, мы включили в состав диска весь код из руководств, все программы из *Hotricks* и все программы, упомянутые в журнале, рядом с которыми вы видите значок **На DVD**. 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 8 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=lx171.iso of=/dev/sdX
```

при необходимости включив путь к загруженному ISO, где X изменяется в соответствии с вашим устройством 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:



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** День дистрибьютеров 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/



LXF170
Май 2013

250 руб.

- » **Взлом Сети** Учимся парировать атаки
- » **Дистрибутивы для продвинутых** Характер нордический
- » **Samba 4** Докладываем: есть совместимость с Active Directory!
- » **Kazan** Рабочий стол как герой кинофильма

LXFDVD: PCLinuxOS, Porteus 2.0 и еще 3 дистрибутива, 10 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_170/
PDF-версия: shop.linuxformat.ru/elxf_170/

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



6 месяцев
1230 руб.

12 месяцев
2280 руб.

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: Pipe Night Dreams, Wotminator и множество других игр

Frozen Bubble » Blockout II » Lincity » Emilia Pinball » GNU Backgammon

Fedora 18 Live с рабочим столом XFCE

ИГРОВОЙ DVD LXF

А ТАКЖЕ: MP3rip 2.5.1, Terminator 0.96, 10 книг о Linux и многое другое

И ЕЩЕ: Kali 1.0 » Trisquel 6.0 LTS

32-разрядная сборка с рабочим столом KDE

openSUSE 12.3 Live



Июнь 2013
LXF DVD 171

LINUX
FORMAT

Июнь 2013
LXF DVD 171

LINUX
FORMAT

Содержание

Сторона 1

ДИСТРИБУТИВЫ

Kali 1.0 Дистрибутив на базе Debian с набором средств для тестирования на проникновение, 32-разрядная сборка (загрузка с LXFDVD)

openSUSE 12.3 Live 32-разрядная сборка с рабочим столом KDE (загрузка с LXFDVD)

Trisquel 6.0 LTS Полностью свободный дистрибутив на базе Ubuntu, 32-разрядная сборка (загрузка с LXFDVD)

Сторона 2

Игровой LXHFDVD Fedora 18 Live XFCE с сотней различных игр, 32-разрядная сборка (загрузка с LXHFDVD)

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 Раймонда [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 Кроа-Хартманом [Greg Kroah-Hartman]

System Administrators Guide Руководство по базовому администрированию Linux

GNU Tools Summary Руководство по работе в командной строке и обзор основных утилит GNU

Видеозапись лекции Ричарда Столлмена [Richard Stallman] «Свободное цифровое сообщество», прочитанной им в Университете Мичигана в январе 2013 года

НОТРИСКИ

Half Timer Ассистент выключения системы

I7z 9.27.12 Инструмент получения полного отчета о работе процессоров Intel Core i3, i5 и i7

Iriamb 0.1.1 Физическая игра

JSeSh 6.3.2 Редактор древнеегипетских иероглифов

Mosh 1.2.3 Мобильная оболочка, сохраняющая открытые соединения при смене сети

MP3split 2.5.1 Сплиттер аудиофайлов

OnetWalk 1.4 Любимая игра системных администраторов

The Silver Searcher 0.14 Программа поиска по исходному коду

Terminator 0.96 Усовершенствованный терминал

Tilet 0.9.0 Редактор 2D и псевдо-3D карт для игр

Torsocks 1.3 Ассистент Tor

Пожалуйста, перестаньте использовать именем Адного Диска озанкомьтесь с опубликованной в журнале инструкцией!

КОММЕНТАРИИ Присылайте ваши пожелания и предложения по электронной почте: info@linuxformat.ru

ДОВОЛННЫЕ ДИСКАМИ В маловероятном случае обнаружения дефектов на данном диске, обращайтесь, пожалуйста, по адресу disk@linuxformat.ru

Настоящий диск тщательно тестировался и проверялся на всех этапах производства, однако, как и в случае с любым новым ПО, мы рекомендуем вам использовать аналитический 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, предоставленных нашей программой или данных. Прежде чем устанавливать какое-либо ПО на компьютер, пожалуйста, скачайте программу для загрузки с нашего сайта: www.linuxformat.ru

Тираж издательства ООО «Уральский электронный завод», 620137, Россия, г. Екатеринбург, Студенческая ул., д. 9. Лицензия МПТР ВАР 77-15.

Создание установочных дисков при помощи cdcrcord

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):

```
Plextor= 0,3,0 12 16M
```

Первое слово в этой строке — метка; затем после адреса SCSI-устройства вы должны указать скорость и размер буфера. Теперь можете заменить SCSI-адрес в командной строке на выбранную вами метку. Все будет еще проще, если вы добавите следующее:

```
CDR_DEVICE=Plextor
```

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?

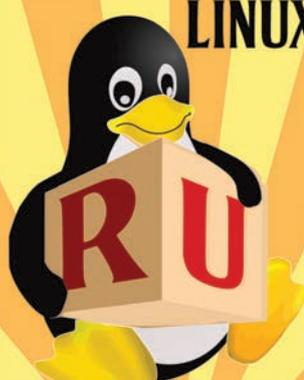
Вам не обязательно использовать Linux для записи компакт-диска. Все необходимые файлы уже включены в ISO-образ. Программы вроде *cdrecord* просто переносят данные на чистую матрицу. Если у вас нет устройства для записи дисков, можно найти того, у кого оно есть, и записать диск на его компьютере с Windows, 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).

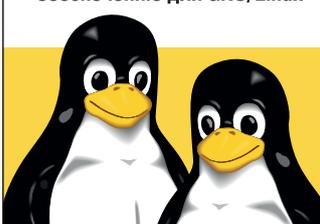
Некоторые дистрибутивы умеют монтировать образы дисков и выполнять сетевую установку или даже установку с раздела жесткого диска. Конкретные методы, конечно, зависят от дистрибутива. За дополнительной информацией обращайтесь на web-сайт разработчика дистрибутива.

**ФОРУМ
№1**
ДЛЯ ВСЕХ
ПОЛЬЗОВАТЕЛЕЙ
LINUX



LINUXFORUM.RU

LinSoft.Info
Путеводитель по программному
обеспечению для GNU/Linux



WWW.LINSOFT.INFO

Linux информационно-энциклопедия
по-русски

**Библиотека
книг, статей
и переводов
о Linux**

WWW.RUS-LINUX.NET

alubuntu.ru



**сообщество
пользователей
UBUNTU**

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/

**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

Аппаратно-программный комплекс

**DR.WEB
OFFICE SHIELD**

Комплексное решение задачи антивирусной
и антиспам-защиты для малых и средних предприятий

	88 679 руб.	
DR.WEB OFFICE SHIELD TWISTER <small>Расчитан на 250 ПК</small>		DR.WEB OFFICE SHIELD NEO <small>Расчитан на 50 ПК</small>

Dr.WEB®

Linux center
www.linuxcenter.ru

WWW.LINUXCENTER.RU/SHOP/ANTIVIR/DR_WEB/

UnixEducation Center
Россия, 190000, Санкт-Петербург
Черноморский переулок, дом 4
Тел.: + 7 (812) 611-1575

ORACLE **Symantec** **redhat**

PEARSON VUE AUTHORIZED TEST CENTER **PROMETRIC TEST CENTER** **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 7908

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

[Efrain Hernandez-Mendoza] efrain.hernandez-mendoza@futurenet.com

Выпускающий редактор Гэри Уокер [Gary Walker] gary.walker@futurenet.com
(до свидания, Гэри!)

ПОДГОТОВКА МАТЕРИАЛОВ

Джоно Бэкон [Jono Bacon], Нейл Ботвик [Neil Bothwick], Крис Браун [Chris Brown], Алекс Кокс [Alex Cox], Роб Добози [Rob Doboz], Бен Эверард [Ben Everard], Марко Фиоретти [Marco Fioretti], Джон Лэйн [John Lane], Джеймс Литтон [James Litton], Грэм Моррисон [Graham Morrison], Майк Сондерс [Mike Saunders], Маянк Шарма [Mayank Sharma], Шашанк Шарма [Shashank Sharma], Ник Вейч [Nick Veitch], Евгений Балдин, Артем Зорин, Михаил Остапкевич, Игорь Спиринов, Алексей Федорчук

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 лицензированы Future Publishing Ltd (Future plc group company). Все права зарегистрированы. Никакая часть данного журнала не может быть повторно опубликована без письменного разрешения издателя.

Все письма, независимо от способа отправки, считаются предназначенными для публикации, если иное не указано явно. Редакция оставляет за собой право корректировать присланные письма и другие материалы. Редакция Linux Format получает неисключительное право на публикацию и лицензирование всех присланных материалов, если не было оговорено иное. Linux Format стремится оставлять уведомление об авторских правах всюду, где это возможно. Свяжитесь с нами, если мы не упомянули вас как автора предложенных вами материалов, и мы постараемся исправить эту ошибку. Редакция Linux Format не несет ответственности за опечатки.

Ответственность за содержание статьи несет ее автор. Мнение авторов может не совпадать с мнением редакции.

Все присланные материалы могут быть помещены на диски — CD или DVD, поставляемые вместе с журналом, если не было оговорено иное.

Ограничение ответственности: используйте все советы на свой страх и риск. Ни при каких условиях редакция Linux Format не несет ответственности за повреждения или ущерб, нанесенные вашему компьютеру и периферии вследствие использования тех или иных советов.

Linux — зарегистрированный товарный знак Линуса Торвальдса [Linus Torvalds].

“GNU/Linux” заменяется на “Linux” в целях сокращения. Все остальные товарные знаки являются собственностью их законных владельцев. Весь код, опубликованный в журнале, лицензирован на условиях GPL v3. См. www.gnu.org/copyleft/gpl.html

За информацией о журналах, издаваемых Future plc group company, обращайтесь на сайт <http://www.futureplc.com>

В июльском номере

Что утратил Ubuntu?

Тех, кого боги хотят истребить, они награждают титулом «любимейший в мире дистрибутив».

Мы прорвемся сквозь завесу рекламных преувеличений и мифов и разберемся, что не так, ребята.

Дистрибутивы для сенсорных экранов

Будущее осязаемо, и мы отобрали пять самых «трогательных» дистрибутивов и посмотрели, какой лучше.

Trisquel внутри

Племя неукротимых галлов противостоит вползанию в Linux несвободного ПО. Отведаем их магическое зелье...

Родословная

Взгляните на *Gramps* — открытую и бесплатную программу, которая построит ваше фамильное древо.

Содержание будущих выпусков может меняться — вдруг нам придется спасаться от убунтофилов...



© Linux Format 2005

© Future Publishing Ltd 2005

BATH • LONDON • MILAN • NEW YORK • PARIS • SAN DIEGO • SAN FRANCISCO



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.

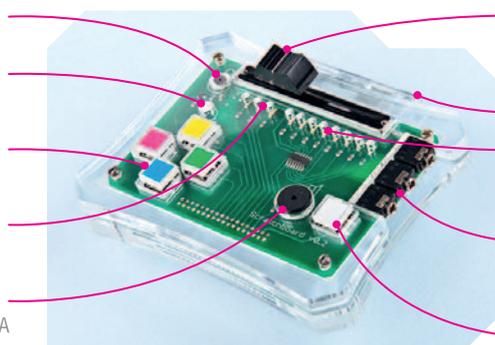
ДАТЧИК ЗВУКА*

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

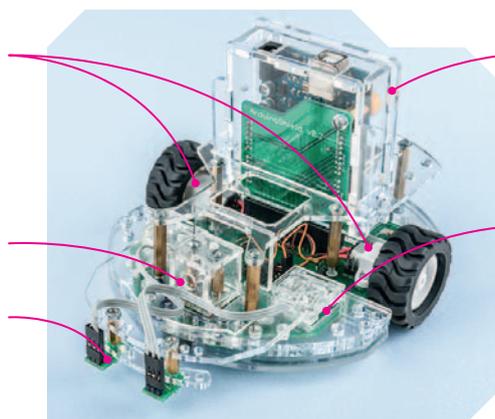
обеспечивает возможность:

- 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)