

LINUX MINT

Что в нем новенького?
Отличный Timeshift!

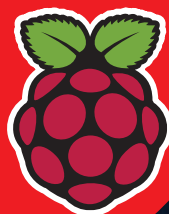
ЗВУЧНЫЙ СОВЕТ

Ищем лучшие открытые
инструменты аудио

ДАРИМ 8-ГБ DVD!

НАБОР ХАКЕРА

- + ПРОЕКТЫ ДЛЯ Pi
- ДОСТУП ИЗДАЛИ
- РУЛИМ LED & ЕЩЕ



LINUX FORMAT

Главное в мире Linux

ТАЙНЫ ХАКЕРОВ

Защитите свою систему, изучив
инструменты и трюки злоумышленников



64 страниц
статей
и учебников



Гвидо ван Россум
Отец-основатель Python
открыл, как это начиналось!

ПОНАПИШЕМ
ЭЛЕКТРОННЫХ КНИГ

ОСВОИМ BLUETOOTH
В LINUX

ГОВОРIT РАДИО ТУКСА
Принимаем радиолюбительские
станции в Linux

ВИДЕО С HANDBRAKE
Разберемся с лучшим
средством кодирования видео

СЕНТЯБРЬ 2018 LXF241



4 607130 825167

18009

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
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)

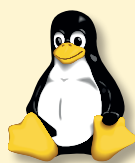
271-49-54

Санкт-Петербург
+7 (812)

309-06-86

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

Linux  center



LINUX FORMAT



» КТО МЫ

В этом номере мы намерены насладиться замечательным новым релизом дистрибутива Linux Mint 19; а что вы больше всего ненавидите в своем повседневном дистрибутиве и как бы вы это изменили?



Джонни Бидвелл

Ненависть — это слишком сильно. И Arch почти всегда мил. Но я ненавижу, когда Systemd тормозит выключение, заставляя ждать 90 секунд или жать Ctrl+Alt+Del семь раз за две секунды для принудительной перезагрузки. Кто такое сочиняет и почему ненавидит [Ред.: — Давешь ненависть!] меня?



Джон Найт

Фу, какой негатив. Я не ненавижу ничего в самом Mint. Однако система пакетов Debian бесила меня много лет. И сейчас у меня такие же проблемы, как в 2003 г., когда одна мелкая заминка сбрасывает всю систему! Давно пора придумать что-нибудь понадежнее.



Ник Пирс

Я все еще на ножах с Ubuntu 18.04 LTS, но, кроме перехода от Unity к Gnome, там нет ничего особо неприятного. Немного раздражает то, что с кнопкой обновления Software Centre действуешь методом проб и ошибок. Слава богу, что на свете есть `sudo apt-get upgrade`.



Александр Толстой

Пока я не обнаружил сервис Systemd, который устанавливает обновления программ тихо и без моего вмешательства, я ненавижу тиранию обновлений на моей машине Tumbleweed. Остальное выглядит довольно приятным!



Лес Паундер

На моем боевом ПК все еще стоит Ubuntu 16.04, и Unity все-таки мешает моему рабочему процессу. Я никогда не щелкаю по логотипу Ubuntu в поиске приложений — нажимаю клавишу Super и начинаю печатать... или просто добавляю еще один значок в лоток.



Валентин Синыцын

Зачем ненавидеть софт, который можно только обругать (а он и не слышит)? Уж лучше обрушить эмоции на оборудование: его хоть пнуть можно. Да, шутка старая, и поддержка «железа» в Linux стала много лучше, но я нет-нет да вспомню ее — скажем, когда открываю крышку ноутбука, а Wi-Fi не включился.

Тридцатилетняя кибервойна



В тематической статье этого номера, посвященной информационной безопасности, упоминается легендарный «Червь Морриса», тридцатилетие которого мы будем «отмечать» 2 ноября этого года. В тот день обычные люди, далекие от ИТ и ИБ, из выпуска новостей узнали, что всего одна нехитрая программа может вывести из строя целую сеть связи. Такое случилось впервые в истории человечества. Кстати, медико-биологическую терминологию (черви, вирусы, эпидемии) к вредоносному коду начали

применять именно тогда. Атаки бывали и раньше, но не становились достоянием гласности. О масштабном взломе 1986-го стало известно только через три года, после выхода книги Клиффорда Столла «Яйцо кукушки».

Тридцать лет назад вчерашний студент Столл пытался донести до ответственных чиновников, в погонах и без, важность проблемы и реальность угрозы. Ответ был практически одинаков: сынок, иди работай и не учи нас, как Родину защищать... Блестящее описание этих персонажей и бесед с ними занимает две трети книги. Сегодня об угрозах информационной безопасности и рисках кибервойны говорят все, в том числе главы ядерных сверхдержав с трибуны ООН.

Прошлую Тридцатилетнюю войну (1618–1648) начинали, говоря современным языком, «частные военные компании». Капитан на свои деньги вооружал роту и занимался воевать за какого-нибудь курфюрста. Завершили войну вполне регулярные армии со всеми сопутствующими атрибутами — единообразным вооружением и организацией, униформой, уставом и некоторым понятием о воинской дисциплине. Тридцать лет назад один аспирант «положил» весь тогдашний Интернет. Без злого умысла, по банальной косорукости при кодировании. Сегодня киберпространство признано полем боя, и многие государства официально объявили о существовании у них вполне регулярных подразделений для действий на нем. Такие вот невеселые итоги тридцатилетия...

Кирилл Степанов, главный редактор

info@linuxformat.ru

» КАК С НАМИ СВЯЗАТЬСЯ

Письма для публикации: letters@linuxformat.ru

Подписка и предыдущие номера: subscribe@linuxformat.ru

Техническая поддержка: answers@linuxformat.ru

Общие вопросы: info@linuxformat.ru

Проблемы с дисками: disks@linuxformat.ru

Вопросы распространения: sales@linuxformat.ru

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

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

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

Знаете всё о Linux?

ДА

НЕТ

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

Зарегистрируйтесь
на сайте
shop.linuxformat.ru



СОМНЕНИЯ

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



Версия для iPad
и iPhone доступна
в App Store

Выберите вид подписки

PDF-версия
на 6 месяцев

1485 ₽

PDF-версия
на 12 месяцев

2760 ₽

Печатная версия
на 6 месяцев

2430 ₽

Печатная версия
на 12 месяцев

4500 ₽

Выберите вид доставки

Оплатите

Читайте Linux Format!

Станьте Linux-гуру

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

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

Печатная версия Linux Format понравится читателям, которые:

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

Способы доставки

- Курьером по Москве и Санкт-Петербургу
- Курьером и в пункты выдачи **iml.ru** по всей России
- Почтой по всей России заказной или простой бандеролью
- Самовывоз из офиса «ГНУ/Линуксцентра» в Санкт-Петербурге

Способы оплаты

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



«ГНУ/Линуксцентр»

Санкт-Петербург,
пр. Медиков, 5, корп. 7
(метро «Петроградская»)

(812) 309-0686

www.linuxformat.ru

Содержание

ОБОЗРЫ

AryaLinux 1.0 11

Дистрибутив показывает, чего можно добиться, вдохновляясь советами *Linux From Scratch* — в таком подходе присутствуют и плюсы, и минусы.

Feren OS 2018.07 12

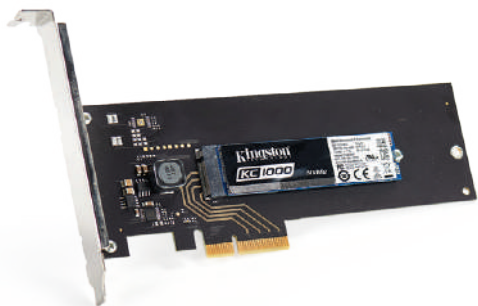
Заметно, что этот дистрибутив сделан с большой любовью и энтузиазмом. Он попал на наш диск, и, думается, мы им еще займемся...

Austrumi 3.8.3 13

Истинно патристический, этот дистрибутив, тем не менее, дружелюбен к гостям-иностранцам; главное — поздороваться с ним на родном его языке.

Kingston KC1000 960 ГБ 14

Настоящий мужчина выносивший независимо от габаритов. Этим и отличается SSD Kingston; но его менее впечатляющая скорость чтения расхолаживает.



Samsung 970 PRO 512 ГБ 15

Мы плохо относимся ко всем производителям SSD в мире, кроме Samsung: его новейшие технологии хранения разносят тесты в пух и прах.

Ruiner 16

Зловещая эстетика кибер-панковской антиутопии уж точно сработает, если вам нужна перемена обстановки — но кайф, к сожалению, не вечен.

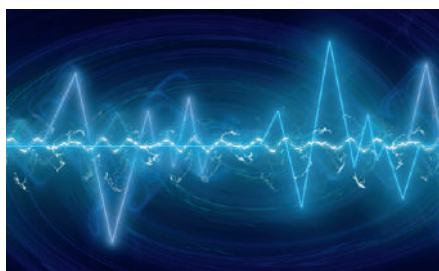


ТАЙНЫ ХАКЕРОВ

Приглашаем вас в таинственный мир хакеров и вооружим вас инструментами и советами, чтобы вы там не пропали. Стр. 26



СРАВНЕНИЕ



Музыкальные проигрыватели 20

По-прежнему любите слушать музыку старомодным способом, а храните на сервере под кроватью? Тогда вам нужен лучший музыкальный плеер с открытым исходным кодом.

ИНТЕРВЬЮ



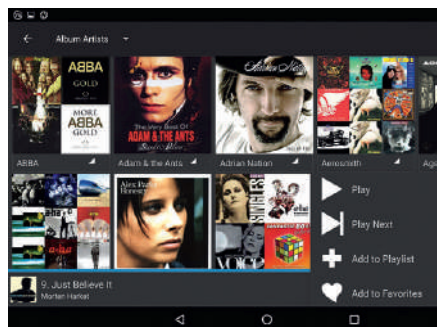
PyCon 2018 34

Наш репортаж из самого сердца мира Python, включая (нашу последнюю?) беседу с отцом-основателем Python **Гвидо ван Россумом** о его мыслях насчет будущего Python.

ПОЛЬЗОВАТЕЛЯМ RASPBERRY Pi

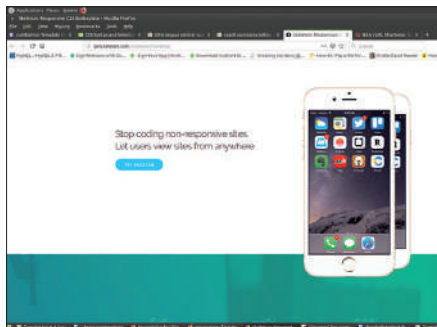


- Новости Pi** 87
Марсоход NASA делаем своими руками, писать документацию стало проще, и слышны звуки Му...
- Etcher 1.4.4** 88
Простой и быстрый способ записывать сразу несколько SD-карт очень выручит на Pi Jam.
- Pi-hole 3.3.1** 89
Ненавистная реклама засоряет ваши маршруты в Интернете, но средство от нее есть.
- GPIO на Bash** 90
Возьмите под свой контроль GPIO с помощью магии *Bash*... ну и нашей немножко.
- Музыка повсюду** 92
Описываем, как озвучить хоть весь дом, добавив контроллеры сенсорного экрана и Bluetooth.
- Есть контакт** 94
Компьютер-крошка ладит и с большими ПК (в т.ч. с Windows), и с устройствами на Android.



АКАДЕМИЯ КОДИНГА

- Боты против взлома** 74
Добрые боты помогут вам определить, что ваша система атаквалась злыми ботами (или людьми).
- Адаптивные сайты** 78
Раскрываем подробности создания сайтов, удобочитаемых на устройствах разных форм-факторов.
- Swift 4** 82
Разбираемся, что нового появилось в очередной версии этого языка программирования.



ПОСТОЯННЫЕ РУБРИКИ

- Новости** 6
AMD и Samsung радуют геймеров, Dell верна Ubuntu, вышли 4 HDD от Seagate, Libre 5 засиделся на старте, «БАРС Груп» занялась аналитикой, Альт 8 получила сертификат, а Athlon ожил.
- Вести мобильных ОС** 18
Все для 5G: чип для IoT и модем от Samsung, процессор от Qualcomm и сеть в США.
- Сравнение** 20
Насладиться музыкой помогут *Audacious*, *Banshee*, *Clementine*, *DeaDBeeF*, *Rhythmbox*.
- Интервью LXF** 34
На конференцию RuCon слетелись именитые эмеепоклонники: сам Гвидо ван Россум, Ива Джодловска, Николас Толлерви и Джесс Инграсселино.

- Рубрика сисадмина** 50
Д-р Валентин Синецын предпочитает Linux, но порой извлекает из загашника вкусняшки от BSD.
- Ответы** 96
32-разрядный UEFI, NextCloud, Nvidia, сервисы, Bluetooth, нанопиксели, *Samba* и *Grub* — почитим!
- HotPicks** 100
Лучшие в мире новинки свободного ПО: *Alacritty*, *Browsh*, *Defend*, *Dilay*, *Elisa*, *MyTetra*, *OpenRA*, *SFXR-Qt*, *Shotwell*, *SolarSystem*, *Stereophotoview*.
- Диск Linux Format** 106
- Предыдущие номера** 108
- Через месяц** 112



ЗАДАРОМ НА DVD

BunsenLabs
Feren OS 2018
Kali Linux Lite
...и не только
стр. 106

УЧЕБНИКИ

- ТЕРМИНАЛ: Wego** 54
Льет ли теплый дождь, падает ли снег — вы обо всем узнаете, не отходя от кассы компа.
- СИСТЕМА: Структура файлов** 56
Горек корень (root) ученья, но дерево файлов пышно зеленеет... ползаем по его ветвям со знанием дела.
- ВИДЕО: Handbrake** 58
Рассчитывая понадеяться потрясающих рипов видео, укрощаем дикого зверя — *Handbrake*. Мир, жди свежих роликов про котятков!
- СИСТЕМА: Bluetooth** 62
Bluetooth и Linux всегда были на ножах, но в их взаимоотношениях, похоже, намечился просвет.
- Е-КНИГИ: Sigil** 66
Кто у нас писатель? Этот инструмент не очень-то поддается кавалерийскому наскоку, зато уж всё будет сразу как полагается.
- РАДИОЛЮБИТЕЛЯМ: Web-SDR** 70
Казалось бы, звуки морзянки, потрескивающей в эфире, ушли в историю. Ан нет, радио перебралось в цифру!

УГЛУБЛЕННО...

- Свежая мята** 40
Извлекаем всё самое лучшее из Linux Mint 19 — мятный коктейль порадует и новичков, и упертых линуксоидов-ветеранов.



Новости

В ЭТОМ НОМЕРЕ: Геймерам и опять геймерам » Ноутбук с Ubuntu » Жесткие диски » Беды Linux-телефона » Для аналитики » Альт сертифицирована » Снова Athlon

МОНИТОРЫ

С эффектом присутствия

Samsung Electronics представила новую модель игрового монитора с изогнутым экраном.

В конце августа на выставке Gamescom 2018 (Кельн, Германия) Samsung представила новую модель игровых мониторов с изогнутым экраном CJG5 с диагональю 32" (C32JG50QQI) и 27" (C27JG50QQI), в которых реализованы ключевые геймерские технологии: высокое разрешение WQHD (2560×1440 точек), изогнутый экран, поддержка частоты обновления 144 Гц кадров и высокий уровень контрастности (3000:1). Благодаря безрамочному экрану, мониторы CJG5 обеспечивают игровой процесс с большим эффектом присутствия. Продукт уже доступен в России в фирменных салонах Samsung и магазинах партнеров; рекомендованная розничная цена составляет 22990 руб.

за версию с диагональю 27" и 27990 руб. за модель с диагональю 32". Благодаря изогнутой VA-панели и отсутствию рамок с трех сторон экрана, монитор CJG5 обеспечивает максимальное погружение в игровой процесс и помогает снизить нагрузку на глаза, чему также способствует высокая степень изогнутости (радиус изгиба 1800 мм).

Прочие технические характеристики: цвет — темно-серебристый; соотношение сторон — 16:9; яркость (типичная) — 300 кд/кв.м; угол обзора — 178° (по вертикали)/178° (по горизонтали); время отклика — 4 мс (GTG); входной сигнал — 2 порта HDMI (1×2,0; 1×1,4); 1 порт DP 1,2; разъем для подключения наушников.



Расширяя линейку своих игровых мониторов, Samsung представляет модель с изогнутым экраном, работающую с разрешением WQHD и частотой обновления 144 Гц.

НОВИНКИ AMD

Наполняя портфолио

Бюджетные процессоры AMD Ryzen 5 2500X и Ryzen 3 2300X рассчитаны на нетребовательных геймеров.

Оба представленных AMD новых процессора, Ryzen 5 2500X и Ryzen 3 2300X, основаны на последнем, втором поколении фирменной микроархитектуры компании Zen+ и выполнены по 12-нм техпроцессу. Ryzen 5 2500X отличается от своего предшественника, Ryzen 5 1500X, наличием четырех активных ядер на одном кристалле: у Ryzen 5 1500X было два четырехъядерных кристалла, причем два ядра на каждом кристалле были отключены. Таким образом, время межкристалловой задержки у Ryzen 5 2500X снижено; это ускоряет некоторые функции, но одновременно вдвое сокращает кэш CPU, что, в некоторых случаях, может снизить производительность. При этом для DIY («сделай это сам») новые Ryzen X недоступны, и будут

распространяться только через OEM. Ryzen 5 2500X уже устанавливается, например, в Acer Nitro N50-100, предлагаемом за \$900 в B&H Photo (за эту цену покупатель Nitro N50 также получает графику Radeon RX 580, 8 ГБ ОЗУ, HDD на 1 ТБ, мышь и клавиатуру в комплекте). Решение предоставлять новые CPU исключительно OEM-производителям идет вразрез с предыдущей политикой компании, когда AMD продавала свои чипы потребителям напрямую. AMD не комментирует смену курса; вероятно, речь идет о желании стимулировать у сегмента DIY спрос на APU Ryzen с графикой Vega. Поскольку CPU используются в дискретных графических сборках, OEM-производители выбирают более быстрые x16 PCIe lane в CPU Ryzen по сравнению с x8 lane в APU Ryzen.

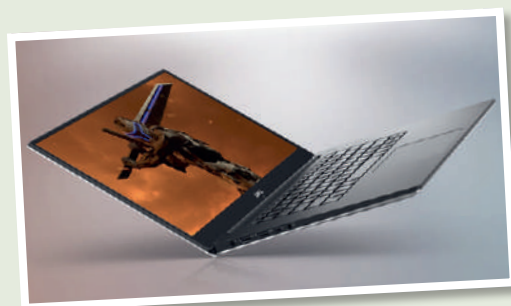
НОУТБУКИ ДЛЯ LINUX

Как всегда, от Dell

Мобильная рабочая станция Dell Precision 5530 Developer Edition поставляется с предустановленной Ubuntu.

Представитель компании Dell Бартон Джордж [Barton George], основатель и лидер проекта по созданию ориентированных на разработчиков ноутбуков (Project Sputnik), объявил о выпуске новейшей модели в новой линейке Dell Precision Developer на Ubuntu, представленной летом текущего года. Вслед за Developer Edition моделей Dell Precision 3530 Mobile Workstation, Dell Precision 7530 Mobile Workstation и Dell Precision 7730 Mobile Workstation, а также Dell XPS 13 Developer Edition с предустановленной Ubuntu 18.04 LTS, последовал выход Dell Precision 5530 Mobile Workstation Developer Edition, самой маленькой, тонкой и легкой 15-дюймовой мобильной рабочей станции с предустановленной Ubuntu 16.04 LTS (Xenial Xerus). Как и ее предшественники в новой линейке Dell Precision Developer, Dell Precision 5530 Mobile Workstation Developer Edition отличается улучшенной графикой, новейшими процессорами Intel Core и Xeon, великолепными дисплеем и дизайном, молниеносно быстрой памятью. Новинка также сертифицирована для Red Hat Enterprise Linux 7.5 со специализированными драйверами для профессиональных графических инструментов. Что касается спецификаций, то Dell Precision 5530 Mobile

Workstation работает на одном из процессоров 8-го поколения Intel i5 Quad Core, Intel i7 и i9 Six Core, или процессоре нового поколения Intel Xeon E-2176M Six Core, оснащается графикой следующего поколения до Nvidia Quadro P2000, до 32 ГБ памяти 2666 МГц, накопителем до 4 ТБ, ярким дисплеем 15,6". Dell Precision 5530 Mobile Workstation Developer Edition комплектуется четырьмя портами Thunderbolt 3.0 PCIe, портами USB 3.1 и HDMI 2.0, одним из беспроводных адаптеров Qualcomm 802.11ac Dual Band 2x2 или Intel Dual Band Wireless AC 9260 802.11ac MU-MIMO 2x2, реализована поддержка Bluetooth 5.0. Тонкий 14-дюймовый корпус доступен в цветах Platinum Silver или Brushed Onyx. Заказать Dell Precision 5530 Mobile Workstation Developer Edition можно непосредственно на сайте производителя. Цены начинаются от \$1331,36 за модель с Intel Core i5-8300H Quad Core 2,30 ГГц, графикой Intel HD, 8 ГБ DDR4-2666 МГц SDRAM, 2,5" 7 мм 500 ГБ 7200 RPM SATA HDD и Ubuntu 16.04 LTS (Xenial Xerus).



Поставляемая с предустановленной Ubuntu рабочая станция Dell Precision 5530 Developer Edition также сертифицирована для RHEL 7.5, и необходимые драйверы будут включены в дистрибутив.

НАКОПИТЕЛИ

Лучшие в своем классе

Seagate представила новую линейку жестких дисков.

Seagate выпустила четыре новых HDD емкостью до 14 ТБ для различных вариантов применения (у компании еще есть время, чтобы выполнить свое обещание — до конца 2018 г. представить модель на 16 ТБ). Для NAS предназначены модели IronWolf (£469) и IronWolf Pro (£529) с прошивкой AgileArray, улучшающей характеристики RAID и поддерживающей технологию dual-plane balance, датчики вращательной вибрации, улучшенное управление питанием и исправление ошибок. Все диски поставляются с гарантией на 3 года или 5 лет (модели Pro), а все модели Pro, кроме того, на 2 года обеспечены поддержкой Seagate Rescue data recovery. Для десктопов предлагается 3,5" HDD BarraCuda Pro (£509), который также можно использовать как систему хранения с непосредственным подключением (например, caddy). Его параметры — 7200 RPM, скорость передачи данных 250 МБит/с, кэш до 256 МБ с использованием технологии Seagate Multi-Tier Caching Technology

(как утверждает Seagate, это самый высокий показатель в отрасли), а также пятилетняя гарантия. Ориентированная на системы видеонаблюдения (включая видеорегистраторы) модель SkyHawk (£459) способна хранить до 9 тыс. часов видео в разрешении HD, поступающее от 64 камер. Используемая прошивка ImagePerfect сводит к минимуму вероятность пропуска кадров даже в режиме 24/7; гарантия для этой модели — 3 года. Exos X14 предназначен для использования в гипермасштабируемых дата-центрах. Как и другие HDD новой линейки, он выполнен в традиционном форм-факторе 3,5", и, как утверждает Seagate, предлагает лучшие в своем классе производительность и энергоэффективность. Как и следовало ожидать, это гелиевый накопитель. Seagate Secure encryption добавляет дополнительный уровень защиты данных без снижения производительности, т.е. шифрование FIPS-2 (государственный стандарт США) выполняется на легком и компактном диске.



Barracuda, один из четырех новых дисков на 14 ТБ.

LIBREM 5

Скоро сказка сказывается...

Проблемы с батареей на три месяца отодвинула начало производства телефона Purism Librem 5.

Производитель Linux-ноутбуков компания Purism вынуждена на три месяца задержать начало производства своего будущего Linux-телефона. Выпуск смартфона Librem 5, средства на который были собраны через кампанию краудфандинга, намечался на январь 2019 г., но теперь объявлено об отсрочке до апреля. Причиной назван «silicon bug» в SoC, вызывающий «экстремальный» разряд батареи: по словам представителя компании, время автономной работы смартфона составляет всего 1 час. На Librem 5 установлена PureOS — ОС на базе Debian, с поддержкой интерфейсов Gnome и KDE Plasma Mobile, а также Ubuntu Touch, разработанный Canonical как альтернатива Android и iOS. Телефон доступен для предзаказа по цене \$599; его ключевые особенности — функции обеспечения высокого уровня безопасности, в т. ч. возможность аппаратного отключения камеры, микрофона, Wi-Fi, Bluetooth и модема сотовой связи — т. е. всех источников, способных передавать данные о местоположении в режиме реального времени. Предполагалось оснастить Librem 5

SoC производства NXP; сначала речь шла о iMX 6, но его заменили новым iMX 8, с 64-разрядным CPU и поддержкой новых GPU, он-то и вызвал проблему с батареей. Librem 5 будет комплектоваться 5 дисплеем, 3 ГБ ОЗУ, 32 ГБ внутренней памяти, фронтальной и задней камерами, 3,5-мм разъемом для наушников, портом USB Type-C и теми самыми аппаратными выключателями. Трудности производства смартфона не-Android, с ОС, одобренной Free Software Foundation, в основном касаются прошивок и ПО для компонентов типа чипа сотовой связи, Wi-Fi, Bluetooth и т. п. — зачастую проприетарных, без исходных кодов и почти безальтернативных. СТО Николь Фербер [Nicole Faerber] сказала, что есть шансы решить проблему раньше, но в феврале производство вряд ли начнут из-за Нового года в Китае: «вся страна целый месяц не работает».



Компания Purism подчеркивает, что работа над дизайном смартфона Librem 5 еще не завершена.

ОТЕЧЕСТВЕННОЕ ПО

Для работы с большими данными

«БАРС Груп», дочерняя компания Национального центра автоматизации, повысила производительность собственной платформы бизнес-аналитики в тысячи раз.

Большие данные требуют не только новых подходов к построению комплекса технических средств, но и пересмотра состава ПО. ПО должно гарантировать быструю доставку информации в «озеро данных», их отказоустойчивое хранение с возможностью горизонтального масштабирования, а также минимум затрат времени на обработку сырых первичных фактов и доставку их до конечного потребителя в BI-инструментарии. Для этого в платформе бизнес-аналитики Alpha BI реализована поддержка хранения данных в файловой системе HDFS, разработаны инструменты доставки, обработки и извлечения показателей из Hadoop. Это позволяет строить хранилища данных с объемом, в миллионы раз большим создаваемых ранее на базе реляционных СУБД. У заказчиков всё чаще возникает задача потоковой обработки часто меняющихся данных и их оперативного анализа, поэтому в Alpha BI дополнительно реализована поддержка интeктору СУБД наряду с традиционными реляционными

СУБД и OLAP-серверами. Это увеличило производительность платформы в тысячи раз за счет использования кэша ОЗУ вместо обращения к дисковой памяти. Сейчас в Alpha BI реализованы основные инфраструктурные задачи. Продукт начал развивать инструменты продвинутого анализа по методам класса Data Mining, машинного обучения, нейронных сетей на основе библиотек, активно развиваемых сообществом OpenSource. По словам представителей компании, Alpha BI — ответ на вызовы рынка, который переходит к активному решению прикладных задач хранения и обработки больших данных. Всё чаще появляются новые проекты по построению высоконагруженных распределенных хранилищ данных с формированием real-time аналитики, требующие высокого уровня компетенций по работе с NoSQL, MapReduce, Hadoop и т. п.



«БАРС Груп» реализовала поддержку стека технологий для работы с Big Data в своей платформе Alpha BI 4-го поколения.

РОССИЙСКИЕ ОС

Несанкционированному доступу — нет!

ОС Альт 8 СП получила сертификат ФСТЭК.

Российская операционная система Альт 8 СП получила сертификат Федеральной службы по техническому и экспортному контролю. Программный продукт соответствует новым требованиям ФСТЭК России к операционным системам. Действие сертификата № 3866 распространяется до 10 августа 2023 г.

Сертификат ФСТЭК России удостоверяет, что ОС Альт 8 СП является операционной системой типа А со встроенными средствами защиты от несанкционированного доступа к информации, не содержащей сведения, составляющие государственную тайну. На основе ОС Альт 8 СП можно осуществлять построение государственных информационных систем (ГИС) и информационных систем персональных данных (ИСПДн) до 1-го класса (уровня) защищенности включительно.

Альт 8 СП разработана компанией «Базальт СПО» и производится АО «ИВК» для организаций, которым необходимы отечественные решения с высоким уровнем защиты. Операционная система позволяет организациям выполнить требования Приказов ФСТЭК №17 «Требования

НА ОСНОВЕ АЛТ МОЖНО ОСУЩЕСТВЛЯТЬ ПОСТРОЕНИЕ ГИС И ИСПДН ДО 1-ГО КЛАССА.

по защите информации, содержащейся в государственных информационных системах» от 11.02.2014 и №21 «Требования по защите персональных данных» от 18.02.2014. Альт 8 СП включена в Единый реестр российских программ для электронных вычислительных машин и баз данных.

БИТВЫ ЧИПМЕЙКЕРОВ

Позабытое слово Athlon...

AMD Athlon 200GE обеспечивает бюджетные ПК графикой Vega.

Рыночный дебют чипов Ryzen состоялся в начале 2017 года; у AMD, несомненно, получилось очень неплохое решение в сегменте процессоров среднего и высокого классов, кое-чем даже превосходящее предложения от Intel. Однако в бюджетном сегменте продукты AMD долгое время были привлекательны разве что для поклонников игр типа Team Red и пользователей, чьи приложения могли использовать лучшую поддержку процессорами AMD многопоточных задач. Ситуация может измениться после выхода AMD Athlon 200GE. 2 ядра, 4 потока, частота 3,2 ГГц — с такими параметрами возрожденный Athlon не конкурент ни семейству Ryzen, ни CPU Intel Core, но заинтересует пользователей бюджетных систем: его цена всего лишь \$55. Такой чип можно использовать, в частности, для создания минимедиацентра на ПК; параметр TDP у Athlon 200GE составляет всего 35 Вт — устройства на его основе будут отличаться низким энергопотреблением, чем не могут похвастаться, например, новые процессоры для ноутбуков Intel Whiskey Lake. Наряду с новыми потребительскими чипами Athlon, AMD продемонстрировала



В конкурентной борьбе за сектор бюджетных процессоров AMD возродила бренд Athlon.

и Pro-варианты своих продуктовых линеек, от Athlon Pro 200GE до Ryzen 7 Pro 2700X (последний поддерживает 8 ядер, 16 потоков, и тактовую частоту до 4,1 ГГц). В целом же, очевидно, что расширение AMD спектра своих процессоров не останется без внимания Intel, что, в свою очередь, должно стимулировать новые инновации в мире процессоров. **LXF**

Короткой строкой

» Intel сообщила о трех новых уязвимостях в своих процессорах Core и Xeon, связанных со спекулятивным исполнением побочного канала.
Источник: safe.cnews.ru

» Внесено в Реестр и опубликовано на сайте Минпромторга РФ заключение на производство в РФ линейки серверных системных двухпроцессорных материнских плат компании «Рикор Электроникс» на процессорах Intel Xeon E5.
Источник: minpromtorg.gov.ru

» В GIMP 2.10.6 реализована возможность вертикального размещения текста, добавлен фильтр Little Planet, в инструмент Measure добавлена возможность вертикального выравнивания.
Источник: <https://www.gimp.org>

» Почти за два года подготовлен релиз десктоп-окружения Trinity R14.0.5, продолжающего развитие кодовой базы KDE 3.5.x и Qt 3.
Источник: <http://www.opennet.ru>

» В "Astra Linux Common Edition" релиз Орел версия 2.12 включено ядро Linux 4.15, обновлен графический интерфейс, обновлены офисные приложения.
Источник: <http://astralinux.ru>

» Компания Valve представила проект Proton: решение на базе Wine, обеспечивающее возможность запуска под Linux всех Windows-игр из каталога Steam.
Источник: www.steamcommunity.com

» Oracle представила релиз ОС Solaris 11.3 SRU 35 с серией исправлений и улучшений, а также новый релиз Solaris 11.4.
Источник: <https://blogs.oracle.com>

» В рамках проекта "The Intel Safety Critical Project for Linux OS" создается высоконадежный дистрибутив Linux для применения в областях, критичных к возникновению сбоев.
Источник: <https://01.org>

» Более 20 комплексных ИТ-решений на основе операционных систем АЛБТ от «Базальт СПО» покажут на выставке «Импортозамещение-2018».
Источник: www.basealt.ru

Обзоры



Алексей Федорчук
Тэг <сарказм>
по умолчанию,
смайлики по вкусу.

» UBUTNU LTS: «ТОЧЕЧНЫЙ» РЕЛИЗ

Традиция что-нибудь сломать накануне выхода очередного «долгоиграющего» релиза Ubuntu имеет давнюю историю, начавшуюся с первой LTS-версии 6.06, которая потребовала коррекций в виде «точечного» релиза через несколько недель после анонса. Иногда, правда, к моменту официального релиза «поломки» чинятся. Так, ошибки в выводе кириллицы в консоли версии 16.04 LTS были исправлены буквально за несколько дней до ее выхода.

Однако релиз 18.04 не нарушил «долгоиграющей» традиции. О том, что почти накануне его выхода пакет *gksu* был не только исключен из поставки, но и удален из репозитория, уже говорилось (см. **LXF237**). Правда, это относится скорее к категории «не бага, но фиша»: майнтейнеры *Salopical*'а (вслед за Debian'овцами) сочли эту утилиту устаревшей. Но это доставило некоторые неудобства, ибо *gksu* фигурировал как зависимость ряда важных пакетов из PPA-репозитория. В частности, он создал сложности при подготовке очередной версии Cuntu — впрочем, преодолимые (см. **LXF238**).

Но уже после релиза 18.04 LTS в Ubuntu выявились и другие «поломки», на сей раз связанные с пакетами, зависящими от Perl'a. Правда, и их можно отнести к «фишам, а не багам». И распространялись они на небольшое количество пакетов, так что большинство применителей-Ubuntu'идов могли их просто не заметить, а майнтейнеры исправлением ошибок не занимались. Однако пакеты эти критичны для Cuntu и существенно ограничивали ее функциональность. Пока не вышел первый точечный LTS-релиз 18.04.01, в котором существовавшие «поломки» были, наконец, ликвидированы — к вящей радости моей и моих товарищей.

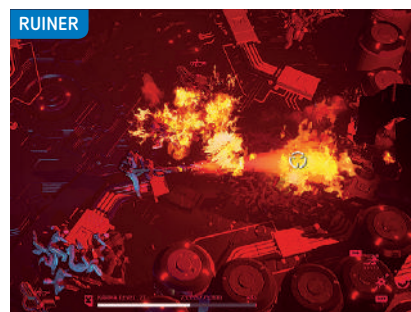
alv@posix.ru

Сегодня мы рассматриваем:

- | | | | |
|---|----|-------------------------------------|----|
| AryaLinux 1.0 | 11 | Kingston KC1000 960 ГБ | 14 |
| Действуя по инструкциям LFS, мы сможем все, решили разработчики, и соорудили дистрибутив, непохожий на других. На его основе легко и вам сделать свой. | | | |
| Feren OS 2018.07 | 12 | Samsung 970 PRO 512 ГБ | 15 |
| Вроде бы Mint — сама простота и дружелюбие, но Feren ухитрился стать еще проще. Вдобавок он сам себя обновляет. Правда, пока путается в показаниях насчет своей базы. | | | |
| Austrumi 3.8.3 | 13 | Ruiner | 16 |
| Попользоваться этим дистрибутивом вам удастся, если вы изучите латышский. Впрочем, вам хватит словарного запаса в 1 слово, чтобы переключиться на язык вашего выбора. | | | |
| Ваш герой, в молчании и в маске, странствует по трущобам в тени индустриального мегаполиса будущего, сокрушая врагов... но в какой-то момент вы уже и заскучете. | | | |



Выход Feren OS означает, что у Linux Mint появился серьезный соперник.



Этюд в багровых тонах: ваш герой отчаянно палит из огнемета.

» СРАВНЕНИЕ: МУЗЫКАЛЬНЫЕ ПЛЕЙЕРЫ СТР. 20



Пуск отовсюду слышится потоковое интернет-вещание, но не простаивать же вашей, любовно собираемой долгие годы, музыкальной коллекции.

AryaLinux 1.0 Gnome

Как автор личных инструментов, умеющий сам клепать дистрибутивы, **Шашанк Шарма** заинтригован дерзанием этого дистрибутива, вдохновленного *Linux From Scratch*.

ВКРАТЦЕ

Это первый релиз после смены схемы именования дистрибутива по типу год.месяц. Проект создает два варианта, один с Gnome, другой с рабочим столом Xfce, и является хорошей отправной точкой, если вы заинтересованы в создании своего собственного дистрибутива Linux, но еще не готовы иметь своим руководством только LFS.

Одним из самых больших изменений, которые претерпел проект со времени нашего обзора в **LXF217**, является изменение в соглашении о наименованиях. Еще одно серьезное изменение — введение версии с Gnome. Согласно примечаниям к релизу, проект планирует сделать Gnome настольной средой по умолчанию для будущих выпусков, хотя другой рабочий стол по-прежнему будет поддерживаться.

В последние годы наблюдается всплеск дистрибутивов с использованием специально разработанных инструментов — будь то среды рабочего стола или что-либо попроще, например, экран приветствия — и AryaLinux не является исключением.

В дистрибутиве имеется собственный установщик, который проводит пользователей через шестистапный процесс установки. Единственный недостаток установщика — отсутствие инструмента разбиения на разделы, и предполагается, что пользователи будут вручную отводить пространство для дистрибутива, притом по особой схеме. Придется создать три раздела — по одному для разделов подкачки, корневого и **/home**, и эта схема разбиения обсуждению не подлежит. Вы не можете, например, выбрать AryaLinux без отдельного раздела **/home**.

К счастью, в дистрибутиве не наложены ограничения на объем разделов подкачки и **/home**, и нам удалось успешно установить дистрибутив на корневой раздел размером 10 ГБ с 1,2 ГБ, зарезервированных для **/home**, и всего 512 МБ для подкачки. Даже с 4 ГБ оперативной памяти на двухъядерном компьютере AryaLinux работает превосходно. Это намного быстрее, чем у таких его аналогов, как Fedora и Linux Mint.

С придурью, но хороший

Кроме ограничительной схемы разбиения, дистрибутиве также порочит отсутствие графического инструмента управления ПО. Вместо этого в проекте используется самодельная *AryaLinux Packaging System* (ALPS), вызываемая командой `alps`. Написанный на Python, этот отличный инструмент умеет распознавать зависимости, а также устанавливать и удалять пакеты, но не так, как обычно делают его более популярные навороченные коллеги вроде *pacman*, *portage* или даже *dnf* и *apt*. Кроме того, в отличие от других менеджеров пакетов, которые полагаются на прекомпилированные бинарники, *alps* использует для сборки пакетов *tar*-архивы исходников, а это процесс более медленный, поскольку каждое приложение, инструмент или утилита загружается и компилируется отдельно.

С последней версией дистрибутив теперь также поддерживает приложения Flatpak, но, к сожалению, нет документации, описывающей, как устанавливать такие пакеты с помощью *alps*.

Фишка дистрибутива — его собственные скрипты, позволяющие вам создавать на базе AryaLinux свои ответвления.



Хотя AryaLinux — очень умелый настольный дистрибутив, у проекта есть особенности, которые противоречат большинству его собратьев.

Вдохновленный учениями руководств *Linux From Scratch* [LFS, «Linux с нуля»] и *Beyond Linux From Scratch*, дистрибутив поддерживает две разные модели сборки. Вы можете собрать свой продукт, взяв за основу среду Live-установщика, или использовать частичную сборку AryaLinux. Проект предусматривает три варианта частичной сборки: неграфическую базовую систему, базовую систему с X-сервером и базовую систему с X-сервером и Gnome. Выбор частичной сборки, которую вы используете, будет зависеть от цели вашего продукта.

Завоевав свою нишу, проект должен теперь выделять ресурсы на создание и улучшение своей документации. А пока AryaLinux зарекомендовал себя как полезный дистрибутив для настольных компьютеров, ориентированный на зрелых пользователей Linux, особенно намеренных создавать свои ответвления. **LXF**

ВЕРДИКТ

РАЗРАБОТЧИК: AryaLinux Team

САЙТ: www.aryalinux.org

ЛИЦЕНЗИЯ: GPL

ФУНКЦИОНАЛЬНОСТЬ	8/10
ПРОИЗВОДИТЕЛЬНОСТЬ	7/10
УДОБСТВО В РАБОТЕ	8/10
ДОКУМЕНТАЦИЯ	7/10

Несмотря на непривычную установку и управлению ПО, AryaLinux идеально подходит пользователям, желающим готовить свои собственные варианты.

» **Рейтинг 8/10**

Feren OS 2018.07

Псевдоплавающий дистрибутив на базе Linux Mint и Ubuntu, говорите? **Шашанк Шарма** придвигает стул, и на лице его написано: «расскажи еще»...

ВКРАТЦЕ

В дистрибутиве есть рабочий стол Cinnamon, и поставляется он с различными стандартными приложениями для большинства домашних пользователей. Помимо основного релиза, проект также выпускает облегченный вариант, называемый *Lite Edition*, для мало-мощных машин. Если вам удобно работать с самообновляющимися дистрибутивами, то *Manjaro Linux*, *Sabayon* и *Solus* — отличные и более зрелые альтернативы.

Feren OS начиналась как производная Mint, но с достаточным объемом переделок для занятия собственной ниши. Заявленная как идеальная отправная точка для новых пользователей, только что отбросивших свою проприетарную ОС, будь то Windows или Mac, последняя версия использует в качестве основной базы Ubuntu 18.04. Но вместо фиксированного цикла выпуска Feren OS выбрала самообновляемую модель, чтобы поддерживать вашу систему в актуальном состоянии.

Несмотря на молодость, дистрибутив в последние годы подвергнулся существенным изменениям, почти как и его ведущий разработчик, который проходит под псевдонимом The Feren Dev и должен был недавно приостановить проект, чтобы готовиться к экзаменам на получение аттестата за среднюю школу. Разработчик является постоянным блоггером и делится планами по разработке и целями дистрибутива на сайте своего проекта.

Хотя на сайте Feren OS отсутствует вики, доска форума и документация, там размещаются страницы и видеоуроки с обсуждением некоторых из его собственных эзотерических инструментов, таких как утилита *Settings Transfer* [перенос настроек], которую можно употребить для переноса ряда параметров с вашей установки Windows на Feren или же наоборот. К сожалению, нет общего индекса, где вся такая информация предоставлена для упрощения доступа.

Работа в процессе

Feren OS позаимствовал несколько инструментов из своего ставшего смешанным наследия: установка кодеков и функция моментального снимка наряду с загрузчиком *CinnVII Stark Menu* взяты из Linux Mint, а менеджер ПО и установщик — из Ubuntu. Прошлые его релизы весили более 3 ГБ и отличались широким спектром приложений, но этот последний поставляется как ISO-образ менее 2 ГБ для 32- и 64-разрядных машин и содержит все обычные инструменты, представляющие интерес для повседневного домашнего применения.

Рабочий стол украшен красочными обоями и часами, а апплет погоды на нижней панели настроен на отображение прогноза на следующие три дня. Экран приветствия не похож на встречающиеся в наши дни в большинстве настольных дистрибутивов, и дает быстрое представление о дистрибутивах и их функциях. Вам также предоставляется бесплатная ссылка для подключения к сообществу пользователей на Discord — это бесплатная платформа онлайн-чата.

Для всех ваших потребностей в программном обеспечении дистрибутив предоставляет *Software Manager*, а также *Synaptic*, однако можно настроить систему и на установку пакетов Snap и Flatpak. Фактически, по нажатию кнопки *Install Software* [Установить ПО] экран приветствия дает возможность устанавливать пакеты из всех трех источников.



Лучшее свойство этого дистрибутива — его четкий упор на сохранение простоты во всем. Но он также разработан, чтобы привлечь пользователей, не чуждых шика FOSS.

Themes — еще один отличный инструмент, применимый для настройки внешнего вида рабочего стола; он дает пользователям возможность загрузки из числа нескольких разных тем, а также внесения других изменений.

Хотя проект заявил о переходе на Ubuntu 18.04 в качестве базы, при загрузке Feren OS заставка по-прежнему сообщает о Linux Mint 19. На сайте также есть несколько примеров неясности проекта в его текущей базе и процессе разработки. Даже раздел *Introduction* [Введение] на экране приветствия ссылается на Linux Mint как базу дистрибутива, хотя проект перешел на Ubuntu уже давно.

Несмотря на небольшие разночтения и отсутствие документации, дистрибутив работает как задумано и сделан с большой любовью и энтузиазмом, что очевидно во всех аспектах, будь то переключатель окон Alt+Tab или менеджер входа в систему и многие специальные инструменты. Разработанная с целью обеспечения долгосрочного и простого в работе дистрибутива Linux, Feren OS определенно заслуживает внимания. **LXF**

ВЕРДИКТ

РАЗРАБОТЧИК: Feren OS Developer

САЙТ: <http://ferenos.weebly.com>

ЛИЦЕНЗИЯ: Разные

ФУНКЦИОНАЛЬНОСТЬ	9/10
ПРОИЗВОДИТЕЛЬНОСТЬ	8/10
УДОБСТВО В РАБОТЕ	8/10
ДОКУМЕНТАЦИЯ	5/10

Будь у дистрибутива больше разработчиков, он мог бы затмить другие дружелюбные к новичкам проекты, типа Zorin OS.

» **Рейтинг 8/10**

Austrumi 3.8.3

Маленький и проворный — это **Шашанк Шарма** не про себя говорит, а про дистрибутив Austrumi, который ему видится именно таким...

ВКРАТЦЕ

Дистрибутив производный от Slackware, способен полностью запускаться из памяти. Несмотря на объем менее 350 МБ, оснащен массой полезных повседневных приложений, таких как браузер, клиент чата, аудиоплеер, офисный пакет и даже некоторые игры. См. также: Porteus, тоже производный от Slackware, или попробуйте AntiX на базе Debian.

Austrumi [что по-латышски значит «восток»] — устанавливаемый Live-дистрибутив на базе Slackware. По умолчанию он работает прямо из ОЗУ, вытолкнув CD. Вторая опция — без извлечения CD. В первом варианте вы можете вытащить CD-привод или USB-накопитель, как только попадете на графический рабочий стол, поскольку Austrumi будет полностью работать из ОЗУ. И для оптимальной производительности этого ОЗУ требуется всего около 100 МБ. К сожалению, ссылка на скачивание выдает ошибку: вместо этого используйте ссылку на torrent из Linuxtracker.

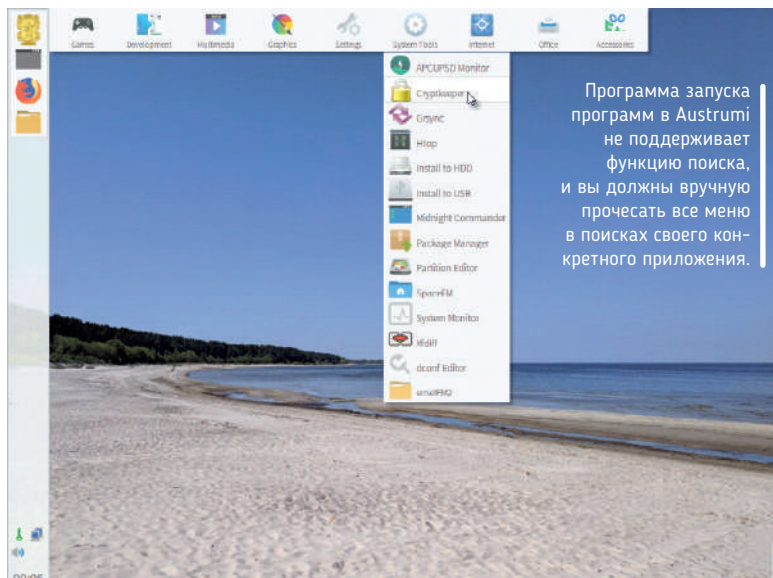
В отличие от большинства других дистрибутивов Linux, Austrumi не требует настройки. Он по умолчанию использует латышский язык и почти мгновенно загружается прямо в рабочий стол. Затем вам надо прочесть рабочий стол в поисках переключателя на английский или любой другой из поддерживаемых языков. Чтобы переключить язык, нажмите кнопку в верхнем левом углу экрана и перейдите к пятой кнопке на вертикальной полосе с изображением значка ключа и шестеренки. Затем нажмите последнюю кнопку на этой полосе, с подписью Volūdis [Языки], и выберите нужный язык в открывшемся окне. При переключении языка Austrumi автоматически выйдет из системы на рабочем столе, и надо будет войти снова. Пароль и для обычного пользователя, и для суперпользователя-root — “austrumi”.

Освежающе непохожий

Дистрибутив загружается в рабочий стол с угловой панелью. По вертикали слева содержится только несколько кнопок. За исключением загрузчика вверху и значков для запуска Firefox, терминала и менеджера файлов, есть только две другие кнопки: регулятор громкости и сетевые настройки. Обе расположены в нижней части экрана рядом с часами.

Верхняя кнопка на панели — для запуска программ. При нажатии на нее отображается горизонтальная полоска кнопок, например, Games [Игры], Development [Разработка], Multimedia [Мультимедиа], Settings [Параметры], Office [Офис] и System Tools [Системные утилиты]. При нажатии каждой кнопки отображается отсортированный в алфавитном порядке вертикальный список приложений, инструментов и утилит. Инструмент Languages мы уже использовали в разделе Settings для перехода на английский язык. Будучи основан на Slackware, дистрибутив также поставляется с менеджером пакетов Gslapt, но если вы хотите установить дополнительные приложения, надо будет добавить репозитории через Preferences. И, несмотря на отсутствие жалоб при установке таких известных пакетов, как Kmahjongg и VLC, оба не заработали в Austrumi из-за недостающих зависимостей.

Хотя в дистрибутиве используется оконный менеджер FVWM, представлен также ряд опций настройки. Помимо автоматического изменения обоев рабочего стола, Austrumi также предлагает несколько различных тем, таких как R-панель,



U-гаджеты и Windows. Вы можете получить к ним доступ, щелкнув правой кнопкой мыши в любом месте рабочего стола и переместившись в раздел Themes. Это еще одна область, где Austrumi отличается от своих коллег. Другие дистрибутивы редко предоставляют инструменты защиты вашей конфиденциальности или данных, а Austrumi позаботится о тех и других, благодаря Cryptkeeper, для создания зашифрованных папок, и утилите резервного копирования Grsync.

Для получения списка всех установленных пакетов надо запустить Gslapt и переключить фильтр на Installed packages. Быстрый просмотр списка даст вам представление обо всех бесчисленных пакетах, которые составляют Austrumi.

Благодаря своей поддержке графических карт Intel и Nvidia, Austrumi является хорошо собранным дистрибутивом и настоятельно рекомендуется не только хозяевам маломощных машин, но и тем, кто ищет альтернативу своей текущей установке. **LXF**

ВЕРДИКТ

РАЗРАБОТЧИК: Andrejs Meinerts

САЙТ: <http://cyti.latgola.lv/ruuni>

ЛИЦЕНЗИЯ: GPL

ФУНКЦИОНАЛЬНОСТЬ	9/10
ПРОИЗВОДИТЕЛЬНОСТЬ	10/10
УДОБСТВО В РАБОТЕ	8/10
ДОКУМЕНТАЦИЯ	Нет

Только отсутствие документации мешает нам считать этот удивительно быстрый и позитивный дистрибутив почти идеальным.

» **Рейтинг 8/10**

Kingston KC1000 960 ГБ

Kingston «играет в долгую» со своим диском NVMe, смекнул **Джереми Лэйрд**.

СПЕЦИФИКАЦИИ

- **Емкость** от 240 до 960 ГБ
- **Шина** PCIe 3.0×4 NVMe
- **Интерфейс** M.2
- **Контроллер** Phison PS5007-E7
- **Тип** Toshiba 15 нм плоская TLC
- **Последовательные чтение/запись** 2,7/1,6 ГБ/с
- **IOPS чтение/запись** 190/165 К
- **Ресурс** 1 ПБ
- **Гарантия** 5 лет

Вспомним июль 2013 г. Именно тогда вышел первый SSD с поддержкой этого крутого нового протокола управления NVMe. Четыре года спустя Kingston наконец появился на тусовке. Было ли это появление для KC1000 лишь «пентовым» опозданием? Или Kingston пропустил всё самое интересное?

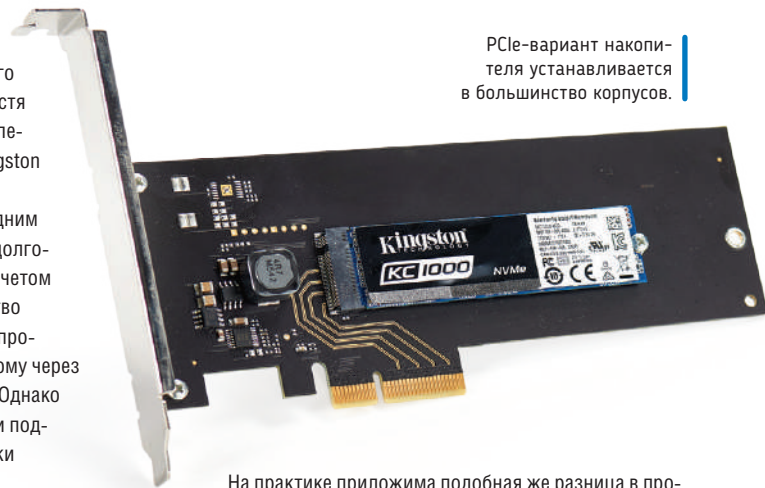
Ответ зависит от того, что именно вы ищете в SSD. Одним из главных коммерческих аргументов KC1000 является долговечность, а не быстродействие. Вроде и удивительно, с учетом того, что производительность — очевидное преимущество перехода от диска SATA старой школы на базе древнего протокола управления AHCI к накопителю M.2, подключаемому через PCIe через оптимизированный под SSD протокол NVMe. Однако KC1000 не только предлагает пятилетнюю гарантию, но и подкрепляет это двумя миллионами часов средней наработки на отказ (MTBF).

Займемся писаниной

Более того, Kingston оценивает этот 960-ГБ KC1000 как способный на 1 ПБ (1000 ТБ) циклов перезаписи. Это уже гораздо ближе к 1,2 ПБ лидера рынка, 1-ТБ Samsung 970 Pro, чем у более дешевого 1-ТБ 970 Evo с его 600 ТБ. Для справки, наш образец шел как часть пакета поставки HNNL, каковая буквенная абракадабра означает Half Height Half Length и указывает на входящую в комплект плату PCIe-адаптера указанного форм-фактора (половинная высота и длина), что позволяет использовать накопитель с материнскими платами без слота M.2. Вариант без адаптера обойдется вам чуть дешевле.

Подпирает всё это контроллер Phison PS5007-E7, плюс флеш-память MLC NAND по 15-нм технологии Toshiba — плоской, а не 3D. На момент выпуска в 2016 г. предполагалось, что Phison PS5007-E7 — в числе быстрых контроллеров NVMe. Сейчас он уже не такой конкурентоспособный. Данная 960-ГБ модель привязана к показателям 2,7 ГБ/с для скорости последовательного чтения и 1,6 ГБ/с — для записи. Производительность на 4К-блоках при произвольном доступе, опять же, далеко не самая лучшая. При 190К и 165К для чтения и записи, соответственно, он не побеспокоит, к примеру, такие продукты, как ТБ 970 Pro от Samsung с его «500К IOPS».

PCIe-вариант накопителя устанавливается в большинство корпусов.



На практике приложима подобная же разница в производительности. Пиковые значения пропускной способности в тестах неоднородны. С пиковым последовательным чтением чуть более 1,5 ГБ/с (для несжатых данных) это несколько меньше заявленного Kingston. Аналогично, результаты теста 4К с произвольным доступом — 42 МБ/с для чтения и 125 МБ/с для записи — показывают значительное отставание не только от диска-лидера Samsung 970 Pro, но и от более дешевого терабайтника 970 Evo.

При всём этом, в реальных приложениях прямой дефицит производительности KC1000, бесспорно, наиболее очевиден. Оба диска Samsung примерно вдвое быстрее в нашем тесте внутреннего копирования файлов. Аналогично, во время настройки перед тестами мы обнаружили, что последовательная пропускная способность KC1000 при последовательном копировании сотен ГБ может упасть аж до 200 МБ/с.

Это означает, что с KC1000 не всё так однозначно. Он легко уделает диск SATA, но по сравнению с последними твердотельными накопителями с PCIe выглядит довольно заурядным. Противовес этому — пятилетняя гарантия и обещание экстремальной долговечности. Если надежность находится на вершине вашего списка приоритетов, можете побрататься с KC1000. **LXF**

» СРАВНЕНИЕ ПРОИЗВОДИТЕЛЬНОСТИ

Накопитель	Kingston KC1000 960 ГБ	Samsung 970 Evo 1 ТБ
30-ГБ внутреннее копирование (сек)	62	31
Последовательное чтение (МБ/с)	1561	2911
Последовательная запись (МБ/с)	1507	2495
4К чтение (МБ/с)	42	68
4К запись (МБ/с)	125	171

ВЕРДИКТ

РАЗРАБОТЧИК: Kingston

САЙТ: www.kingston.com/en/ssd

ЦЕНА: £374 — 960 ГБ (£114 — 240 ГБ, £200 — 480 ГБ)

ФУНКЦИОНАЛЬНОСТЬ	9/10
ПРОИЗВОДИТЕЛЬНОСТЬ	6/10
УДОБСТВО В РАБОТЕ	9/10
ОПРАВДААННОСТЬ ЦЕНЫ	6/10

Отличная пятилетняя гарантия и высокий уровень надежности при не самой высокой производительности и цене.

» **Рейтинг 7/10**

Samsung 970 Pro 512 ГБ

Джереми Лэйрд тестирует SSD, которому нужны бы полосы побыстрее.

СПЕЦИФИКАЦИИ

- **Емкость** 512 ГБ
- **Шина** PCIe 3.0×4 NVMe
- **Интерфейс** M.2
- **Логика** Samsung Phoenix
- **Тип** 64-слойный 3D MLC V-NAND
- **Последовательные чтение/запись** 3,5/2,3 ГБ/с
- **IOPS чтение/запись** 370/500K
- **Ресурс** 600 ТБ
- **Гарантия** 5 лет+

Диск 970 Pro является частью третьего поколения высококачественных твердотельных накопителей NVMe от Samsung. Его предыдущие успехи были примерно на пределе потребительских SSD-накопителей. Итак, если на третий раз действительно везет, и 970 Pro еще лучше... ну, Samsung просто устрашает.

Прошло более 18 месяцев с момента дебюта Samsung 960 Pro (см. **LXF207** с нашим обзором 950 Pro). В быстрорастущем мире твердотельных хранилищ это вечность. За этот период Intel перезагрузила ожидания своей революционной технологией 3D Xpoint. Между тем, еще несколько производителей SSD тоже повысили ставки в игре. Итак, что же выкладывает 970 Pro на стол с SSD формата M.2? Неизбежно присутствие нового чипсета контроллера, Samsung Phoenix. Мы уже видели его в 970 Evo (см. **LXF238**), более дешевом брате 970 Pro. Столь же неизбежно то, что о Phoenix известно мало. Как и предыдущие контроллеры NVMe от Samsung, Phoenix использует четырехъядерный интерфейс PCIe 3.0 с восемью каналами. Он также имеет пять ядер процессора, таких как исходящий Polaris. На этом и всё.

Контроль производительности

Samsung никогда не сообщает детали, когда речь заходит о спецификациях его SSD-контроллеров. Однако Phoenix используется также и на промышленных SSD-устройствах Samsung за бешеные деньги, так что это, безусловно, лучшая технология контроллера компании. Мы также знаем, что 970 Pro использует новую 64-слойную память 3D MLC V-NAND от Samsung. Плотность данных этого нового поколения флэш-памяти NAND такова, что его 512-ГБ TLC-матрицы вмещают 1 ТБ в одном пакете BGA. Это делает большие емкости проще и дешевле, хотя эта модель Pro и остается предложением по премиум-стоимости.

Тем не менее, за эту приплату Samsung обеспечивает значительно большую выносливость жесткого диска. Новый 970 Pro намного опережает в этом отношении и своего предшественника, и своего собрата 970 Evo. Старое устройство 960 Pro 500 ГБ было рассчитано на 400 ТБ записей, а 970 Evo — на 300 ТБ, тогда как 970 Pro 512 ГБ рассчитан на полных 600 ТБ.



Последний SSD от Samsung отнюдь не тормоз: он обеспечивает скорость копирования реальных данных 1 ГБ в секунду.

При всём сказанном, формально прогресс от 960 Pro до 970 Pro значителен, но не всегда разителен. Максимум скорости последовательного чтения всё еще привязан к 3,5 ГБ/с, что, вероятно, отражает физический предел 4-полосного интерфейса PCIe. Скорость последовательной записи для этой модели 512 ГБ увеличилась с 2,1 ГБ/с до 2,3 ГБ/с. Что касается производительности 4K, она составляет от 330K IOPS для чтения до 370K IOPS. Запись возросла с 330K до 500K. А что показали тесты? Согласно нашим результатам, заявка на максимум пропускной способности в значительной степени зависит от денег. В отличие от более дешевых накопителей, таких как Kingston KC1000 на стр. 14, 970 Pro поддерживает производительность, близкую к заявленной максимальной пропускной способности, независимо от сжатия. Еще более впечатляет, бесспорно, производительность 970 Pro на реальных данных. Он буквально порвал наш внутренний тест копирования файлов, завершив его всего за 29 секунд.

Действительно, во время нашей предварительной тестовой настройки, которая включает заполнение диска перед его полной очисткой, скорость копирования 970 Pro составила около 1 ГБ/с. Очень немногие SSD могут хоть как-то приблизиться к настолько устойчивой производительности на сотнях и сотнях гигабайт.

Вполне вероятно, что 970 Pro остается одним из лучших, если не самым лучшим потребительским SSD. Шаг вперед на сей раз немного скромнее, чем раньше, но это отражает лишь ограничения технологии памяти NAND, подключенной через интерфейс PCIe 3.0. Пока что 970 Pro настолько хорош, насколько вообще способны быть SSD-накопители, хотя это и стоит дорого. **LXF**

» СРАВНЕНИЕ ПРОИЗВОДИТЕЛЬНОСТИ

Жесткий диск	Samsung 970 Pro 512 ГБ	Intel SSD DC P4510 1 ТБ
30-ГБ внутреннее копирование (сек)	29	32
Последовательное чтение (МБ/с)	3112	1431
Последовательная запись (МБ/с)	2338	2032
4K чтение (МБ/с)	66	35
4K запись (МБ/с)	231	207

ВЕРДИКТ

РАЗРАБОТЧИК: Samsung

САЙТ: www.samsung.com/ssd

ЦЕНА: £204 — 512 ГБ (£385 — 1 ТБ)

ФУНКЦИОНАЛЬНОСТЬ	9/10
ПРОИЗВОДИТЕЛЬНОСТЬ	9/10
УДОБСТВО В РАБОТЕ	9/10
ОПРАВДАНОСТЬ ЦЕНЫ	7/10

Производительность в общем отличная, но цена при такой емкости кусается, а преимущества производительности вы можете и не заметить.

» Рейтинг 9/10

Ruiner

Завидя Башни *Linux Format*, **Омри Петитт** предпочитает сбежать в безжалостное, сверх-жестокое, мрачное будущее киберпанков. А кто нет?

СПЕЦИФИКАЦИИ

- **ОС** Ubuntu 16.04/18.04, Linux Mint 18.3, Debian 9, Fedora 27
- **Процессор** Intel Core i5-760, AMD Athlon II X4 645 (желательно 64-битный)
- **ОЗУ** 4 ГБ
- **Графика** GeForce GTX 460, Radeon HD 6850, 1 ГБ VRAM
- **На диске** 20 ГБ

Ruiner — великолепный, чувственный пир, вдохновленный работами расцвета киберпанков в 1980-х гг., где главный герой, в молчании и в маске, странствует по ночным дорогам и индустриальным джунглям злое-технологичного 2091 г. Внизу в игре-боевике грохочет острейшая битва, привычность ее дизайна отмечается постоянным стремлением просто стоять на месте и впитывать всё.

Вы содрогнетесь от жестокой мощи эстетики *Ruiner*. Она бросится в глаза в момент щелчка по значку New Game: огромные жирные буквы, раскиданные по монитору ярко-красного цвета, китайские иероглифы и предупреждающие полосы, разбросанные по перевитым проводами тротуарным дорожкам, стробирующие вспышки света при любом обновлении цели и достаточно хроматических аберраций, чтобы *Far Cry 3: Blood Dragon* показала сентиментальным натюрмортом. К счастью, мигрень, которой мы боялись, не случилась, но мы бы не рекомендовали *Ruiner* для расслабления на сон грядущий.

Цветовая палитра *Ruiner* проникает глубоко в спектр киберпанков, с неоновыми основными цветами и перенасыщенной искусственностью. И враги, и типа свои одеты по грубой футуристической моде 1980-х гг.: шкеты с «ирокезами» и в грязных майках-алкоголичках, головорезы в прямоугольных солнечных очках с каменными лицами и злое-ухмылками, прямо как в манге Кацухиро Отомо [Katsuhiro Otomo]. Этот вид намертво закреплен в киберпанковом аниме и кино и придет полям битвы клаустрофобный переизбыток машинного и электронного беспорядка. Но, как и в большинстве работ этого жанра, именно город играет самую важную роль в задании тона. Для *Ruiner* это Ренгкок [Rengkok].

Ренгкок — то место, где вы ощутите самую сильную связь с *Ruiner*. Это не крупный хаб или многоквартирный муравейник,



но вы всё равно будете странствовать по его улицам. Его узкие переулки и стены, испещренные граффити, создают ощущение самой низкопробной трущобы, гниющей в тени небоскребов мегаполисов, хотя ее скудная масштабность разочаровывает: всё, что вы получаете — несколько открытых блоков и пара крыш.

То, что мы исследовали, восхитило нас изящными деталями и атмосферой отчасти благодаря мягкому звучанию и нежным напевам *Island Door* от Сусуму Хирасава [Susumu Hirasawa], еще один признак влияния аниме. И хотя вы часто покидаете город, всякий раз, возвращаясь, вы находите мало перемен хода истории.

Обитатели Рэнгкока добавляют индивидуальности, в отличие от молчания главного героя, которое осложняет, когда вы пытаетесь найти мотивацию любых их действий, кроме неубедительного мотива «брат в беде». После очередного диалога типа «кивнуть» или «пожать плечами» вы утратите надежду проникнуть в мысли мужчины, одетого в странную маску.

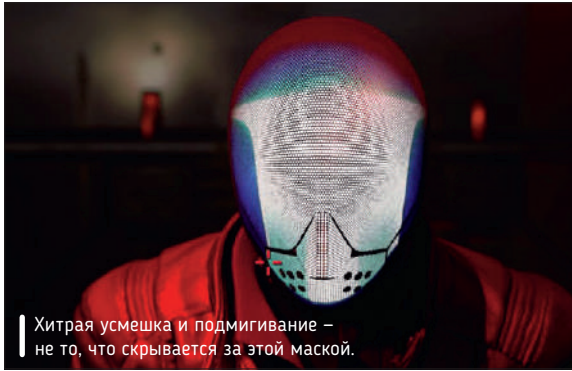
Можно ощутить некоторую человечность зазеркалья, общаясь с людьми в Рэнгкоке. Они изображают этот часто забываемый фрагмент киберпанка, который имеет дело с отверженными будущего, где технология обожествляется, проклинаема и обладает жуткой мощью. Старуха предупреждает о надвигающейся опасности разумных спутников. Девочка в шапочке с кошачьими ушками втягивает вас в войну против голографических уличных кошек, которые на самом деле корпоративные устройства видеонаблюдения. Полуодетая монахиня предлагает торговать болью — умирать снова и снова на протяжении всех миссий, в терминах игры — ради кармы: это валюта игры *Ruiner* за прокату и разблокировку возможностей.

Пули, сабли, красота

Крипы [Creeps], первая группа врагов, которую вас послали убить, лучше всего воплощает то, как *Ruiner* мастерски увязывает свою карательную трудность с историей своего мира. Принужденные к отчаянному выживанию, Крипы напичканы имплантатами и кибернетикой, для ускорения движения и твердого прицела. Спускаясь в их дом, подземный паркинг, вы ощущаете, как будто входите в залы забытого королевства. Их лидер, колючий парень по имени Нерв [Nerve], отбрасывает свою меркантильную сущность и призывает к благородной схватке на мечах.



В промежутке между боями есть шанс получше узнать обитателей Ренгкока.



Хитрая усмешка и подмигивание — не то, что скрывается за этой маской.

Когда приходит время сражаться, боевые действия быстрые и плавные — танец уклонений, стремительные атаки, переброски-отступления и ускользающая цель. Мышь контролирует, куда вы целитесь, и для перемещения вас вверх, вниз, влево и вправо отведены клавиши WASD, независимо от того, в какую сторону направлено ваше оружие, делая меткие попадания возможными даже во время уклонения от ударов, способных быстро вас убить, если задедут сразу несколько.

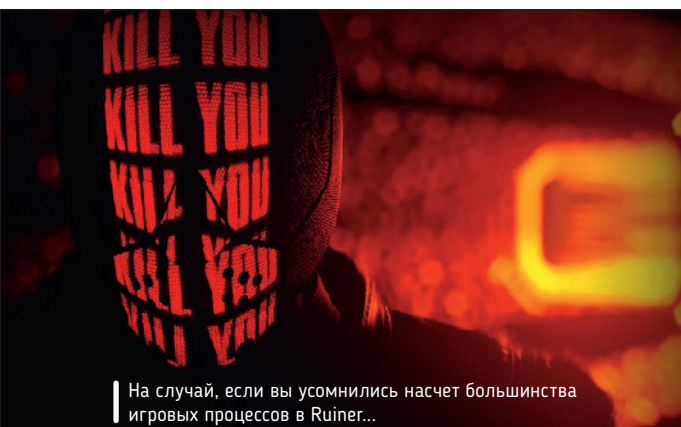
Всё это зависит от способности Dash — аккуратных щелчков правой кнопкой мыши, чтобы рвануть на небольшое расстояние в текущем направлении. Играть на Hard — уровень сложности, рекомендуемый Reikon — это мучительное зрелище пугающе длинных цепочек смерти; но вы никогда не почувствуете, что ваше разочарование побеждает ваше желание улучшений, продвигающее вас в следующую жизнь.

Помогают этому чувству «ну еще один ход» орудия и холодное оружие, уроненные врагами и разбросанное по всему уровню. Разнообразие его впечатляет — классические баллистические винтовки, взрывоопасные пусковые установки, огнемёты, пулеметы, корпускулярно-лучевые пушки и многое другое; но мы часто выбирали комбинацию для ближнего боя, дробовик и нож.

Бойцовский клуб для одного

Очарование *Ruiner* захватывает ударами в разгаре битвы, выплескиванием облаков с эффектами частиц, плавными анимациями, короткими движениями, огненными оранжевыми взрывами и глубокими тенями. Когда вы врежетесь в отряд боевиков с разящим хромированным мечом в руках, у вас, несомненно, мелькнет воспоминание о главном герое Хиро [Hiro Protagonist] из *Snow Crash*.

После покорения Крипсов вы не будете сталкиваться с большим разнообразием, чем враги ближнего боя, идущие напрямик на вас, и их вооруженные прихвостни, бегающие кругами, чтобы вас одурманить, при этом целясь в вашу скрытую маской башку. Было бы неплохо сразиться с более непредсказуемыми врагами.



На случай, если вы усомнились насчет большинства игровых процессов в *Ruiner*...

Некоторые бойцы босса также становятся жертвой повторений, и методы победить их не особенно сложны.

Так же и со многими схемами карт игры: схема коридор-уби-вальня-коридор во втором акте *Ruiner* начинает казаться несвежей. Конечно, есть исключения: замечательный лабиринтоподобный гараж Крипсов и, на более позднем уровне, прекрасная тема спуска в подземный мир проклятых.

Вы будете азартно пользоваться преимуществами гибкости системы начального накопления оружия для персонажа, которая использует баллы, обмениваемые на включение или отключение пассивных бонусов и активных навыков «на лету». Тем не менее, некоторые из вариантов, таких как падение уровня запасов или развертываемая защитная стена, в основном бесполезны по сравнению с первостепенной важностью Dash, из-за головокружительной скорости боя *Ruiner*.

Ударный эффект пушки, наоборот, чересчур ошеломителен. Возможно блокировать группы и врагов босса с минимальными затратами энергии, и легко их устранять. Тем не менее, свобода выбора при определении начального арсенала оружия хороша, и неизбежные ускорения «без возможности» — это то, что можно будет рассмотреть.

Мы потратили на *Ruiner* чуть более шести часов. Конец настал неожиданно, когда внезапно исчезли кредиты после проезда по 15 зонам и заключительной схватки, которая была не более чем церемониальным выяснением отношений. Reikon планирует дополнительные игровые режимы через обновления DLC, но мы бы хотели, чтобы они развили или расширили исследование города. Пока что это время затеряться в биомассе настолько, насколько сможете. **LXF**



Убийство своих врагов с максимальной жестокостью скоро становится второй натурой.

ВЕРДИКТ

РАЗРАБОТЧИК: Reikon Games

САЙТ: www.ruinergame.com

ЦЕНА: £ 14,99

СЮЖЕТ	7/10
ГРАФИКА	8/10
УВЛЕКАТЕЛЬНОСТЬ	6/10
ОПРАВДАНОСТЬ ЦЕНЫ	8/10

Ослепительно, опасно и источает стиль; вкратце, *Ruiner* — водоворот из кибер-насилия и осмотра достопримечательностей.

» **Рейтинг 8/10**

Мобильные новости

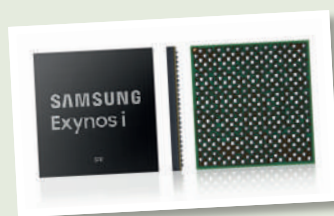
ДЛЯ ИНТЕРНЕТА ВЕЩЕЙ

Меньше энергии, больше охват

Чип Samsung Exynos i S111 обеспечивает высокую эффективность и надежность работы устройств NB-IoT.

Новое решение для узкополосного Интернета вещей (NB-IoT) от Samsung — процессор Exynos i S111. Он дает чрезвычайно большую зону охвата IoT-сети и высокую точность геопозиционирования, потребляет мало энергии и обладает мощными функциями безопасности, а также оптимизирован для современных задач мониторинга в режиме реального времени, в том числе в носимых устройствах или умных счетчиках. Exynos i S111 включает модем, процессор, память и систему глобального позиционирования Global Navigation Satellite System (GNSS), позволяя производителям умных устройств повысить эффективность и гибкость своих продуктов. Для большей надежности коммуникаций на больших расстояниях в технологии NB-IoT

применяется механизм ретрансляции данных, который постоянно осуществляет попытки передачи данных, пока они не передадутся или не выполнится установленное число ретрансляций. Благодаря поддержке нескольких сессий ретрансляции, устройство S111 будет передавать данные на расстояние более 10 км. Модем на чипе поддерживает сети LTE Rel. 14, со скоростью приема данных до 127 кбит/с и передачи до 158 кбит/с, и может работать в автономном, внутриполосном и защитном режимах. Для длительной работы в режиме ожидания предусмотрены режим сохранения энергии (power saving mode, PSM) и режим расширенного периодического приема (expanded discontinuous reception, eDRX), что обеспечивает автономную работу устройства до 10 лет и более.



Новое IoT-решение включает модем, процессор, память и функции GNSS, реализованные в едином чипе.

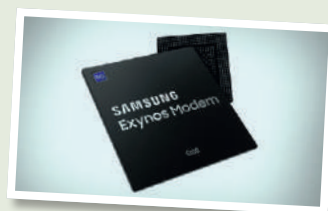
МОДЕМЫ ОТ SAMSUNG...

Готовя переход на 5G

Представлен «первый в мире» модем Samsung 3GPP 5G со скоростью 6 Гбит/с.

Модем Samsung Exynos Modem 5100, выполненный по техпроцессу 10-нм, поддерживает скорость загрузки до 6 Гбит/с в спектре миллиметровых волн (mmWave), и до 2 Гбит/с в спектре ниже 6 ГГц. По утверждению компании, ее первый модем 5G соответствует последней спецификации стандарта 3rd Generation Partnership Project (3GPP) для 5G New Radio. Samsung доказала работоспособность своей технологии, проведя тестовую беспроводную передачу данных через базовую станцию 5G и новый модем, встроенный в прототип устройства для конечного пользователя. А пока сети 5G не вышли из тестового режима, Samsung Exynos Modem 5100 может работать и с сетями 4G, обеспечивая быструю и стабильную передачу данных

с «нисходящей скоростью» 1,6 Гбит/с. Samsung заявила, что ее модем поможет «обеспечить ускоренную разработку и коммерциализацию мобильных устройств 5G», но он имеет потенциал не только для смартфонов Galaxy следующего поколения, а и устройств Интернета вещей (IoT), формируя интеллектуальные сети гаджетов и далее расширяя возможности Samsung в области совместимой с SmartThings электроники. Exynos 5100 также упростит использование технологий ИИ, обеспечивая интеллектуальным устройствам подключение к выделенному соединению с облачной или центральной системой, что снизит требования к мощности процессоров таких устройств. Samsung обещает, что новый модем появится в реальных устройствах к концу 2018 г.



Samsung Exynos Modem 5100 – первый в отрасли модем 5G, полностью совместимый со стандартом 3GPP версии 15.

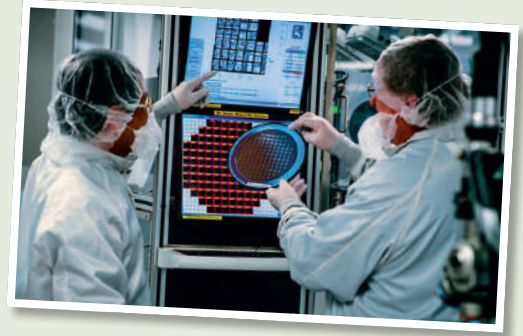
...И ОТ QUALCOMM

Новый флагман

OEM-производители получили образцы 7-нм мобильного процессора Qualcomm с поддержкой 5G.

Компания Qualcomm подтвердила, что процессор Snapdragon следующего поколения будет выполнен по техпроцессу 7 нм и получит поддержку 5G — чипсет комплектуется модемом Snapdragon X50. Официального наименования у будущего SoC пока еще нет; можно предположить, что это будет Snapdragon 855 (вслед за Snapdragon 845 и Snapdragon 850). Известно, что уже начата отправка нового SoC производителям аппаратных средств; в новейших смартфонах и планшетах (а потенциально — и в других смарт-гаджетах) этот чипсет мы скоро увидим (хотя Qualcomm не упомянула, о каких именно OEM-производителях идет речь). «Мы очень рады работать с OEM-производителями, операторами, поставщиками инфраструктуры и органами стандартизации всего мира, а также готовы к запуску первых мобильных точек доступа 5G к концу 2018 г., и выпуску смартфонов, использующих мобильную платформу следующего поколения, в первой половине 2019 г., — заявил президент Qualcomm Incorporated Криштиану Амон [Cristiano Amon]. — Qualcomm Technologies остается лидером в области исследований и разработок, что

позволит в будущем увеличить инновации во многих секторах, поскольку соединения 5G станут повсеместными». Эта цитата позволяет предположить, что новый чипсет найдет применение не только в смартфонах, но и в устройствах Интернета вещей (IoT), потенциально предоставляя способ связи множества гаджетов 5G, позволяющих воспользоваться интеллектуальной городской инфраструктурой и тому подобными вещами. Новый чипсет также поставляется с технологией искусственного интеллекта Qualcomm, благодаря чему оснащаемые им гаджеты выполняют свои «интеллектуальные» функции, не полагаясь на непрерывное подключение к Интернету. Полностью технические особенности нового флагманского мобильного процессора будут раскрыты в четвертом квартале 2018 г., как раз перед анонсом новых устройств, этим процессором укомплектованных.



Будущий флагманский процессор Qualcomm получит поддержку стандарта 5G.

СЕТИ 5G

Сдвигая сроки

В 2019 г. в США будут сети 5G, а основные OEM-производители выпустят телефоны этого стандарта.

Qualcomm неоднократно предсказывала, что массовые сети 5G появятся не ранее 2020 г., но теперь назван новый срок. «Несмотря на переживаемый кризис, мы поддерживали уровень расходов на исследования и разработку, так что появление 5G ускорится более чем на год», заявил президент Qualcomm Криштиану Амон. По его словам, сетевые операторы США заинтересованы во внедрении стандарта 5G, и смогут относительно быстро достичь массового покрытия, добавив за следующий год оборудование 5G к уже существующим сотовым вышкам. «Мы собираемся начать запуск сетей 5G в США уже в апреле 2019 г. К этому времени пользователи смогут купить и телефоны с поддержкой 5G». Это стало возможным благодаря тому, что в последние годы Qualcomm расширила базу своих клиентов почти до 9 тыс., среди которых 18 из 25 ведущих мировых производителей автомобилей, множество IoT-компаний, брендов

бытовой электроники, производителей сетевого оборудования Wi-Fi. «Мы хотели поглотить компанию NXP, но без дела не сидели. 21 месяц, ожидая решения, мы неуклонно соблюдали свою стратегию и получили \$5 млрд доходов от бизнеса по производству чипов [без учета смартфонов], т.е. за 2 года рост составил 70%. Это убеждает нас в правильности стратегии, и мы ее продолжим, так как мобильные устройства изменяют все отрасли». Г-н Амон добавил, что все топовые производители смартфонов из числа клиентов Qualcomm планируют выпустить в следующем году телефоны на Android с поддержкой 5G. «Я не встречал человека, не желавшего бы более быстрого интернета. Имея дисплей 4K, отличную камеру и фантастическую картинку, вы хотите как можно быстрее поделиться ею. Оператор сети 5G получает более низкую стоимость каждого бита, а потому у него есть стимул для перехода на безлимитные планы 5G». **LXF**



Президент компании-чипмейкера Qualcomm Криштиану Амон уверен, что в США мобильные сети 5G будут доступны уже к апрелю 2019 г.

Сравнение

Audacious » Banshee » Clementine
» DeaDBeeF » Rhythmbox



Шашанк Шарма

днем — благово-
спитанный адвокат
в суде Нью-Дели,
а ночью — каратель
открытого кода!

Музыкальные плееры

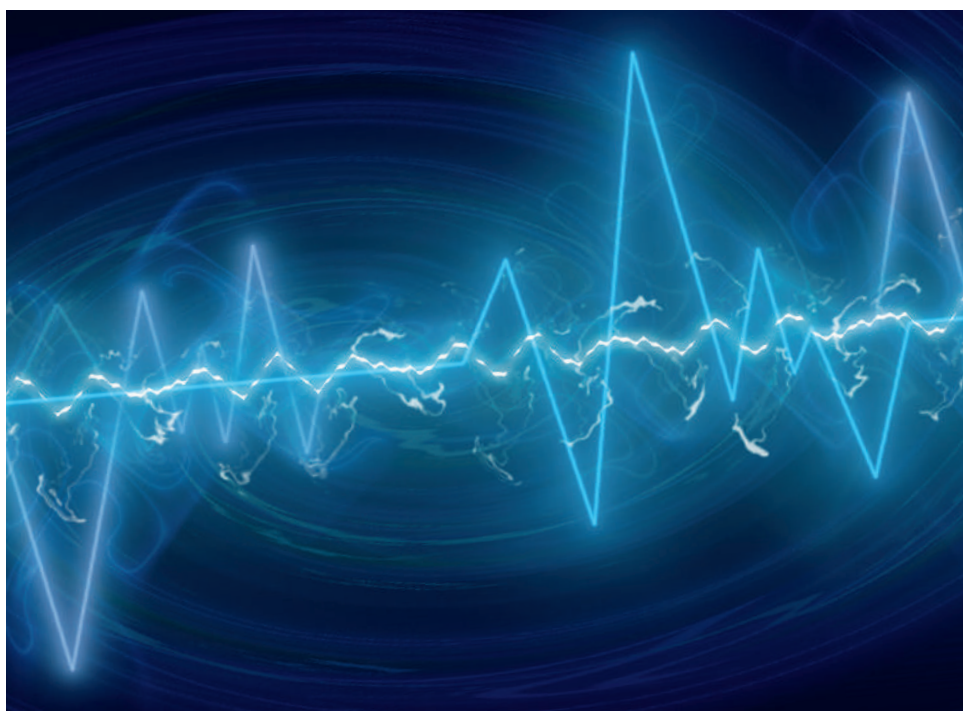
Работая, **Шашанк Шарма** предпочитает не слушать музыку с вокалом, поэтому ищет плееры, которые убирают голоса, да и другое умеют...

ПРО НАШ ТЕСТ...

Помимо забавы и простоты, мы ищем в инструменте поддержку множества форматов файлов, таких как MP3 и OGG. Нам также интересно узнать, можно ли с помощью инструмента подключиться к интернет-радио. И, возможно, главное: нам нужен проект, не зависящий от дистрибутива.

Мы протестировали наши плееры на Ubuntu и на Fedora, запустив Budgie и Gnome, чтобы проверить их навыки в интеграции с рабочим столом. В отличие от своих коллег-видеоплееров, музыкальные плееры не особенно алчны до ресурсов. Тем не менее, мы по-прежнему не против того, чтобы плеер был как дома и на более старых, мало-мощных машинах, и на современных вычислительных монстрах.

Поскольку музыкальные вкусы со временем меняются, а также потому, что многие из ваших любимых групп будут продолжать выпускать музыку — увы, кроме *The Beatles* и *The Spice Girls* — мы также посмотрим, поддерживают ли плееры приобретение музыки и можно ли расширить их основную функциональность плагинами.



Недавний семейный послеобеденный разговор плавно перешел в обсуждение предстоящих письменных заданий для **LXF**, и ваш автор сообщил коллеге из **LXF** Маянку Шарме о своем намерении написать *Сравнение* музыкальных плееров. «А что, остались люди, которые всё еще пользуются плеерами?» — удивился старший Шарма, пояснив, что с появлением потоковых сервисов и доступных интернет-радиостанций, возможно, спрос на такие приложения исчез.

«А что, по-твоему, людям следует делать со всеми своими музыкальными файлами?» — самоуверенно спросил автор, и тем выиграл спор. Это было очень забавно и очень легко. И к тому же эти два

определения лучше всего описывают идеальный музыкальный плеер.

Лучший во всех этих настольных приложениях — то, что они обеспечивают немедленный доступ к вашей любимой музыке, а также и к различным онлайн-источникам. Хотя онлайн-радиостанции легкодоступны, они не обеспечивают столь быстрое удовлетворение, как своя музыкальная коллекция.

При таком количестве файлов вполне естественно иметь некую систему каталогизации, и многие плееры выступают заодно и в качестве органайзеров.

Мы здесь ограничились графическими инструментами, но есть и опции для командной строки, предлагающие ту же функциональность. Мы упоминаем о них в разделе *Рассмотрите также*.

Популярные функции

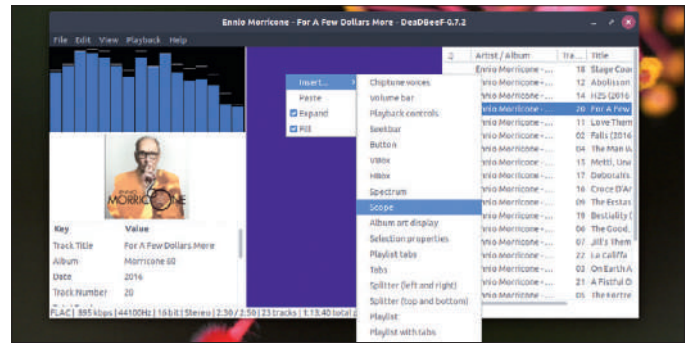
Что выделяет их из толпы?

В отличие от своих прародителей, позволявших только создавать списки воспроизведения (плей-листы) и, иногда, настраивать эквалайзер, современные плееры умеют гораздо больше.

Простоватый на вид *Audacious* прекрасно себя чувствует даже на мало-мощных машинах. Потомок XMMS, он поддерживает скины *Winamp 2*, и вы можете переключить свой внешний вид с *GTK* по умолчанию на более яркую альтернативу, нажав **File > Settings > Appearance** [Файл > Настройки > Внешний вид]. Кроме воспроизведения CD, этот инструмент пригоден для копирования всего альбома или отдельных треков. В отличие от многих других, *Audacious* также предлагает набор элементов управления воспроизведением с терминала и может похвастаться кучей горячих клавиш по умолчанию.

Как и *Audacious*, *Banshee* пригоден для импорта треков с CD. Помимо обычных пользовательских плей-листов, *Banshee* может еще создавать интеллектуальные плей-листы. Приложение предоставляет вам на выбор различные параметры, а затем создает динамический список воспроизведения соответствующих треков из всех файлов в вашей музыкальной коллекции. Например, можно создать список воспроизведения, содержащий только треки, которые вы оценили четырьмя или более звездами.

В отличие от остальных, *Clementine* поддерживает ряд онлайн-сервисов хранения и позволяет воспроизводить файлы из вашей учетной записи на Dropbox, Диске Google, OneDrive и т.д., а также довольно эффективен при перекодировании файлов в MP3, Ogg Vorbis, FLAC и другие форматы. Наряду с обложкой из Last.fm и Amazon, приложение также использует MusicBrainz для извлечения отсутствующих тегов метаданных ваших музыкальных файлов. И *Audacious*, и *Clementine* позволяют создавать плей-листы со вкладками, а значит, вы можете переключаться между жанрами или настройками одним щелчком мыши. Как и *Banshee*, *Clementine* поддерживает



Все инструменты предлагают функцию плавного воспроизведения, но в *Banshee*, *Clementine* и *DeaDBeeF* она реализована не идеально.

множество музыкальных плееров для Android, Apple и прочих, и можно перемещать файлы между этими устройствами и туда, и обратно.

DeaDBeeF управляется еще и из терминала, и это единственный инструмент, предлагающий переносимую версию, т.е. он не требует установки. Он поддерживает почти все возможности своих соперников, в том числе Ogg Vorbis, MP3 и онлайн-радиопотоки AAC.

Поскольку *Rhythmbox* долго был музыкальным плеером по умолчанию в Gnome, он неплохо интегрирован с *Nautilus*, и это единственное приложение, предлагающее проигрывание при наведении и интеграцию контекстного меню. Вы также можете использовать его для записи треков на CD.

ВЕРДИКТ

AUDACIOUS	9/10	DEADBEEF	9/10
BANSHEE	10/10	RHYTHMBOX	10/10
CLEMENTINE	10/10		

Все плееры предлагают похожие функции, но некоторые, а именно *DeaDBeeF* и *Audacious*, имеют крутую кривую обучения.

Онлайн-радио

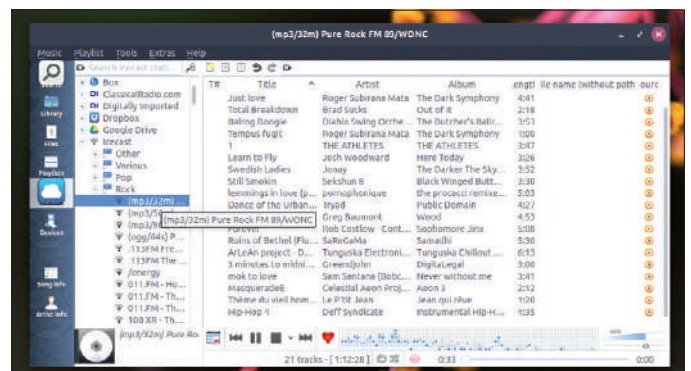
Услуги никуда не деваются.

К ак вам ни трудно такое представить, есть люди, чье единственное взаимодействие с музыкой — это повтор трека в качестве мелодии звонка на телефоне соседа по автобусу. Это, да еще радио. Если вы тоже из таких личностей или просто хотите расширить свою палитру новыми звуками, можете использовать любое из наших приложений для подключения к онлайн-радиослужбам, которые часто дают возможность выбирать жанры музыки для прослушивания.

И если вы хотите постоянно быть в курсе, что новенького появилось в роке или хип-хопе — вместо того, чтобы раскошелиться на какой-нибудь новый альбом, лучше потратить некоторое время на онлайн-радиостанцию, посвященную этому жанру. Вы обязательно столкнетесь с чем-нибудь интересным для вас.

DeaDBeeF умеет воспроизводить вещание онлайн-радиостанции, если они предоставляют либо загружаемый плей-лист, либо URL-адрес для потока MP3. В первом случае, чтобы использовать радио, вы должны выбрать опцию **File > Load playlist** [Файл > Загрузить плей-лист]. Или нажмите **File > Add Location** [Файл > Добавить местоположение], если у вас есть доступ только ко второму варианту. В отличие от приложений типа *Rhythmbox*, *Clementine* и *Banshee*, *DeaDBeeF* автоматически не обеспечивает потоки от Jamendo или других сервисов, предоставляющих бесплатный доступ к их коллекциям.

Audacious также поддерживает воспроизведение онлайн-радиостанций, будь то потоки MP3 или OGG, или плей-листы. Для немедленного запуска воспроизведения локальных музыкальных файлов или потоковой передачи



Будьте готовы изумиться огромным количеством интернет-услуг, предлагаемых *Clementine* — это самый богатый функциями плеер в нашем Сравнении.

из списка воспроизведения радио вы должны выбрать опцию **Open File** [Открыть файл].

Многие радиостанции, например, в Shoutcast, дают возможность прослушивать плей-листы M3U, но почему-то не всегда ладят с *Rhythmbox*.

ВЕРДИКТ

AUDACIOUS	10/10	DEADBEEF	6/10
BANSHEE	10/10	RHYTHMBOX	8/10
CLEMENTINE	10/10		

Хотя *DeaDBeeF* и воспроизводит онлайн-радио, его интерфейс оставляет желать лучшего.

Взаимодействие с пользователем

Приятный на вид
и отзывчивый = Win!

Несмотря на то, что состав функций всех плееров в нашем *Сравнении* примерно одинаков, с одними справляться легко и приятно, а другие даже выглядят неуклюжими. Дело не просто в размещении элементов управления, которое, впрочем, тоже имеет значение, но и в способе их работы. Не все приложения интуитивно понятны, а отсутствие подробной документации может означать, покинете вы приложение или нет — особенно при таком богатом выборе.

Если вы любитель поработать под спокойную фоновую музыку, вы, скорее всего, будете равнодушны к внешнему виду плеера и, вероятно, предпочтете тот, который сворачивается в системный лоток и не мешает рабочему процессу. А если вы на другом конце спектра и относитесь к своим музыкальным удовольствиям серьезно, то лучшим вариантом будет привлекательное, хорошо разработанное приложение, имеющее встроенный эквалайзер, с которым можно поиграть.

Мы уверены, что учли здесь обе пользовательские базы, и вы легко найдете плеер, который вам понравится.

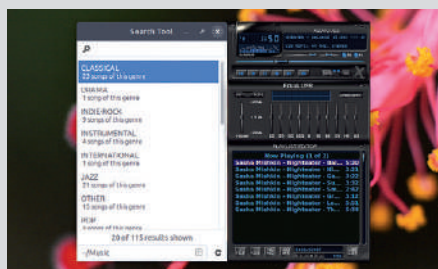
Audacious

9/10

В отличие от многих других плееров, которые пытаются втиснуть все элементы управления в единственное окно приложения, *Audacious* не возражает против их частичного выноса в новые окна. Эквалайзеру, например, отведено отдельное окно, без возможности его интеграции в основной корпус.

Это становится еще заметнее, когда вы переключаетесь на тему в стиле *Winamp* и обнаруживаете, что функция поиска, которая является отличительной чертой приложения, прикована к собственному окну. Здесь показан интерфейс *Winamp* этого инструмента, но присущий ему собственный внешний вид показан далее, в разделе «Расширения».

Интерфейс *Winamp* также скрывает от пользователя массу важных функций, которые в теме *GTK* легко доступны по умолчанию. Производительность плеера не зависит от используемой темы, и если вы не включите уведомление OSD, плеер будет играть себе, не отвлекая вас от основной деятельности.



Banshee

7/10

В *Banshee* левая панель упрощает перемещение по вашей музыкальной коллекции. Полезные ссылки на онлайн-сервисы вроде Magnatune и даже Internet Archive позволяют легко переключаться с локальных файлов на альтернативные источники. По умолчанию инструмент не показывает обложку для воспроизводимой дорожки, но вы можете это изменить, нажав View > Show Cover Art [Просмотр > Показать обложку]. Однако учтите, что включение данной опции сужает левую боковую панель, приводя к неприятно-захлапленному внешнему виду. В отличие от других инструментов, если у вас установлены соответствующие кодеки, *Banshee* может также удвоиться, став еще и видеоплеером, и поддерживает воспроизведение нескольких различных форматов.

Хотя при воспроизведении треков *Banshee* отнюдь не тормозит и чутко реагирует на настройку онлайн-радио или других сервисов, его функция импорта разочаровывает. Он часто «падал» на нашей машине с Ubuntu и был таким же шатким на Fedora.



Поддержка и документация

Мастерство приходит с пониманием.

A*udacious* поддерживает ответы по ряду распространенных проблем, способных возникнуть, но за прочими запросами и рекомендациями, связанными с самим программным обеспечением или любыми его плагинами, вы должны обращаться к официальным форумам.

Banshee содержит часто задаваемые вопросы (ЧаВо, FAQ) по целому ряду тем, таких как его бесчисленные функции, поддерживаемые аппаратные средства, файлы настройки, база данных фонотеки и т.д. Если ваш конкретный запрос еще не охвачен, можно обратиться к официальным форумам.

У *Clementine* нет ни ЧаВо, ни руководства пользователя, но на странице GitHub помещена малонаселенная вики, про установку и базовое использование. Для конкретной помощи надо присоединиться к *Clementine* в Google Groups, подключившись к сообществу пользователей и разработчиков, или связаться с командой разработчиков через IRC-канал #clementine на Freenode.

ЧаВо *DeaDBeeF* затрагивают только пару вопросов, относящихся к новым пользователям, и вики пока столь же скудная, без значимого контента, кроме некоторых ссылок на несколько сторонних плагинов. Приложение *DeaDBeeF* может похвастаться встроенной опцией помощи, но предлагает

немногом больше, чем информирование об отображении обложки для ваших музыкальных файлов.

У *Banshee* и *Rhythmbox* есть руководства пользователя, доступных в Интернете или из инструмента по нажатию F1. Помощь *Banshee* лучше организована и разделена на категории: это импорт файлов, подключение к онлайн-радиостанциям и подкастам, редактирование музыкальных тегов и других метаданных, синхронизация с переносными плеерами, работа с расширениями и многое другое. Кроме канала #rhythmbox на irc.gimp.org, есть ЧаВо по категориям: обложки, работа с плагинами и т.д.

ВЕРДИКТ

AUDACIOUS	7/10	DEADBEEF	2/10
BANSHEE	10/10	RHYTHMBOX	8/10
CLEMENTINE	7/10		

Наряду с *Banshee*, *Rhythmbox* — один из наиболее тщательно документированных инструментов, представленных в нашем Сравнении.

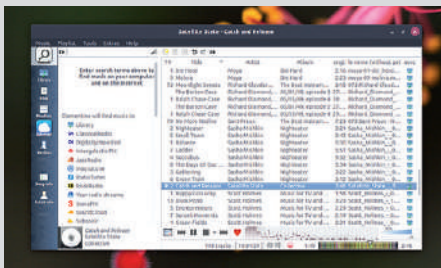
Clementine**10/10**

Помимо боковой панели, списка вкладок слева и поддержки большого количества онлайн-сервисов, еще одна особенность, которая отличает *Clementine* — это поиск. Хотя поиск *Banshee* также достоин похвалы, *Clementine* находится в отдельной лиге.

Как показано на рисунке, инструмент позволяет использовать диалоговое окно поиска для поиска файлов на локальном компьютере, а также по множеству различных настраиваемых онлайн-источников, и даже указывает те, где он искать не будет, и причину для этого. Например, на нашей машине *Clementine* заявляет, что не будет искать файлы в OneDrive, Box, DropBox и т.д., так как мы не вошли в нашу учетную запись.

На основании ваших предпочтений в музыке и настройках прослушивания, приложение также автоматически запускает несколько поисковых запросов, чтобы помочь вам найти музыку, которая вас обрадует.

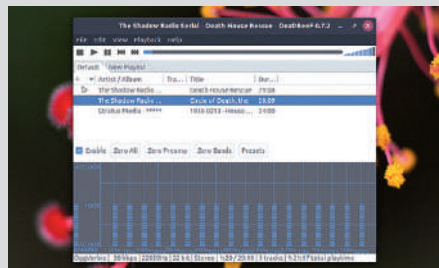
С поддержкой gpodder.net вы также можете применить этот инструмент для поиска подкастов.

**DeaDBeeF****8/10**

Интерфейс всех других приложений позволяет одновременно просматривать обложку и плей-лист, но в *DeaDBeeF* такого сделать нельзя — он заставляет вас выбирать, какой элемент (обложка, браузер списка воспроизведения, вкладки списка воспроизведения, громкость, и т.д.) отобразить в основном окне. К счастью, можно создавать в настраиваемом главном фрейме добавочные горизонтальные или вертикальные блоки, которые затем можно заполнить другими вариантами.

Настраиваемость в сочетании с переносимостью и производительностью оправдывает высокий балл *DeaDBeeF*, но он не для пользователей, желающих, чтобы плеер выглядел классно прямо из коробки.

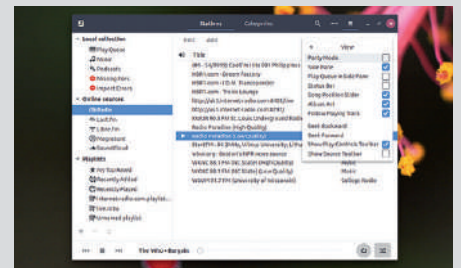
Довольно нелепо, что если вы импортируете дополнительные файлы во время прослушивания дорожки из плей-листа, приложение создает новый плей-лист и переключается на него, вместо добавления новых треков в существующую очередь.

**Rhythmbox****8/10**

Banshee и *Rhythmbox* имеют почти одинаковый внешний вид, и между ними существуют только тонкие различия. Оба эти инструмента Gnome хорошо приняты в различных дистрибутивах Linux, хотя *Rhythmbox* всегда пользовался большей популярностью.

Несмотря на то, что эти инструменты существуют целую вечность, интерфейс у них довольно примитивен. Если бы *Clementine* не хватило ума применить левую боковую панель со вкладками, которая несколько расцветивает интерфейс, этот плеер тоже угодил бы в список унылых зрелищ; зато благодаря своей простоте *Rhythmbox* представляет более чистый интерфейс, чем *Clementine*. Добавив соответствующие плагины, вы можете легко изменить внешний вид, разместив текст, обложки и т.д.

Некоторые дополнительные плагины, например, браузер радио, способны поломать инструмент, но удаление подобных надстроек моментально возвращает плеер к его музыкальной сущности.



Музыкальные онлайн-магазины

Когда вы в настроении потратиться.

У пользователей ныне есть выбор множества различных интернет-магазинов музыки. Но многие коммерческие услуги ограничивают свои предложения определенными странами и территориями, так что этот тест может не иметь отношения ко всем пользователям.

Audacious и *DeaDBeeF* не поддерживают интернет-магазины, и вы не можете импортировать файлы из своей учетной записи в любую из служб.

Если вы поклонник службы Amazon MP3, импортировать приобретенную музыку в *Banshee* вы сможете без особых затрат времени и энергии. Функциональность предлагается через плагин, включенный по умолчанию — приложение постоянно предлагало эту услугу за последние несколько лет и выпусков. К сожалению, другие коммерческие магазины *Banshee* не поддерживает. В зависимости от вашего дистрибутива можно также установить кое-какие расширения от сообщества для Jamendo и Magnatune. На наших тестовых машинах мы не могли заставить плагин Jamendo работать, но Magnatune работает как ожидалось.

Clementine поддерживает огромный спектр услуг, включая Digitally Imported, Magnatune и SoundCloud. Поддерживаются и дополнительные

сервисы, такие как Spotify, но по умолчанию плагин не установлен из-за ограничений лицензирования. Можно настроить эти службы, нажав Tools > Preferences [Инструменты > Параметры] и перейдя к любой на левой боковой панели в разделе Internet Providers [Интернет-провайдеры], хотя для многих из этих служб вам потребуется премиум-аккаунт. Тем не менее, вы можете бесплатно скачать с Jamendo каталог ~150 МБ, получив столько треков, что на их прослушивание вам и жизни не хватит!

Даже *Rhythmbox* выдает доступ к службам Magnatune и SoundCloud — через плагины, которые надо явно включить в диалоговом окне Preferences.

ВЕРДИКТ

AUDACIOUS	Нет	DEADBEEF	Нет
BANSHEE	6/10	RHYTHMBOX	5/10
CLEMENTINE	9/10		

Несмотря на поддержку пары онлайн-сервисов, из *Rhythmbox* нельзя покупать треки и альбомы в Интернете.

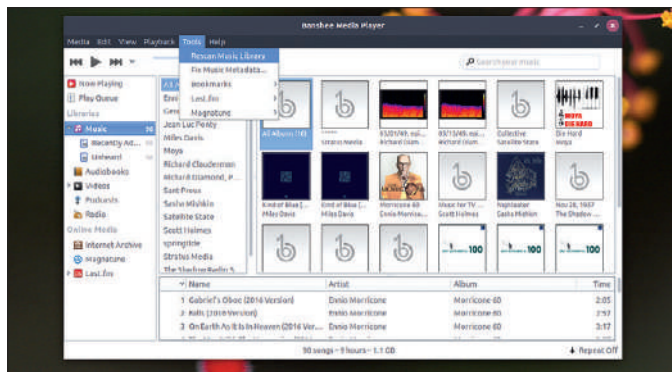
Организация

Понимание вашей музыки — и вашей библиотеки.

Когда вы добавляете файлы в *Audacious*, для них автоматически создается плей-лист, но для этих треков не предусмотрено организации. А треки в музыкальной библиотеке приложение обрабатывает иначе, разделяя их по альбомам, исполнителям и даже жанрам. Это, кстати, означает, что ваша фонотека может показывать 12 песен под именем Майлса Дэвиса [Miles Davis] и те же треки под заголовком Джаз [Jazz]. Несмотря на повторяемость, такая демаркация особенно полезна, если у вас обширная коллекция. В приложении также наличествует инструмент поиска, пригодный для поиска определенных треков в библиотеке. Опять же, просматриваются только файлы, импортированные в библиотеку, но не список воспроизведения. Однако *Audacious*, кроме того, предлагает кучу всяких опций, таких как Title, Artist, Album, Length и т.д., для сортировки треков в любом имеющемся плей-листе.

У *Banshee* и *Rhythmbox* есть опция умных плей-листов, пригодная для автоматического создания плей-листов по заданным пользователем критериям — например, «недавно добавленные». Если ваши файлы случайно угодили в одну папку, установка *Rhythmbox* по умолчанию не сможет рассортировать их по значимым категориям на основе имени исполнителя или альбома. Желая это сделать, вы должны установить, а затем включить плагин органайзера, который затем можно употребить для сортировки ваших файлов.

DeaDBeeF, напротив, не имеет каких-либо организаторских навыков, достойных упоминания — разве что позволяет создавать плей-листы. Даже



Banshee хранит библиотеку и другие данные в базе данных SQLite, хотя на Ubuntu склонен к внезапным авариям.

функция поиска ищет ключевые слова только в текущем выбранном плей-листе, а не во всей музыкальной коллекции, что невероятно расстраивает по сравнению с другими плеерами нашего Сравнения.

Левая боковая панель *Clementine* позволяет легко перемещаться между устройствами, файлами, интернет-провайдерами, по библиотеке и другим местам, и даже распределяет вашу музыку по исполнителям и альбомам.

ВЕРДИКТ

AUDACIOUS	10/10	DEADBEEF	4/10
BANSHEE	10/10	RHYTHMBOX	10/10
CLEMENTINE	10/10		

Имя у *DeaDBeeF* не слишком вразумительное, но с его помощью вы сможете образумить свою коллекцию — дайте только срок.

Расширения

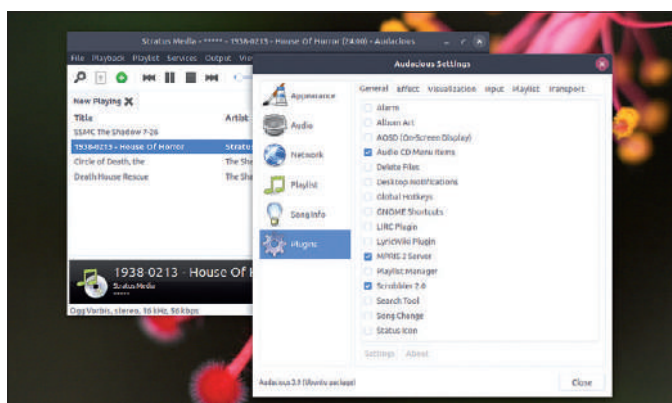
Когда базовой установки мало.

При всех мириадах функций, предлагаемых музыкальными плеерами из нашего Сравнения, они могут сделать еще больше — особенно если включить правильные плагины. В *Audacious* для доступа к дополнительным функциям не нужно делать ничего особенного. Нажмите File > Settings и выберите Plugins [Плагины] на левой боковой панели. Все имеющиеся плагины делятся на разумные категории, предлагаемые как вкладки, такие как Effects [Эффекты], Visualization [Визуализация], Input [Вход], Transport [Подача] и т.д. Более общие настройки, такие как Alarm [Сигнал], Global hotkeys [глобальные горячие клавиши], Desktop Notifications [Уведомления], Search Tool [Поиск] и т.д., переносятся на вкладку General [Общее]. Для включения плагина достаточно поставить галочку. Иногда вы также можете настроить конкретный плагин, нажав кнопку Settings.

В дополнение к базовым расширениям, поставляемым вместе с *Banshee*, можно устанавливать расширения от сообщества. Как и в *Audacious*, набор расширений по умолчанию охватывает массу категорий, таких как Utilities [Утилиты], Online Sources [Интернет-источники], Device Support [Поддержка устройств] и т.д. Многие дистрибутивы Linux предлагают дополнительные расширения в своих репозиториях программного обеспечения, и их можно установить по отдельности или как единый пакет *banshee-community-extensions*. Два интересных и забавных расширения — FolderSync, применяемое для синхронизации музыкальных файлов с определенными папками, и Karaoke, которое фильтрует вокалы из песен.

Clementine может похвастаться изобилием функций «из коробки», но, к сожалению, механизма для расширения нет, и его даже нет в списке задач.

DeaDBeeF тоже поставляется с несколькими плагинами, которые управляются через диалоговое окно Preferences. Сайт GitHub и сайт проекта содержат



Все проекты могли бы поучиться у *Audacious* умному управлению плагинами.

ссылки на другие плагины, такие как боковая панель текста, браузер файлов и т.д., но вы должны вручную установить их сами.

Страница *Rhythmbox* в Gnome Wiki содержит 49 плагинов для серии 3.x. Интересен плагин Remote, для управления плеером из web-браузера.

Banshee и *Clementine* предлагают для управления своими плеерами Android-приложения. Android-приложение *Banshee* также может использоваться для управления плеером *Rhythmbox*.

ВЕРДИКТ

AUDACIOUS	10/10	DEADBEEF	6/10
BANSHEE	10/10	RHYTHMBOX	10/10
CLEMENTINE	Нет		

Отсутствие у *Clementine* плагинов не отнимает ничего у проекта.

Музыкальные плееры

Вердикт

Xотя нам очень понравилось проводить время с *DeaDBeeF* (и это очень функциональный музыкальный плеер, который не мешает вам наслаждаться музыкой), его дизайн не назовешь интуитивно понятным. Вместо этого вы вынуждены тратить время на настройку внешнего вида, прежде чем сможете толком его использовать. Однако его переносимость и независимость от среды заслуживают похвалы. Мы уверены, что инструмент найдет больше поклонников, если его разработчики постараются улучшить документацию.

Все остальные плееры предлагают подобный же набор функций, даже если некоторые используют для этого плагины, добавленные сообществом. Как минимум, вы можете нацелить эти плееры на добывание обложки и, возможно, даже текста для различных треков из онлайн-ресурсов, помимо прослушивания онлайн-радиостанций и других музыкальных сервисов.

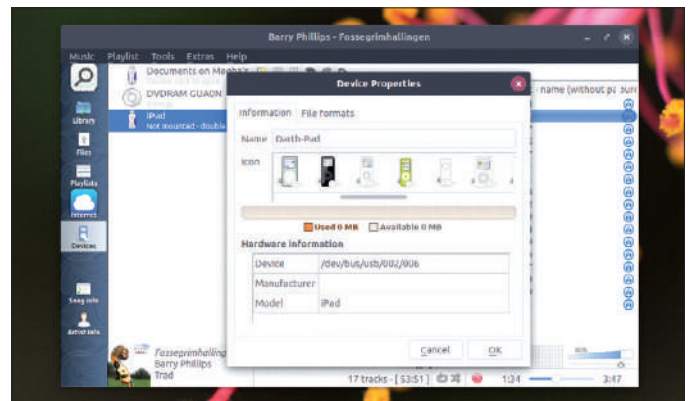
Функция поиска *Banshee* в общем впечатляет, но постоянные сбои при импорте файлов сильно разочаровали. И несмотря на то, что он несколько лучше своего ближайшего конкурента *Rhythmbox* и предлагает больше функций, скажем, поддержку импорта музыки из магазина MP3 Amazon, попасть на пьедестал плеер не смог.

Легкая расширяемость с помощью дополнений является одним из общих атрибутов самых успешных программ с открытым исходным кодом, будь то браузеры вроде *Firefox* или даже системы управления контентом вроде *Drupal*. Музыкальные плееры ничем не отличаются. Если бы не бездна его плагинов, *Rhythmbox* сразу слетел бы с пьедестала.

Несмотря на наличие темы *Winamp*, с *Audacious* лучше всего работать в его теме *GTK* по умолчанию, если вы хотите задействовать все его плагины. Дополнительный плагин для динамиков в *Effects* и удаление вокала легко преобразует вашу музыку и немало повеселит участников вечеринки, если караоке — это ваше.

Но несомненным победителем нашего Сравнения становится *Clementine*. Несмотря на меньший объем документации, чем у *Banshee* или *Rhythmbox*, и полное отсутствие расширяемости плагинами, *Clementine* умеет вполне достаточно, чтобы заработать первое место.

Все наши плееры управятся с вашей музыкальной коллекцией, но *Clementine* также невероятно легко и просто обеспечивает доступ к музыке из онлайн-источников, часто даже без реальных усилий с вашей стороны. Почему и выходит в победители. **LXF**



1^е место **Clementine** 9/10

Версия: 1.3.1 Сайт: clementine-player.org Лицензия: GPLv3

Предлагает всё, что вам нужно от музыкального плеера, прямо из коробки.

2^е место **Audacious** 8/10

Версия: 3.9 Сайт: <http://audacious-media-player.org> Лицензия: BSD

Почти не хуже Clementine, но не поддерживает столько онлайн-сервисов.

3^е место **Rhythmbox** 7/10

Версия: 3.4.2 Сайт: <https://wiki.gnome.org/Apps/Rhythmbox> Лицензия: GPL

Впечатляет, но можно бы еще позаботиться о своем интерфейсе.

4^е место **Banshee** 6/10

Версия: 2.6.2 Сайт: <http://banshee.fm> Лицензия: MIT

В последние годы застопорился. Часты сбои во время рутинных операций.

5^е место **DeaDBeeF** 6/10

Версия: 0.7.2 Сайт: <http://deadbeef.sourceforge.net/> Лицензия: GPL

Настраиваемый интерфейс — это весело, но нужно больше документации.

» РАССМОТРИТЕ ТАКЖЕ

Хотя мы ограничились графическими инструментами, есть и несколько вариантов командной строки, которые стоят вашего времени. За исключением поддержки онлайн-радиостанций и сервисов, таких как Magnatune и Jamendo, в остальном *cmus* — чрезвычайно полезная утилита терминала, со всеми прочими функциями, которыми обладают ее графические соперники.

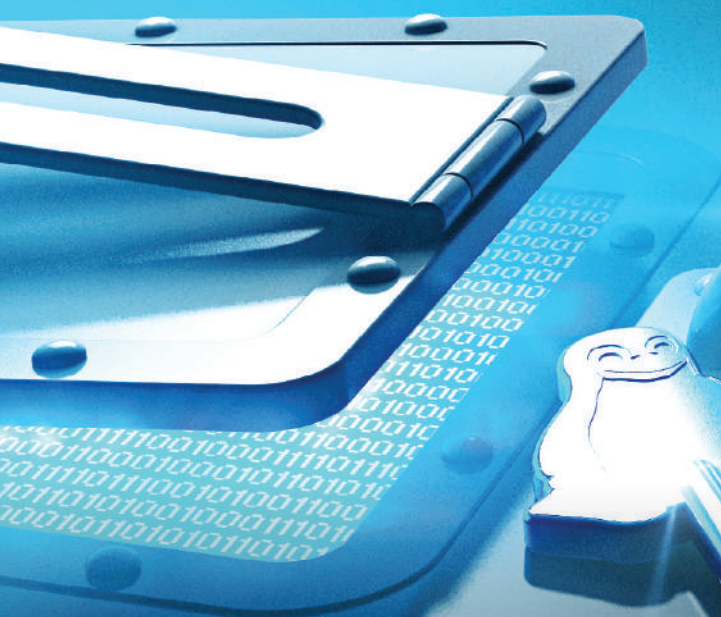
Если вместо вашей собственной локальной фонотеки вас интересуют только онлайн-источники, *Nuvola* — исключительно для вас.

Для тех, кто предпочитает тему *Winamp*, жизнеспособной альтернативой является *qmtmp*, но, как и *cmus*, это приложение не имеет поддержки для онлайн-сервисов. Как и *cmus*, *qmtmp* умеет извлекать текст песни с текущего воспроизводимого трека.

Пользователи с фонотеками на несколько ГБ могут рассмотреть вариант *Quod Libet*. Помимо воспроизведения музыки, приложение также является редактором тегов и органайзером библиотеки, и дает вам свободу распоряжаться, как организовывать ваши музыкальные файлы.

ТАЙНЫ ХАКЕРОВ

Наконец-то наступает ночь, и **Джонни Бидвелл**, облачившись в черную толстовку, тарачится в терминал. По правде, он это делал целое лето...



Понятие «хакерство» в популярных медиа обозначает незаконное проникновение в систему и ее злонамеренное использование ради профессионального интереса или просто для удовольствия (ха-ха). Однако изначально хакеры были просто энтузиастами, заинтересованными в создании и применении технологии с целью достижения новых и новаторских способов работы.

Многие программисты и любители спокойно называют свою повседневную работу хакерством, однако, вероятно, употребят другое слово, если им придется разговаривать с кем-то, скажем, на пограничном контроле. Так что простите нас за то, что мы здесь больше склоняемся к медийному определению. Нам просто очень нравится

фильм 1995 г. с Джонни Ли Миллером [Jonny Lee Miller]. А также впечатляющая история астронома Клиффорда Столла [Clifford Stoll], который в 1986 г. почти в одиночку вычислил хакера, сливавшего информацию в КГБ.

ГЛАВНОЕ – КОНТЕКСТ

«Многие программисты и любители называют свою повседневную работу хакерством.»

Злостные «черные» хакеры или киберпреступники являются истинным бичом общества, однако используемые ими хитрости и инструменты можно применять и в благих целях. На самом деле, «белые» хакеры и тестировщики на вторжение

честно зарабатывают себе на жизнь, делая практически то же самое, что их злонамеренные коллеги. Хотя и стараются воздерживаться от хищения денег с банковских счетов, удаления данных и взлома публичных web-страниц.

Знание технологии этих атак помогает нам защититься от них. Знание используемых ими уязвимостей помогает нам создавать более безопасные программы. А знание того, что доступ к системе без разрешения является незаконным, надо надеяться, уберечь вас от тюремного заключения.

Всю информацию этой статьи (разве что без лингвистических изысков) можно найти в Интернете, но используйте только свои компьютеры. В конце концов, мы здесь даже не научим вас замечать следы.

Чего хотят хакеры?

Взломанный компьютер — очень полезная вещь в доме, так что давайте начнем с рассмотрения того, как это сделать и что с ним делать.

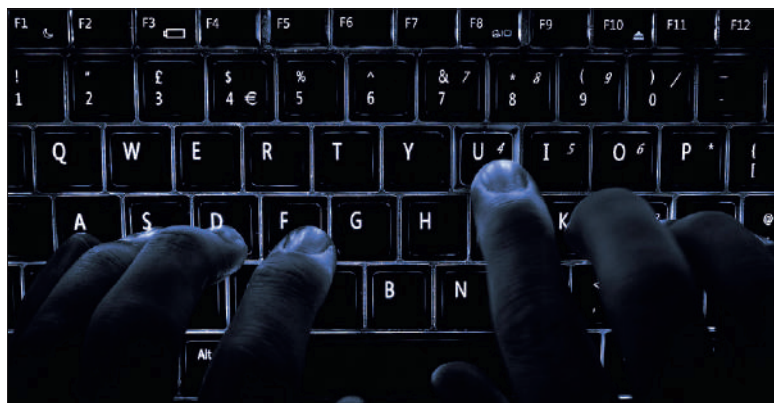
Если компьютер взломан, это нередко делается через парадную дверь. Или выводили чей-то пароль, или компьютер был неправильно настроен таким образом, что гостям позволялось делать намного больше, чем следовало бы. Пароль можно похитить с помощью клавиатурного шпиона, социального инжиниринга или потому, что его использовали на взломанном сайте (так называемые буферы паролей найти легко, если знать, где искать). Обычные ошибки неправильной настройки включают оставление учетной записи по умолчанию открытой и настройку чересчур щедрых разрешений для файлов и сервисов.

Будь это единственный вид взлома, жизнь была бы немного проще, но, конечно же, это не так. Часто программу или сервис, работающий на компьютере, вынуждают сделать то, чего она делать не должна, или осуществить взлом определенным способом, который дает злоумышленнику доступ к недопустимому (эскалация привилегий), шанс запустить что угодно (выполнение произвольного кода), или еще что-то злокозненное. Хакер, на которого в 1986 г. охотился наш вдохновитель, Клиффорд Столл, использовал дыру в программе *movemail*, части *GNU Mailutils*, которая обеспечивала доступ суперпользователя к компьютеру хоста и к остальным системам Национальной лаборатории Лоуренса Беркли [Lawrence Berkeley National Lab] — эскалация привилегий самого худшего типа.

Атаки могут быть направлены на человека или организацию, а могут быть неизбирательными. Когда выходит пилотная версия кода [Proof of Concept (PoC)] по новой уязвимости, ее применение в качестве оружия — лишь вопрос времени. На сайтах вроде Shodan можно получить список компьютеров с уязвимостями — золотое дно для хакеров-дилетантов, заводчиков ботов и всех прочих жаждущих нарушить закон. При уязвимости типа исполнения удаленного кода злоумышленник прилагает к коду эксплойта некую целевую нагрузку (прикрываясь какой-либо дымовой завесой по методу обфускации [obfuscation — англ. замутнение], если он достаточно умен). Если всё идет хорошо (или плохо, если это ваш компьютер подвергается атаке), то на удаленном компьютере запустится код. На домашнем компьютере этот код может оказаться клавиатурным шпионом или иным шпионским ПО. На сервере святым граалем является реверсивная оболочка [reverse shell], где целевой компьютер подключается к компьютеру злоумышленника и дает возможность запускать команды терминала.

Платите, а не то...

Атаки с использованием программ-вымогателей (когда файлы шифруются и за дешифрацию требуют выкуп в биткоинах) за долгие годы доказали свою прибыльность. Однако прошлогодние атаки WannaCry (которые подкосили Национальную службу здравоохранения Великобритании) собрали около \$140 000. Это не очень много, с учетом того, что заражено было 200 000 компьютеров. Атака могла быть хуже, если бы не действия британца Маркуса Хатчинса [Marcus Hutchins] (он же — MalwareTech). К сожалению, предыдущее исследование Маркусом вредоносного ПО привело к тому, что ему предъявили обвинения в США, где его и взяли в прошлом году после посещения



конференции по безопасности. В последнее время наметилась тенденция убирать посредника в виде конечного пользователя и напрямую устанавливать программу майнинга криптовалюты (cryptojacking). Вследствие анонимности валюты Моно и тому факту, что выгодно майнить без дорогостоящего оборудования, она стала любимой валютой такого рода атак. В августе этого года около 200 000 роутеров в Бразилии оказались зараженными кодом Coinhive.

Синяя подсветка и чистые ногти крайне важны для каждого уважающего себя хакера.

» ХАКЕРСТВО С РАЗМАХОМ

Наметить целью одну машину или сеть, конечно, хорошо, но некоторые люди (а то и государства) мыслят масштабнее. Как минимум в одном случае в прошлом году большие порции интернет-трафика (принадлежащего крупным уважаемым компаниям, таким как Facebook, Apple и Google), были перенаправлены через российские сети. О таких взломах протокола пограничной межсетевой маршрутизации [Border Gateway Protocol] (BGP) предупреждали давно, поскольку BGP была изобретена главным образом как временная мера. Интернет — это сеть, состоящая из сетей, так называемых Анонимных Систем [Autonomous Systems], и все они предназначены для того, чтобы сообщать о своих схемах пиринга и возможностях соединения открыто и честно, чтобы можно было быстро и эффективно маршрутизировать трафик. Однако какой-то конкретной защиты от неверного использования этой системы нет, и сайт BGPmon (<https://bgpmon.net>) регулярно сообщает об аномальных извещениях от роутеров. BGP — штука сложная, так что многие из них являются результатом человеческой ошибки, но за остальными может скрываться мрачная история.

В мае 2018 г. обнаружилось, что вредоносная программа под названием *VPNFilter*, заразила более полмиллиона роутеров в домах и малых офисах. Анализ вредоносного ПО показал, что оно способно проходить через брандмауэр, шпионить за трафиком и даже блокировать роутеры (возможно, чтобы воспрепятствовать экспертизе). Оно использовало известные уязвимости, которые стоило бы залатать провайдерам оборудования/интернет-провайдерам, хотя и пользователям тоже стоит принять на себя часть ответственности. *VPNFilter* вносит вредоносный контент в web-страницы и пытается шпионить за HTTPS-соединениями через SSL stripper. Сочетание широко распространенных инфекций типа *VPNFilter* и крупномасштабных взломов BGP рисуют леденчащее зрелище того, насколько хрупка используемая нами инфраструктура.



Орудия труда

Есть масса инструментов FOSS, способных помочь как профессиональному хакеру, так и любителю — мы покажем вам наши любимые...

Опытный хакер использует огромное количество инструментов для рекогносцировки, теста на вторжение, использования и хищения данных. Он может получить доступ к эксплойтам, от которых нет заплат, или коротать время, создавая собственные коды с целевой нагрузкой. Хакеры могли бы получить доступ к огромной сети из взломанных компьютеров (botnet) и использовать ее для DDoS-атак или как цепочку прокси, чтобы укрыться за ними.

С другой стороны спектра находятся любители, которые рыщут в Интернете в поисках готовых эксплойтов (процесс, который, в зависимости от критериев поиска и того, где вы нажимаете, может выйти боком), в конце концов изловчатся запустить Kali Linux и затем бомбардируют какие попало компьютеры эксплойтами. Не будьте хакером-любителем: это выглядит дурно и может вовлечь вас в кучу неприятностей. Однако познакомьтесь с используемыми инструментами — и вот вам подборка часто используемых программ, которые помогут вам освоить азы.

Wireshark

Wireshark захватывает пакеты «из сети», чтобы вы могли спокойно изучить их со всеми удобствами GUI. Очевидно, что если вы в оживленной сети, то трафик будет плотным, поэтому *Wireshark* позволяет делать фильтрацию по компьютеру или протоколу. Люди часто сталкиваются с трудностями, начиная использовать *Wireshark* (и другие инструменты захвата пакетов), поскольку для выноживания сетевого трафика необходимы особые права. Не запускайте *Wireshark* от имени root, это плохая идея. Вместо этого добавьте своего пользователя в группу *wireshark* с помощью `gpasswd -a youruser wireshark`, затем выйдите и снова войдите. *Wireshark* использует привилегии для запуска `dumpcap` с помощью инструмента `setuid` в фоновом режиме. Это намного безопаснее, чем запускать всё приложение как root.

Новички в сети могут почувствовать себя неуверенно, увидев, что весь трафик, подключенный к одному коммутатору, видимый, но именно так работает сеть. Трафик, не предназначенный для конкретного хоста, тихо игнорируется, но с помощью *Wireshark* мы можем его изучить. Пароли и номера кредитных карт, которые вводятся на сайте, должны шифроваться через HTTPS — собственно говоря, в наше время это должно делаться с большей частью web-трафика. Но вы никогда не знаете, чем чреват обычный перехват пакетов. Если вы жаждете сунуть куда-то нос и выяснить, какие сайты посещают ваши знакомые, перейдите в раздел *Name Resolution* [Разрешение имен] и познакомьтесь с опцией «Resolve network addresses [Разрешение сетевых адресов]».

Wireshark является основой атак на сети Wi-Fi, например, настройки фальшивой точки доступа Wi-Fi в кофейне и запуска атаки обдирания [striping] SSL. Старые атаки на шифрование WEP часто зависели от перехвата и воспроизведения пакетов запросов ARP, однако шифрование WEP больше никто не использует. *Wireshark* также полезен для изучения поведения проприетарных приложений, чтобы увидеть, к кому они обращаются и (в некоторых случаях) что именно они сообщают.

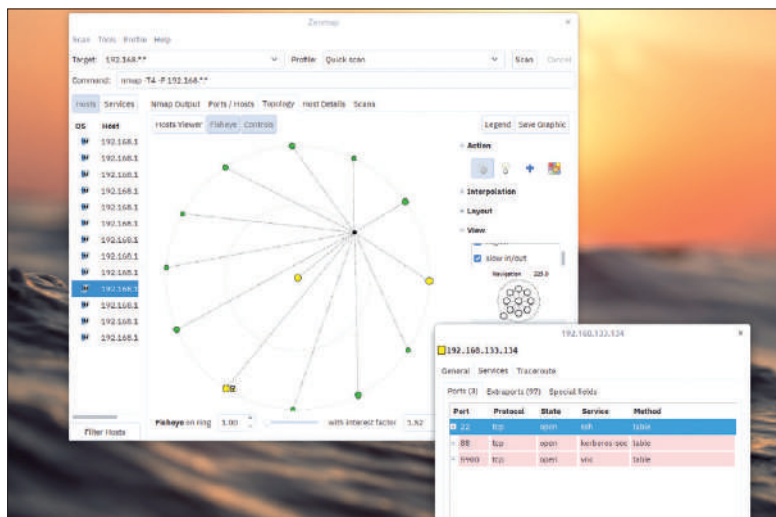
John the Ripper

Давным-давно шестизначный пароль считался безопасным, тем более, если он содержал заглавные и маленькие буквы, знаки пунктуации и не основывался на слове из словаря. Эти правила не совсем устарели (восемь символов почти приемлемы, если там есть спецсимволы), и многие до сих пор используют безнадёжно слабые пароли для защиты своих данных. *John the Ripper* — инструмент для взлома паролей, который делает этот факт кристально ясным.

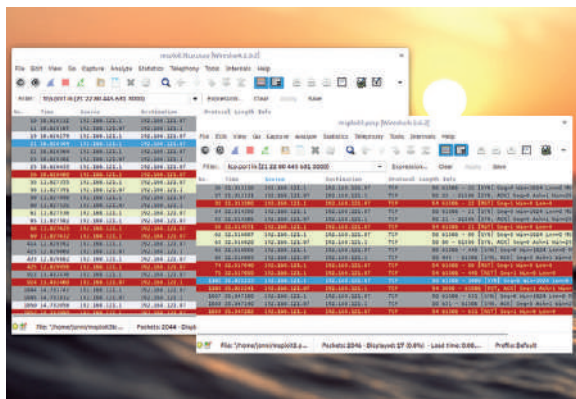
Было бы глупо хранить пароли в открытом доступе, так что обычно происходит следующее: пароль проходит через подводящую функцию хэша (например, SHA256), и хранится именно результат этого действия. Хэш-функции полагаются на определенные математические свойства (хэш не должен быть легко реконструируемым), и когда пароль вводится, вычисляется его хэш и результат сравнивается с тем, что уже хранится. Весьма маловероятно, чтобы у двух паролей был одинаковый хэш, так что если хэши совпадают, доступ разрешается.

Когда компанию взламывают, ее базы данных часто похищают и продают — или отдают бесплатно. Это дает множество хэшей, которыми может воспользоваться взломщик паролей типа *John the Ripper*. Некоторые системы блокируют вас после определенного количества неудачных попыток ввода пароля, но эти правила не применяются, если у вас есть украденная база данных (чтобы узнать об «онлайн-взломе» паролей, ищите Hydra). *John the Ripper* также способен задействовать возможности GPU для проверки нескольких тысяч паролей в секунду.

Помимо случайных комбинаций символов, *John the Ripper* может использовать словари, что очень помогает в процессе взлома паролей. Мало того, *John the Ripper* может использовать правила сочетания словарных слов друг с другом и со случайными символами, имитируя процесс создания паролей особо



Беглое сканирование сетей издательства засекло несколько сервисов, которые не очень порядочный мог бы употребить злонамеренно (не будь этот некто постоянно в цейтноте по сдаче статей).



Полный скан TCP (слева) создает куда больше шума, чем установленный по умолчанию в Nmap скан SYN (справа). А еще очень полезны фильтры Wireshark.

одаренными. Например, популярно использовать заглавную букву в начале слова и ставить цифру в конце. В синтаксисе *John* это правило записано как `sAz[0-9]`. Проще некуда. Современный подход к созданию паролей включает комбинирование словарных слов для создания длинного пароля, а не попытки умничать, случайно сочетая заглавные буквы и символы. Похожий подход указан в стандарте BIP39 по созданию паролей для кошельков Bitcoin, только там каждое слово уникально обозначено первыми четырьмя буквами.

Kismet

Взлом беспроводных ключей либо тривиален — в случае давно устаревшего шифрования WEP, либо очень сложен — в остальных случаях. Однако иногда можно получить желаемое без ключа. В октябре 2017 г. было объявлено об атаке на WPA2 (применяемое большинством домашних роутеров), которая позволила третьей стороне перехватывать трафик в непосредственной близости к цели. Важные данные нужно шифровать через HTTPS, снизив опасность, однако факт тот, что практически всё оборудование Wi-Fi было (и, вероятно, многое до сих пор остается) уязвимым.

Прежде чем начать переживать насчет взлома сетей Wi-Fi, полезно было бы сначала их определить. Обнаружение карты беспроводных сетей на определенной территории известно как «вардрайвинг [wardriving]», и инструмент *Kismet* в этом помогает. Чтобы стать вар-драйвером, вам нужен модуль GPS и беспроводное устройство, которое ладит с Linux (оно должно поддерживать «режим мониторинга»). Возможно, вам также понадобится машина, в зависимости от исследуемой вами территории. Мы просто заставили Джонни побродить вокруг Башен издательства с Raspberry Pi. Как только будет собрано достаточно данных о сигнале, их можно конвертировать в файл `.kmz` и импортировать в Google Earth. Имеется также GUI-клиент, *Kismet*.

Nmap

Нередко первый шаг в определении целевого компьютера — посмотреть, какие порты открыты и какие сервисы на них работают. Есть множество инструментов сканирования портов, но *Nmap* — один из самых лучших. Он работает в командной строке, но в большинстве дистрибутивов предлагается также GUI (*Zenmap*), превосходный для визуального представления карты сетей. Результаты *Nmap* можно сохранить как XML и импортировать в другие инструменты для дальнейшего анализа, а также импортировать их, например, прямо в базу данных Metasploit, чтобы можно было попробовать разные атаки на разных хостах.

У *Nmap* достаточно опций, и в его map-странице 3000 строк, так что здесь мы не будем рассказывать обо всех. Однако, в духе

слогана Kali Linux «Будь потише — больше услышишь», давайте вполголоса поговорим о сканировании портов.

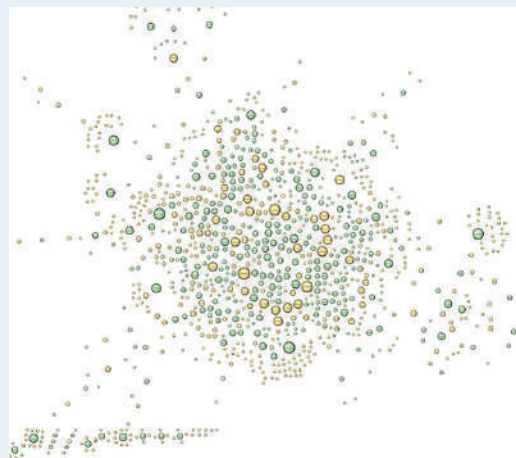
Сканирование стандартного TCP-порта включает трехстороннее квитирование (мы отправляем флаг SYN, и если порт открыт, цель шлет ответ SYN-ACK, и тогда мы посылаем ACK), и тогда (мгновенно) устанавливается TCP-сессия между сканером и целью. Если администратор целевого компьютера посмотрит журнал брандмауэра, он сразу увидит множество TCP-соединений от сканера портов.

Сканирование *Nmap* по умолчанию немного другое. Оно сбрасывает скан после ответа сервера, так что оставляет меньше следов. Этот сканер генерирует пакеты напрямую, не используя сокетов API, поэтому требует привилегий root. Вы можете увидеть различия между TCP-сканом и SYN-сканом на рис. вверху слева. SYN-скан не является невидимым, но передается меньше пакетов для получения той же самой информации (является ли порт открытым, закрытым или заблокированным).

» СОЦИАЛЬНЫЙ ИНЖИНИРИНГ

Атаки социального инжиниринга так же эффективны в получении привилегированной информации, как сложные эксплойты нулевого дня или тщательно подготовленные фишинговые атаки. Большинство людей сейчас уже в курсе насчет обыкновенных телефонных жуликов, представляющих техподдержку, когда для звонящего главное — получить удаленный доступ, установить клавиатурного шпиона и потом собрать банковскую информацию, пароли или адресную книгу для применения в последующих атаках. Однако возможны и другие формы атаки. Например, в начале августа 2018 г. сообщалось об атаке на сервера Reddit — через двухфакторную аутентификацию на основе SMS (2FA) злоумышленникам удалось пробиться в учетную запись администратора, частично посредством известной слабости сотовой сети и частично через атаку перенаправления защитного кода [Verification Code Forwarding Attacks] (VCFa). Отправляя с виду вполне законное сообщение, где пользователя просят прислать сгенерированный аппаратно ключ 2FA от провайдера, злоумышленники получают доступ.

Атаки социального инжиниринга тем эффективнее, чем больше известно о жертве. Преступники часто проводят немало времени, отслеживая значимую цель и оттачивая атаку. Эта практика именуется «китобойством [whaling]», в отличие от более стандартного «рыболовства [phishing]». В наше время практически все как-нибудь да присутствуют в Сети, даже если закрыли свои учетки в соцсетях. Просмотр открытых информационных ресурсов именуется разведкой по открытым источникам [Open Source Intelligence] (OSINT). Хорошая OSINT требует времени и усилий, но популярный *Maltego* автоматизирует этот процесс. Используя разные источники данных ('transforms'), от поисковой машины сервера Shodan до блокчейнов, постов в Twitter и баз данных GeolP, можно отследить все взаимоотношения.



Maltego создает графики, отображающие скрытые связи в открытых данных.

Среда Metasploit

Начав с ведущего инструмента тестирования на вторжения, взломайте свою же виртуальную машину. Устраивайтесь поудобнее, с чашечкой чая...

Мы считаем, что свобода информации об уязвимостях и экспериментальных эксплоитах — это, несмотря на противоположные аргументы, Хорошо. Однако если вы занимаетесь тестированием на вторжения (или мечтаете этим заниматься), то, к сожалению, вы вступили в жесткую гонку.

ЦЕЛЬ ОБНАРУЖЕНА

«Rapid7 выпустил специально разработанную виртуальную машину, Metasploitable 3, уязвимую ко всем возможным атакам.»

Код, созданный для доказательства концепции (PoC, Proof of Concept), вероятно, потребует настройки и установки зависимостей. Для уязвимостей, связанных с памятью, код также надо

скомпилировать. Вдобавок сперва здесь еще присутствует всякая волокита просеивания списков рассылки, чтобы раздобыть этот код. Разве не прекрасно было бы иметь своего рода библиотеку эксплоитов и своего рода среду для их унифицированного запуска? Что ж, дорогой читатель, такая вещь существует, и имя ей — *Metasploit*. Она имеет две редакции: премиальная *Metasploit Pro* (за которую надо платить, но у нее чудесный web-интерфейс) или бесплатная *Metasploit Framework*.

Если у вас хватает ОЗУ (live-диски используют его для хранения), вы можете установить *Metasploit Framework* в Kali Light с *Linux Format* DVD. Просто введите `apt install metasploit-framework`. Но, желая попробовать взломать VM *Metasploitable VM*, установите ее на другой компьютер, поскольку под хранилище ей нужно не менее 10 ГБ.

Если вы уверенный пользователь VM, вы можете спокойно настроить одну такую для запуска Kali (или версию Light из ISO на нашем DVD, или полную большую версию, которую можно скачать с www.kali.org/downloads), и еще одну для запуска *Metasploitable VM*. Альтернатива — добавить *Metasploit Framework* в свой любимый дистрибутив с помощью установщика и инструкций на <https://github.com/rapid7/metasploit-framework/wiki/Nightly-Installers>. Если вы не хотите использовать универсальный установщик и предпочитаете (по большей части) брать пакеты из репозитория Ubuntu/Debian, загляните на <https://kb.help.rapid7.com/docs/installingthe-metasploit-framework-on-ubuntu-linux>.

» СРЕДА METASPLOIT

После установки вы можете запустить *Metasploit* командой `msfconsole`. Режим здесь текстовый, что еще более приблизит вас к l33t hax0r. *Metasploit* содержит иерархию модулей, распределенных по категориям: `auxiliary` [вспомогательные] (сканеры портов и fuzzers), `exploits` [эксплоиты] (код для использования уязвимых систем) и `payloads` [целевые нагрузки] (то, что исполняется на удаленном компьютере, куда проник эксплоит). Можно добавить сторонние модули в директорию `~/msf4/modules`. Модули написаны на языке Ruby, и тысячи их доступны онлайн. В основном тексте вы увидите, как запустить *Nmap* для сканирования портов, но для этого также имеется модуль *Metasploit*. Его использование поможет разобраться, как применять *Metasploit* в целом. Сначала загрузите модуль, затем давайте рассмотрим опции и просканируемся (в конце концов, гигиена начинается дома):

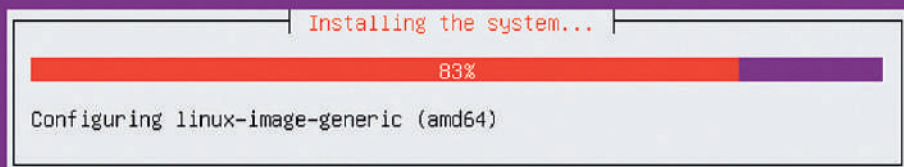
```
> use auxiliary/scanner/portscan/tcp
> show options
> set RHOSTS 127.0.0.1
> run
```

Авось, не будет нежелательных сюрпризов, так что продолжим наши попытки взлома.

Запустите Metasploitable VM

Помимо попыток захвата этого старого пыльного компьютера с Windows XP, который вы много лет не включали после перехода на Linux, Rapid7 выпустил специально разработанную виртуальную машину, *Metasploitable 3*, уязвимую ко всем возможным атакам. В нашем предыдущем материале о хакерстве в **LXF225** наш главный хакер Нейт Дрейк использовал ее предшественницу для демонстрации своих умений. Новая версия использует *Packer* и *Vagrant* для создания виртуальной машины динамически из репозитория GitHub, так что это немного сложнее, но мы полагаем, дорогой читатель, что вы справитесь.

Учтите, что для создания образа машины инструкции требуют не менее 6,5 свободных ГБ, так что вы не сможете сделать это из среды Kali Light live, по крайней мере, если не добавите путей локального хранения. Мы покажем вам, как скомпилировать виртуальную машину `libvirt` (совместимую с `Qemu`) в Ubuntu, но очень легко создать образы *VMware* и *Virtualbox*, причем из любого



Наблюдать, как *Packer* автоматически устанавливает и оборудует машину Ubuntu, одновременно жутко и успокаивающе. Прямо как программу BBC One воскресным вечером.

дистрибутива. Первый шаг — установить *Git*, *Packer* и *Vagrant*, и затем добавить нужные плагины к последнему.

```
$ sudo apt install git packer vagrant
$ vagrant plugin install vagrant-reload
$ vagrant plugin install pkg-config
$ vagrant plugin install vagrant-libvirt
```

Затем клонируем исходники *Metasploitable3* (около 250 МБ) с Github:

```
$ git clone https://github.com/rapid7/metasploitable3.git
```

Следующая стадия — создать базовую VM с помощью *Packer*. Если вы предпочитаете создать образ *VirtualBox*, задайте параметр `--only=virtualbox-iso` в команде ниже. По умолчанию процесс создания использует `/tmp` для хранения временных файлов. Поскольку `/tmp` обычно настроен на использование хранилища `tmpfs` (в памяти) и есть вероятность его переполнения, будет разумно вместо этого указать директорию на диске, где места больше. Мы будем использовать `/var/tmp`:

```
$ cd metasploitable3/
$ TMPDIR=/var/tmp packer build --only=qemu ./packer/templates/ubuntu_1404.json
```

Появится окно *Qemu*, но не поддавайтесь порыву что-нибудь туда ввести. *Packer* сделает всё автоматически (после примерно 20-секундного ожидания). Он запустится, если вы начнете выбирать опции, поэтому просто усадьтесь поудобнее и посмотрите кино, заварите себе еще чашечку чая и вообще погуляйте.

Когда всё будет готово, мы сможем создать образ *Vagrant* с помощью

```
$ vagrant box add ./packer/builds/ubuntu_1404_libvirt_0.1.12.box
--name=metasploitable3-ub1404 --provider=libvirt
```

Образы *Vagrant* хранятся в `~/vagrant.d/boxes`, но если в вашей домашней директории не слишком много места (обычно требуется 2 ГБ), можно это изменить, настроив переменную `VAGRANT_HOME`.

И, наконец, запустим машину:

```
$ vagrant up ub1404 --provider libvirt
```

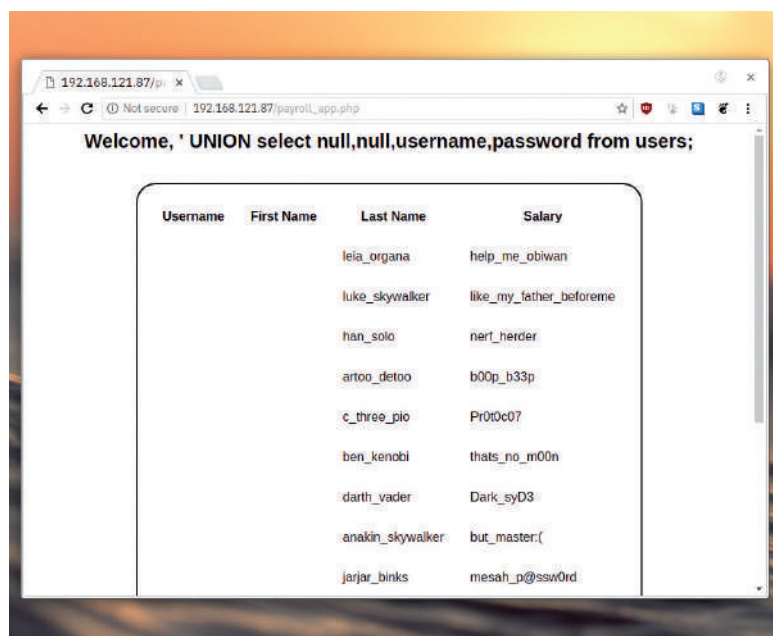
Вас попросят ввести пароль, чтобы можно было изменить `/etc/exports` на машине хоста. Однако мы столкнулись с проблемой — виртуальная машина не смогла смонтировать этот экспорт NFS (она жаловалась, что `rpc.statd` не запускается, в чем и была проблема: мы обнаружили несовместимость с более новой начинкой NFS). Если у вас случится то же, запустите `vagrant destroy ub1404` и отредактируйте `Vagrantfile`, добавив строку `config.vm.synced_folder ".", "/vagrant", disabled: true` и снова запустите команду `vagrant up`. Машина должна появиться в Virtual Machine Manager (если вы используете этот удобный GUI libvirt; если же нет, вам придется применить какую-то другую хитрость, чтобы найти ее IP-адрес), и вы должны быть в состоянии войти в консоль с именем пользователя и паролем `vagrant`. Здесь вы можете найти IP-адрес VM с помощью обычной `ip a`, после чего выйти из консоли.

На машине хоста (или в VM Kali, если вам хочется удвоить виртуальность) запустите *Nmap* (установите его, если вы этого еще не сделали), и просканируйте этот IP (допустим, это `192.168.1.100`) на всех портах командой

```
$ nmap -p0-65535 192.168.1.100
```

Вы должны обнаружить ряд доступных сервисов, включая SSH HTTP, базу данных SQL и сервер IRC, но не только. Работают и другие сервисы, но они доступны лишь локально. Чтобы увидеть их, надо зайти через SSH на целевую VM и снова запустить `netstat -ltun`.

Впрочем, сначала посмотрим, что можно взломать «удаленно». Направьте свой браузер на `http://192.168.1.100`. Привет, это что? Вы должны увидеть список директории, который позволяет вам перейти в чат webarr (где обнаружится Para Smurf), установ-ку Drupal, приложение PHP payroll и панель PHPMyAdmin...



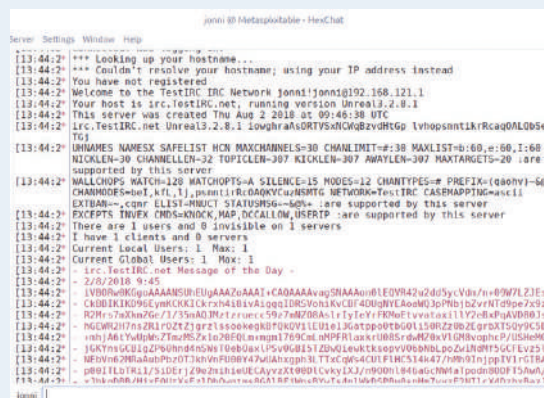
В 2018 г. SQL-инъекция жива-здоровая. Не использовать ли эти пароли заново, где-нибудь на виртуальной машине *Metasploitable3*?

» КАКОЙ IRC ВЫ ВИДИТЕ?

У нас здесь мало места, и мы не хотим портить сюрприз, поэтому предоставим большую часть обнаружения уязвимостей и их использование вам. Но чтобы разогреть ваш интерес, давайте рассмотрим демон IRC, который работает на порте 6697. Побродив по *Metasploitable VM* (в директории `/opt/unrealircd/`), вы обнаружите, что у нас работает версия Unreal IRC демона серии 3.2. Это похоже на недоуверительство, но если вы хотите всё сделать как надо, соединитесь с виртуальной машиной с помощью любого старого клиента IRC, и вы увидите, что у нас работает 3.2.8.1. Так уж вышло, что именно эта версия шла с лазейкой (CVE-02010-2075), и так уж вышло, что у *Metasploit* есть модуль эксплойта для данной лазейки. Чтобы запустить его и получить реверсивную оболочку, сделайте так (заменяя IP-адрес на нужный):

```
> use exploit/unix/irc/unreal_ircd_3281_backdoor
> set RHOST 192.168.1.100
> set RPORT 6697
> run
```

Теперь вы можете исполнять команды оболочки и исследовать систему, как при обычном SSH-соединении. Запустив `whoami`, вы увидите, что демон работает в `boba_fett`, именно этот уровень доступа у нас есть. Это не полнофункциональная оболочка, поэтому мы не можем скакать по директориям, и `Boba` не имеет доступа `sudo`, но с этого, по крайней мере, можно начать.



С помощью Hex-Chat мы можем распознать номер версии UnrealIRCd, работающего на виртуальной машине *Metasploitable3*.



Как всё ломается

И под конец давайте рассмотрим странные и чудесные способы вредоносного употребления ошибок программ ради большего вреда.

С того момента, когда люди начали подключать свои приложения PHP к базам данных SQL, существуют атаки SQL-инъекций. Они используют всю введенную пользователем информацию, обычно в web-формах, чтобы злоумышленник мог запустить произвольные SQL-запросы. Классический пример — ввести нечто вроде `‘; DROP TABLE users;` в поле имени пользователя. Если код PHP этой формы создал запрос SQL как `$sql = “SELECT username from users where username = ‘$user’;”`, то, заменив переменную `$user` на наш вредоносный ввод, в конечном итоге мы получим два запроса SQL по цене одного:

```
SELECT username from users where username = ‘; DROP TABLE users;
```

и наша таблица “users” исчезнет из-за невычищенного ввода. Причина, по которой это работает (на самом деле, больше оно не должно работать нигде, потому что PHP уже не разрешает объединять операторы SQL подобным образом), в том, что нам было разрешено поставить во вводе одни кавычки. Даже если вам не разрешается объединять команды PHP, вы можете ввести нечто вроде

```
‘ OR 1=1 UNION SELECT username, password from USERS;
```

применив для объединения результатов запросов оператор **UNION**, что тоже работает. Повторяющаяся часть `1=1` означает, что мы выбираем из нашей таблицы всех пользователей, а затем выставляем напоказ их пароли (никто в здравом уме не станет хранить пароли в виде простого текста, однако в Интернете хватает полоумных). Вы найдете примеры из реальной жизни для этого в виртуальной машине *Metasploitable*, которую мы рассматривали ранее [Ред.: — Полагаю, вы увидите, что это всё же примеры из виртуальной жизни].

Существование такого рода уязвимостей в наши дни мало чем оправдано, однако они упорно продолжают встречаться. Есть разные способы отфильтровать спецсимволы или избежать их употребления в переменных, и в любом случае плохо создавать операторы SQL простым соединением строковых переменных. О чем стоит помнить.

Уязвимости памяти

Возможно, вы слышали такие выражения, как «переполнение буфера», «использование освобожденной памяти» и «повреждение стека». Все они относятся к проблемам памяти, разъяснить которые будет сложно без понимания того, как программы и используемые ими переменные распределяют и используют память. Когда вы пишете программу на языке скриптов, например, на Python или PHP, управление памятью полностью передается интерпретатору, и пользователь остается в счастливом неведении той кошмарной административной работы, которая ведется за сценой.

Если мы перейдем на C, нам придется немного подрасти и взять на себя ответственность за управление памятью. Сначала

» SPECTRE И MELTDOWN

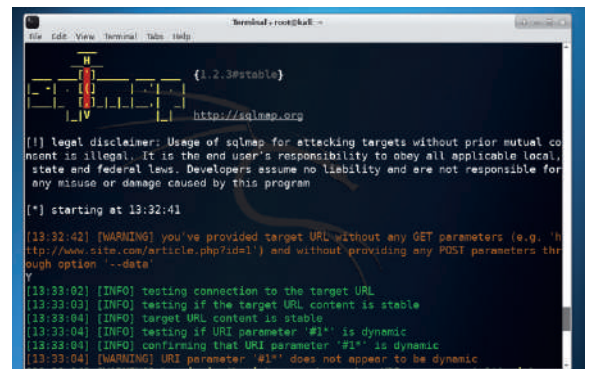
В начале января, как раз когда редакция **LXF** приходила в себя после зимних каникул, распространилась история о далеко идущих уязвимостях CPU. Пока мы дружно терли глаза, не будучи уверенными, что очнулись после праздничных излишеств, начали появляться подробности: чипы от Intel, AMD и ARM были 20 лет под воздействием, простого лечения нет и никаких заплат тоже. Spectre и Meltdown, как их называли, не один месяц держались в секрете, пока исследователи пытались создать отладку.

Spectre использует функцию CPU, известную как прогноз ветвления, которая наудачу исполняет ветви кода, даже несмотря на то, что результат может не потребоваться. Суть в том, что если эти результаты не требуются, их можно отбросить с минимумом лишних усилий (прогноз планируется так, чтобы не мешать другим вычислениям), а если эти результаты окажутся востребованными, тогда мы выигрываем. Эффект Spectre в том, чтобы подтасовать этот прогноз ветвления, и в том, чтобы посредством тщательного планирования последующих вычислений (чтобы увидеть, кэшированы ли их результаты, уже будучи наудачу выполненными) была получена потенциально привилегированная информация. Поэтому призыв (spectre) с веткой (ветвлением) в качестве логотипа весьма вдохновляющий [Ред.: — Но ему далеко до наших коллажей Тукса на обложке, легендарных и сомнительных с юридической точки зрения].

После этого было обнаружено несколько разновидностей Spectre, и широко принято считать, что заплатки к компиляторам и ядрам даже обновления микрокода и прошивки — всего лишь частичный обходной путь. Избавит от этой слабости только полный редизайн оборудования. Выполнение наудачу или загодя были жизненно важными, чтобы добиться максимально возможной скорости работы процессоров, однако за эту скорость пришлось расплачиваться.



Хороший логотип — одна из возможностей повысить знания об уязвимостях, но мы пока не уверены, что лучшая.



SQLMap стал главным орудием охоты за SQL-инъекциями. Как и все полезные орудия, его можно употребить и во зло.

это может показаться разумным: вдруг нам понадобится переменная, содержащая чье-то имя, а имена бывают длинными, и нам придется отвести на этот буфер 63 символа (что требует наличия 64-байтного буфера, с учетом нулевого символа окончания, если только речь идет не о кодировке Unicode).

Однако, если мы считываем введенную пользователем информацию с помощью чего-то небезопасного, например, функции `scanf()` без указателя максимальной ширины, тогда злонамеренный пользователь или даже просто пользователь с длинным именем может доставить больше данных, чем вмещает буфер, и может произойти нечто плохое. Если в смежном буфере памяти хранилась другая переменная, то этот буфер частично затрется последней частью введенной информации, что может или повредить программу, или заставить ее нести околесицу. Простая проверка границы остановит эту беду, да и вообще довольно глупо оставлять введенную пользователем информацию непроверенной (особенно это касается PHP); но суть в том, что в C и C++ всё это отдано на откуп программисту, и во многих случаях далеко не очевидно, что можно использовать со злым умыслом и где; а даже и будь это очевидно, накачанному кофеином кодеру всё равно легко такое упустить.

Есть также особые виды переполнения буфера. При загрузке программы определенная часть памяти отводится под стек вызовов. Эта структура данных отслеживает (помимо всего прочего) место, откуда программный поток должен вернуться к следующему выполнению каждой активной подпрограммы (адрес возврата). И если подлый пользователь сможет это скомпрометировать, то сможет и направить программный поток в выбранное им самим место, возможно, туда, куда он заранее внес вредоносный код облобочки, и случится ужасное.

Эксплуатация списка вызовов является основой возвратно-ориентированного программирования (ROP). Операционные системы и оборудование уже какое-то время включают защиту от переполнения буфера — примечательнее всего возможность отмечать память, где данные могут быть записаны как неисполняемые. Вы можете назвать ROP методом, дружелюбным к окружающей среде, поскольку вместо внедрения вредоносных данных в стек он использует данные, которые уже там есть. Манипулируя только адресом возврата так, чтобы выполнялась некая полезная часть наличного кода (известная как «гаджет») и сцепляя эти гаджеты вместе, можно превратить код-оригинал в нечто зловредное.

В 1988 г. червь Морриса [Morris worm] — возможно, первый в мире интернет-червь — используя несколько уязвимостей, распространялся со скоростью лесного пожара. Одной из уязвимостей было переполнение буфера в популярной утилите `finger`, проверявшей, кто вошел в систему на определенном компьютере. Червь Морриса был задуман как академическое упражнение по измерению количества компьютеров в Интернете, но, к сожалению, оказался настолько вирусным, что вместо этого заразил немало этих самых компьютеров. Наш бесстрашный герой Клиффорд Столл, ясное дело, участвовал в команде зачистки (если вы не понимаете нашего безграничного уважения к этому человеку, почитайте его книгу *Яйцо кукушки* [The Cuckoo's Egg] или посмотрите документальный фильм на YouTube).

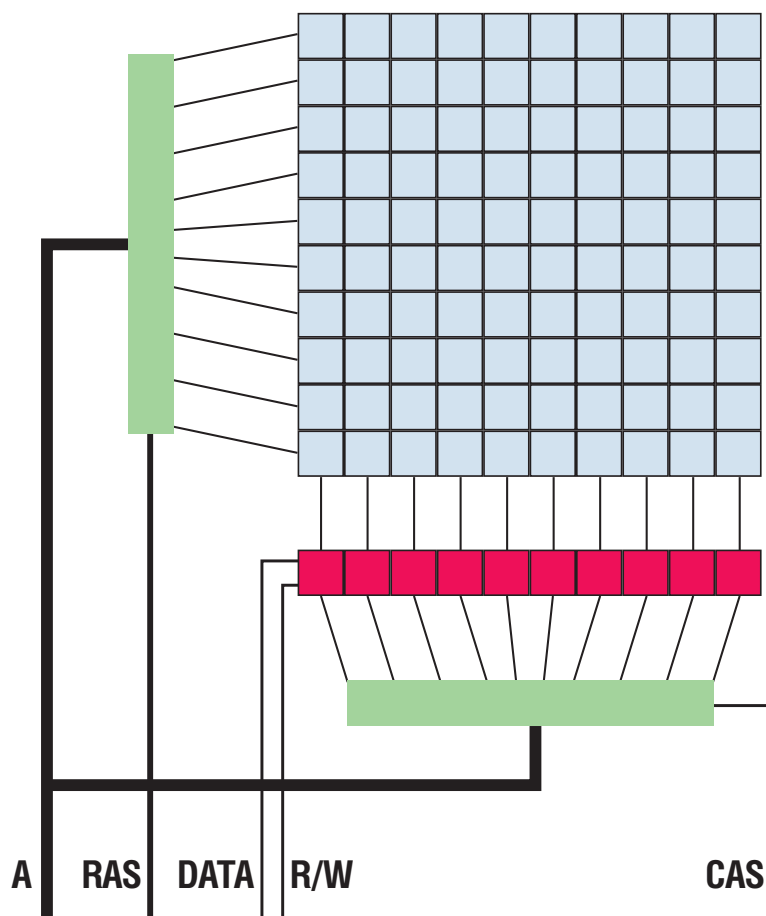
Ошибка Heartbleed (печально известная не только тем, что вынудила миллионы пользователей eВau сменить пароли, но и тем, что это — первый вирус с собственным логотипом) применяет другой тип атаки на буфер (перечитывание буфера), эксплуатируя популярный пакет OpenSSL, который обеспечивает безопасность львиной доли Интернета. В этом случае на сервер отправляется сообщение 'heartbeat' вместе с параметром длины, который якобы является длиной сообщения. Затем предполагается, что сервер ответит тем же сообщением, поддерживая жизнеспособность соединения, отсюда и 'heartbeat' [Сердцебиение].

К сожалению, до создания заплат не выполнялось ни одной проверки, чтобы выяснить, соответствует ли длина сообщения указанной длине. Вредоносный клиент мог отправить сообщение и обеспечить намного большую длину сообщения. Ничего не подозревающий сервер отвечает, возвращая исходное сообщение плюс то, что оказалось в памяти OpenSSL, находящейся рядом с местом хранения сообщения. Сюда могли входить имена

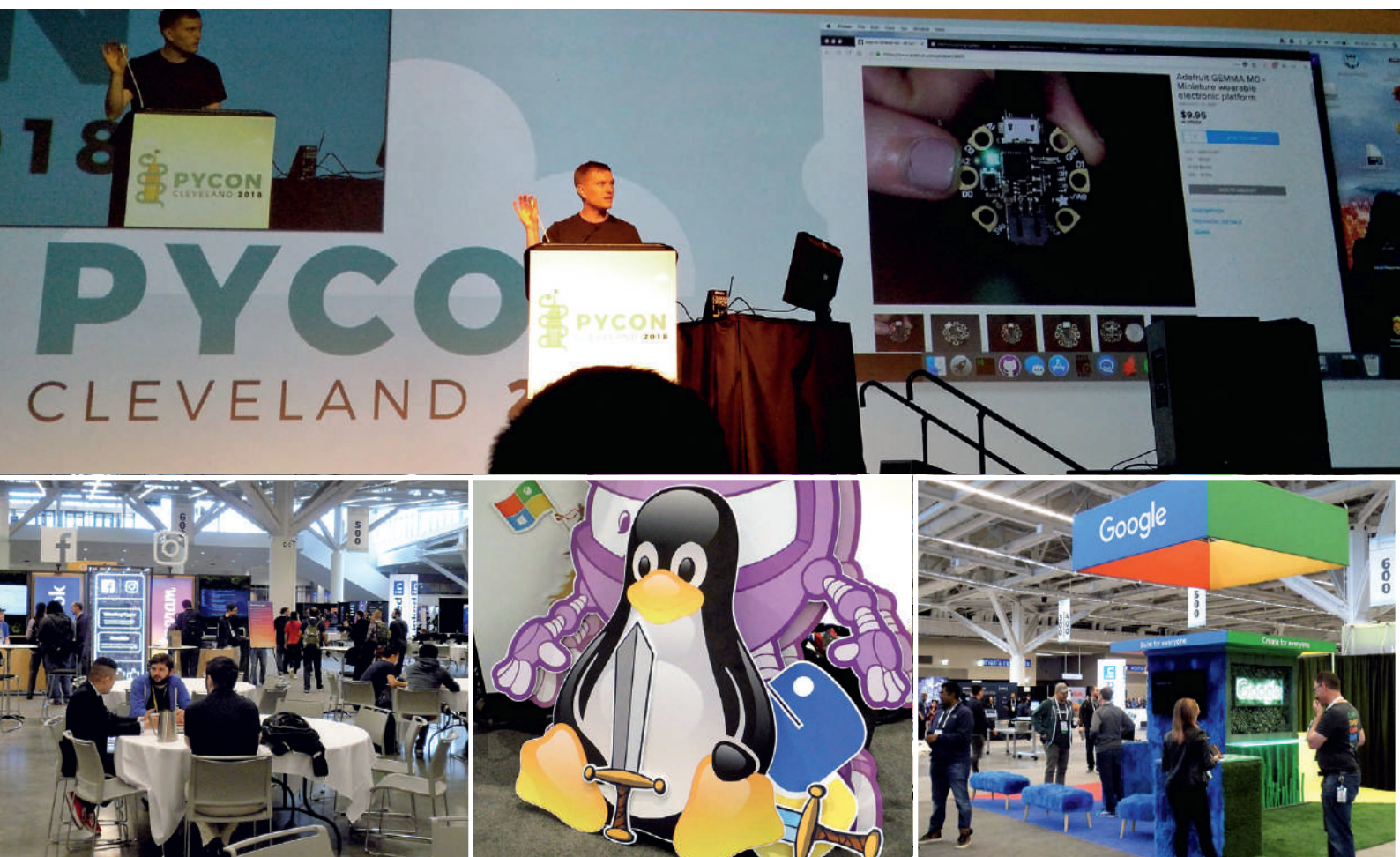
БЛАГИЕ НАМЕРЕНИЯ И ВСЁ ТАКОЕ

«Червь Морриса был разработан как академическое упражнение по измерению количества компьютеров в Интернете, но оказался настолько вирусным, что вместо этого заразил немало этих самых компьютеров.»

пользователей, пароли, ключи сертификатов — короче говоря, достаточно, чтобы завладеть системой или ее пользователями. Из-за способа ввода переменной длины злоумышленник мог запросить 64 КБ за раз, и нет никаких ограничений на частоту отправки такого запроса. Так что очень скоро он сможет наткнуться на нечто полезное — это просто до тривиальности (например, через команду `strings`), и осуществит поиск по огромным двоичным файлам в поисках пригодных для сбора данных. **LXF**



Атака Rowhammer использовала новый побочный канал для манипуляций памятью. Увы, у нее нет стильного логотипа, так что предлагайте свой вариант — естественно, сделанный в GIMP.



PYCON 2018

Лес Паундер приехал на крупнейшую конференцию Python в мире и обнаружил, что язык программирования — это нечто большее, чем сумма его частей.

Python — сообщество с прикрепленным к нему языком программирования. Эта цитата из вступительного выступления на конференции PyCon в нынешнем году подводит итог Python. Пусть это язык программирования, применяемый разной публикой — от простых любителей до космических туристов, но для множества людей Python значит гораздо больше. Для 4260 участников, которые посетили мероприятие в Кливленде, штат Огайо — определенно.

Конференция PyCon, проходившая в конгресс-центре Хантингтона, работала несколько дней и затронула массу разных аспектов. Первый день был ознакомительным для вновь прибывших: обстановка немного тише, и для разработчиков открыты помещения, чтобы они могли встретиться

и обсудить свои проекты. Также был проведен ряд семинаров, где разработчики могли освоить новые навыки, которые они могли бы прихватить себе на рабочее место.

Когда открылась основная конференция, мы увидели переговоры, семинары и однодневный образовательный саммит, во время которого преподаватели со всех США и не только смогли встретиться и обсудить идеи.

Но что бросалось в глаза повсюду на конференции, так это чувство общности. У PyCon есть кодекс поведения — набор правил, которые должны соблюдать делегаты. Тем не менее, люди, с которыми я общался, были самыми обаятельными и открытыми, каких я только встречал. Разговор с хозяевами стендов в выставочном зале отнюдь не был упражнением в навязанных

продажах. Они были там, чтобы привлечь людей и восславить язык, позволивший образованию познакомиться новое поколение с кодированием и внедривший в жизнь последние достижения науки о данных. Что подводит нас к еще одной цитате из конференции: «Python — второй из лучших языков». Это его сила, а не слабость, поскольку Python можно использовать для быстрого прототипирования нового проекта или идеи.

Мощь Python заключается не в самом языке, а в его способности стать инструментом для решения любой задачи. От скромных трюков с Raspberry Pi до серьезных научных проектов, включающих машинное обучение, Python уместен повсюду и вдохновляет людей всех профессий на написание кодов — для удовольствия, а возможно, даже для получения прибыли.

Ива Джодловска

[Ewa Jodlowska]

LinuxFormat: Привет, Ива! Пожалуйста, не могли бы ли вы рассказать читателям *Linux Format* больше о себе.

Ива Джодловска: Я директор по операциям в Python Software Foundation, и одна из моих главных задач — контроль за организацией и проведением конференции PyCon.

LXF: Таким образом, PyCon — это ваше «дитя»?

ИДж: Да, однако я в процессе того, чтобы отпустить его. Я перехожу на надзорную роль и не выполняю всю оперативную работу сама.

LXF: Что PyCon значит для вас?

ИДж: Множество вещей, на самом деле. Это значит — семья, ведь я занимаюсь этим так долго. Но кроме того, PyCon определенно означает для меня «сообщество». Так что, когда мы говорим обо всём напряженном планировании PyCon, которое обычно идет за недели до конференции, я замечаю в социальных сетях, что все ощущают азарт по поводу участия в PyCon, и это поднимает мне настроение, потому что я с головой тону в работе и стараюсь, чтобы всё было сделано, и вот вижу все эти радостные и восторженные твиты. Приятно знать, что сообщество любит бывать здесь.

LXF: Во время основных докладов мы слышали, что «Python — это сообщество, к которому прикреплен язык программирования». Стал ли язык Python лучше благодаря своему сообществу?

ИДж: Я уверена, что это более открытый язык благодаря духу сообщества, стоящего за ним. Может быть, не для всех, но для подавляющего большинства. Именно этого они хотят от языка программирования.

LXF: Мы заметили, что Python вдобавок объединяет такие сообщества, как PyLadies и PyCon Charlas [Charlas — исп. «переговоры», — прим. пер.], испаноязычное мероприятие Python в рамках PyCon. Разнообразие на PyCon явно приветствуется...

ИДж: Я рада, что вы упомянули PyCon Charlas — мы это только что запустили, как раз в этом году. Мы получили несколько запросов от суб-сообществ в Python, желающих быть представленными на PyCon. Идея-то замечательная, но мы не можем вместить всех. Вряд ли кто-то хочет продлить PyCon с текущих семи до девяти дней.

Поэтому мы создали программу инкубатора. Оформляя документы на конференцию, люди представляли предложения о дополнениях, желательных на PyCon. Они либо стали бы постоянной частью PyCon, либо, возможно, через несколько лет доросли бы до отдельного

события. Из инкубатора и пришли к нам PyCon Charlas, и я надеюсь, что они будут и в следующем году, потому что, мне кажется, многие люди этому рады.

LXF: PyCon — это масштабная конференция.

Что нужно для организации мероприятия такого размаха?

ИДж: Это требует опыта, организованности и массы времени! Я бы рада сказать, что «мы потратили на это полгода — и вот оно!», но тратим мы годы — для гарантии, что PyCon состоится. Прошлым летом мы извели три-четыре месяца, выбирая места для PyCon 2020 и 2021. Здесь я их раскрывать не буду, но это показывает, как задолго мы начинаем.

Мероприятие представляет собой непрерывный процесс доработки. Вы видите, что делают другие конференции, и слушаете, чего хочет ваше сообщество от вашей конференции, и вы постоянно вносите коррективы — в пределах разумного. Я писала в блоге о том, что нужно для планирования PyCon. Мы начинаем с 17 городов и проводим их через несколько уровней отбора, исходя из потребностей конференции. Таким образом, от 17 мы пришли к трем городам, которые посетили. Отбор занимает время, и мы проводим всякие



Ива Джодловска — директор по операциям PSF. PyCon проходит благодаря ее руководству и многим волонтерам, которые жертвуют свое время.

LXF: Мы видим, что всё больше детей изучают Python. Насколько важен язык для образования во всём мире?

ИДж: Очень — особенно, если учесть, что PSF (Python Software Foundation) участвует во многих грантовых программах. Я вижу, что он оказывает свое влияние разными путями в разных областях, например, micro:bit в Великобритании, MicroPython, Raspberry Pi Foundation. И я чувствую, что здесь в США нужно догнать и ухватить всё это и лучше использовать инструменты и учебные

ОРГАНИЗОВАТЬ PYCON СТАНОВИТСЯ НЕМНОГО ЛЕГЧЕ

«Это поднимает мне настроение, потому что я тону с головой в работе и стараюсь, чтоб всё было сделано, и вот вижу все эти восторженные твиты. Приятно знать, что сообщество любит бывать здесь.»

оценки стоимости, которые никогда не бывают точными, но, по крайней мере, надо представлять, сколько стоит питание в каждом городе. Это уйма времени, организованность и терпение.

LXF: Вследствие масштабности PyCon большинство выступлений профессионально записываются и выкладываются в свободном доступе после мероприятия.

ИДж: Да, они выкладываются на YouTube (<http://bit.ly/LXF240pycon>) и pyvideo.org, это отличный источник всех связанных с Python конференций и видео. Мы также записываем учебные сеансы. Это трехчасовые тренинги в помощь практическому обучению, потому что вы имеете этот контент в своем распоряжении и под своим контролем.

пособия, которые Великобритания бесплатно создает. Затрат труда на «схватить и убежать» с чужой работой, из любой точки мира, гораздо меньше, и это очень ценно.

Образование — огромная часть этого, поскольку надо обеспечить, чтобы люди, занятые в образовании, делали свое дело квалифицированно и эффективно. Я рада, что в рамках PyCon есть саммит по образованию, для решения ряда этих вопросов с преподавателями, с последующей передачей находок более широкому сообществу. В нашем сообществе много людей, заинтересованных в том, чтобы помочь другим учиться. Саммит по образованию мы начали записывать: люди решили, что обсуждаемые там темы актуальны для всех, и мы должны поделиться этим с другими, кто не смог принять участие в мероприятии.

Николас Толлерви

[Nicholas Tollervey]

LinuxFormat: Николас, спасибо, что нашли время поговорить с *Linux Format*. Что привело вас из Великобритании на PyCon?

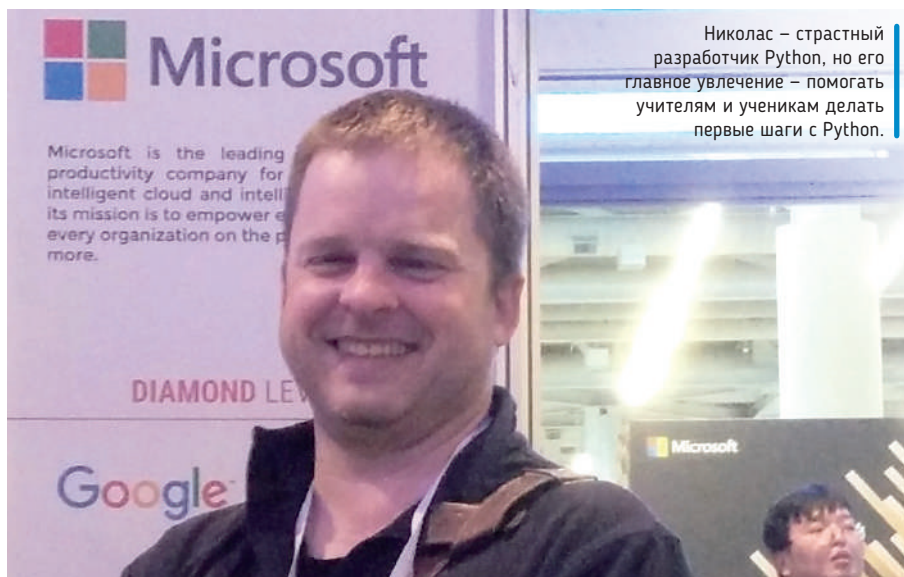
Николас Толлерви: Я разработчик Python, а это же PyCon, самая большая конференция Python в мире! Сюда приезжают люди со всего света, чтобы поучаствовать.

Обычно они принимают участие и в своем национальном мероприятии PyCon, но на PyCon US у нас есть члены сообщества со всего мира, они собираются вместе раз в год, и я никогда не оказался бы с ними в одном помещении, если бы не PyCon. Я бы сказал, что это даже не конференция, а скорее упражнение в построении сообщества.

LXF: Во вступительной речи мы услышали, что «Python — это сообщество, у которого есть язык программирования». Кажется, у сообщества Python это в самом сердце.

НТ: По-моему, PyCon — чрезвычайно позитивный опыт, и именно это объединение сообщества делает всех такими счастливыми. Я знаю, что для некоторых людей это способ набраться энергии и хорошего настроения и обнять друзей, потому что, когда они возвращаются к своим обычным занятиям, там этого нет. Приехав на PyCon, вы оказываетесь в кругу друзей, и это отличное место для подпитки коллективной энергией и энтузиазмом всех присутствующих.

LXF: Речи на PyCon, как правило, «сосредоточены на языке», но вы только что побывали на выступ-



Николас — страстный разработчик Python, но его главное увлечение — помогать учителям и ученикам делать первые шаги с Python.

демонов. Там была организация под названием «Последняя миля», устроившая компьютерные курсы для заключенных, где их обучали кодированию и web-разработке. Крис прошел их, был условно освобожден в прошлом году и поддерживал отношения с «Последней милей» и непредвзятыми компаниями, готовыми оказать поддержку бывшим заключенным, взаимодействуя с ними без предубеждений.

Сейчас ему 43 года, и он прошел шестимесячную стажировку, в результате чего получил предложение о работе. Он изменил свою жизнь, и его

таким образом, что предлагает основные функции, оставаясь доступным.

Когда вы начинаете кодировать, у вас есть среда, в которой очевидно, где вводить код, как его запустить, как диагностировать любые проблемы, а также сохранять и загружать файлы. Всё это через несколько недель либо месяцев подводит вас к тому, что вы, наконец, чувствуете себя достаточно уверенно в качестве программиста, чтобы понять основные концепции работы IDE. Имеется отладчик, REPL и т.д., и вы можете перейти к полноценной IDE, достигнув в Python необходимого уровня навыков.

LXF: Но *Mu* не лишен функциональных возможностей — он обладает модальным интерфейсом для работы со многими платформами Python?

НТ: *Mu* начал свою жизнь под эгидой PSF (Python Software Foundation) как вклад в проект BBC micro:bit. К сожалению, он не был подхвачен micro:bit Foundation.

Тем не менее, большой интерес к моей работе проявили учителя — они хотели применять *Mu* с другими связанными с Python продуктами, такими как PyGame Zero, печатные платы Python от Adafruit, а также в общих проектах Python. В результате я получил поддержку со стороны Raspberry Pi Foundation, который позволил мне добавить режим select для размещения этих функций.

Итак, теперь *Mu* поддерживает micro:bit, схему Python Adafruit, PyGame Zero и Python 3 в одном редакторе. *Mu* можно загрузить с <https://codewith.mu>, причем для Windows, Mac, Linux и Raspberry Pi. На сайте есть ссылки на учебные пособия и ресурсы для учащихся и преподавателей. *Mu* — открытый ресурс, плод бескорыстного труда добровольцев нашего сообщества.

О НЕПРЕХОДЯЩЕЙ ПРИВЛЕКАТЕЛЬНОСТИ PYCON

«Я знаю, что для некоторых людей это способ набраться энергии и хорошего настроения и обнять друзей, потому что, когда они возвращаются к своим обычным занятиям, там этого нет.»

лени с более эмоциональной и личной историей. Можете ли вы рассказать нам об этом больше?

НТ: Случайно я присутствовал на этом выступлении, интересном выступлении Криса Шумахера [Chris Schumacher], который говорил о своей жизни и о том, как у него всё пошло наперекосяк, и он в конечном итоге совершил убийство и попал в тюрьму в Калифорнии, что послужило сигналом к его пробуждению.

17 лет тюрьмы стали для Криса путешествием, в котором он пытался победить своих личных

история вдохновляет. Под конец его выступления выстроилась длинная очередь людей, которые просто хотели пожать ему руку.

LXF: Сейчас вы работаете над новым редактором Python, который упрощает процесс изучения Python. Можете рассказать об этом подробнее?

НТ: Большинство IDE Python предназначены для разработчиков-профессионалов. Мой проект *Mu* — это редактор кода Python для начинающих. Это настоящий редактор кода, но упрощенный

Джесс Инграсселлино

[Jess Ingrassellino]

LinuxFormat: Джесс, спасибо за время, которое вы нам сегодня уделите. Не могли бы вы рассказать читателям *Linux Format* немного о себе?

Джесс Инграсселлино: Привет всем, я Джесс, и в рабочие дни я инженер-тестировщик программного обеспечения, а по выходным преподаю Python детям в возрасте от восьми до 18 лет. Также я работала над учебными планами и методиками, которые были бы доходчивы для детей и помогли бы учителям, интересующимся технологиями, но не обязательно имеющим доступ к внедрению технологии в классе.

LXF: И что же привело вас в Кливленд на PyCon в этом году?

ДжИ: Я на PyCon в этом году, потому что я председатель саммита по образованию, которым являюсь последние два года. На саммите по образованию я пять лет назад начала работать волонтером, потому что хотела видеть таких детей, как мои ученики, которые приходят из городских школ бедных кварталов.

Я хотела, чтобы у них был способ узнать о технологии. Дети, со своими телефонами и планшетами, зачастую лишь потребители. Я считаю, что баланс сил смещается, когда они также могут стать создателями того, что они потребляют. Так что еще в 2012 г. я много на Python не писала, в основном только по работе, но когда набирали добровольцев для организации саммита, я подумала: «Эй, я доброволец, я сделаю это!»

LXF: Образовательный саммит — это место, где преподаватели встречаются, чтобы обсудить, как обучать языку Python по всему миру.

ДжИ: Очень ценно собраться вместе как международное сообщество и поговорить о проблемах, с которыми мы сталкиваемся в наших «маленьких уголках», ведь в течение большей части года бывает трудно дотянуться друг до друга и поговорить. Не потому, что мы не можем (Интернет решил эту проблему), но в плане поиска времени и обсуждений.

Мы также гонимся за тем, что нам нужно, и можем не воспринимать то, что другие пытаются сказать нам, если, по нашему мнению, это не сразу вписывается в нам необходимое. Было довольно весело приглашать других людей, которые могли рассказать нечто вне рамок, но дающее возможность научиться чему-то или способное вдохновить. Я думаю, что действительно важно объединить в этих пространствах учителей, программистов и инноваторов, предоставив им шанс делиться и вдохновляться.

LXF: У детей есть много способов изучать Python, и один из популярных — через онлайн-курсы

и порталы, такие как repl.it и trinket.io. Полезны ли они как образовательные платформы?

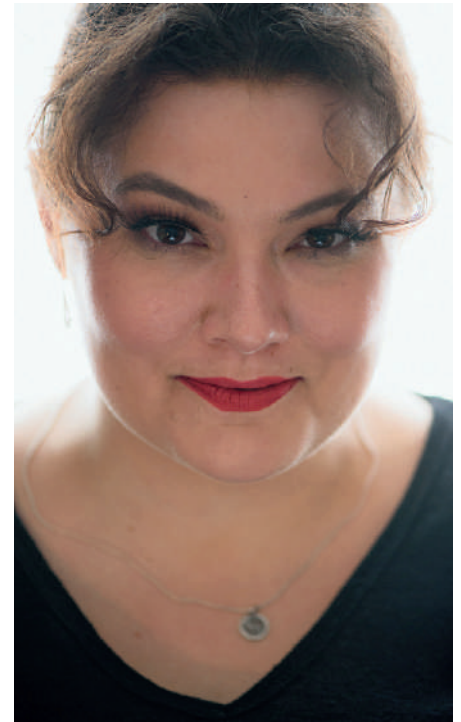
ДжИ: Во время преподавания я обнаружила, что путь через Интернет помогает детям быстро начать, но мой выбор метода зависит от учеников.

Моя текущая учебная программа целиком доступна онлайн на repl.it, и мы используем другую платформу под названием py3.codeskulptor.org, очень похожую на PyGame, только она полностью сетевая. С помощью этих инструментов мы можем создавать базовые 2D-игры, например, *Pong* и *Plants vs Zombies*. Я использую repl.it как платформу для кода и область для хранения, потому что код [codeskulptor](http://codeskulptor.org) работает только на сайте, и вы не можете запустить его с repl.it.

В порядке эксперимента я давала некоторым моим ученикам [micro:bit](http://micro.bit). И заметила, что разные ученики прибегают к разным вещам, и, по-моему, это и обуславливает решение. Так или иначе, системная работа — традиционный подход в классе — это не то, как на самом деле обучение работает: «Если ребенок понимает, так и делай, а если нет, ставь двойки». Это не реалистично, и цели образования нуждаются в пересмотре.

LXF: Интересно, что вы предпочитаете адаптировать обучение к потребностям ребенка. Как правило, обучение основано на сессии, когда дети изучают заданное количество материала за семестр.

ДжИ: В школах США примерно то же. Одна из моих любимых вещей, касающихся преподавания технологии здесь — то, что пока не определено, как оно должно выглядеть. В США образование



Джесс — движущая сила саммита PyCon по образованию. Преподаватели съезжаются сюда со всего мира, чтобы поделиться своими идеями и опытом в обучении Python.

то, что я могу экспериментировать, и делает предмет интересным.

Я свела свою технику преподавания к стилю джазовой импровизации. Джазовые музыканты

КОДИРОВАНИЕ С ЧУВСТВОМ РИТМА

«Я свела свою технику преподавания к стилю джазовой импровизации. Джазовые музыканты знают свои азы, но не знают, что сделают, выйдя на сцену. Я хочу придерживаться тех же рамок.»

считается ответственностью штата. Оно не является федеральной ответственностью.

У нас есть общая основная учебная программа; она является добровольной, и люди выбрали следование ей, но это никто не требует. Технология — это как раз то, что многие люди не понимают, как реализовать, и я видела, как неправильно это делают во многих школах. Они не знают, что детям желательно вынести из этого. Все люди, кажется, видят только, что «дети могут зарабатывать деньги, получив работу, связанную с кодированием, и мы должны этому учить!» Но именно

знают свои азы, настройки, мелодии, аккорды и т.д. Они не знают, что сделают, выйдя на сцену, но представляют себе закономерность. Я хочу придерживаться тех же рамок, так что я продумываю мои поурочные планы не как сценарий, но скорее как способ двигаться вперед, используя основные понятия и принцип: «вот что надо уметь делать, вот навыки, которые вам следует иметь, и если у вас есть эти навыки (гаммы) и основные понятия (серии аккордов), вы сможете построить всё, что захотите, потому что это для вас!»



Гвидо ван Россум

[Guido van Rossum]

LXF: Гвидо, большое спасибо, что нашли время побеседовать с *Linux Format*. Большая удача подобраться к вам сразу после вашего общения со следующим поколением пользователей и разработчиков Python в лице Джошуа Лоу (Joshua Lowe), создателя Edublocks.

Гвидо ван Россум: Было здорово наблюдать, как он объясняет и показывает свой проект, и демонстрирует, как он умен и как заботится о том, чтобы донести эту трудную тему до других, и просто узнать, какой это замечательный ребенок.

LXF: Интересно видеть, как дети используют Python в качестве «первого языка». Когда вы впервые создали Python, думали ли вы, что он будет таким популярным и таким универсальным языком?

ГвР: Я понятия не имел об этом. Я хотел поделиться им, но у меня было гораздо более ограниченное представление о том, с кем я буду делиться. Я ожидал, что всё ограничится небольшим количеством пользователей, похожих по духу на завсегдатаев Usenet начала 1990-х годов. Честно говоря, сперва я нацеливался на профессиональных программистов, являющихся также штатными разработчиками UNIX по существу, или что-то вроде того.

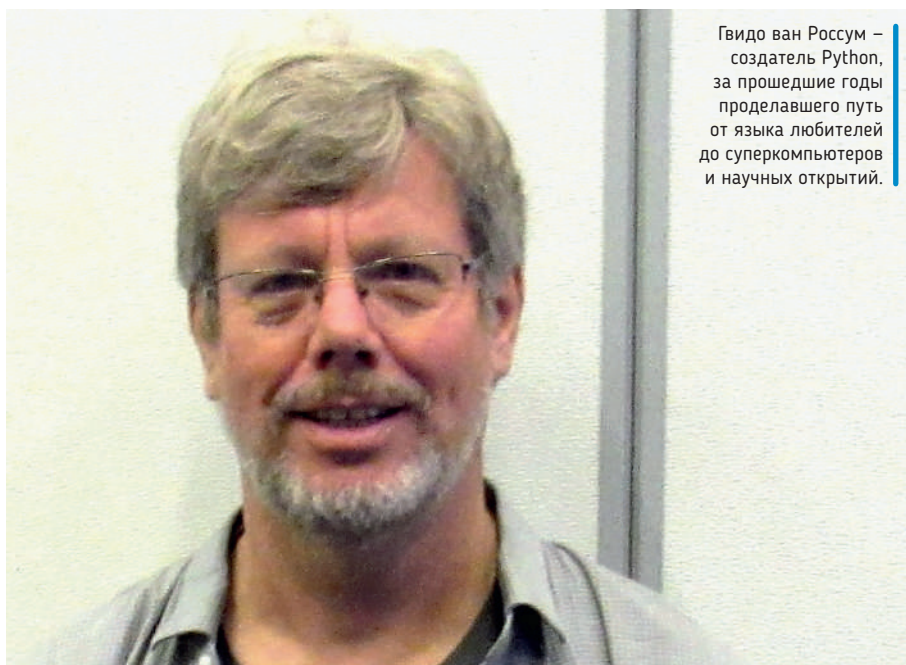
LXF: Переходя от профессионального применения языка, Python теперь хорошо известен своим применением в образовательной среде, главным образом вследствие появления Raspberry Pi. Считаете ли вы, что Raspberry Pi обусловил повышение известности Python среди преподавателей и студентов?

ГвР: Я этого не изучал, и точно не знаю, но думаю, что Pi, кажется, имеет положительный эффект. В смысле, просто невероятно, что они могут делать такое оборудование доступным по такой цене, ведь это очень многое меняет. Это облегчает доступ тем, кто не мог позволить себе компьютер, или тем, кто получал этот доступ только на короткое время, например, при совместном использовании школьного компьютера; а теперь они могут получить свой собственный компьютер и с ним повозиться.

LXF: То есть Pi демократизировал доступ к компьютерной платформе?

ГвР: Похоже, они этого и хотели, и похоже, что они этого достигли. «Raspberry Pi» слышится отовсюду.

LXF: Raspberry Pi также поощрил новое поколение к самоделкам, которые, как хорошо известно, были формирующим опытом вашего детства. [В 1960-х гг. Гвидо конструировал наборы для радиолюбителей от Philips.] Является ли



Гвидо ван Россум — создатель Python, за прошедшие годы проделавшего путь от языка любителей до суперкомпьютеров и научных открытий.

это важным навыком, которому каждый должен научиться?

ГвР: Самоделки выходят далеко за рамки Python, и кажется, что одним людям это нравится, а другим — нет, а есть и такие, которые еще не пробовали.

Я не берусь утверждать, что каждый обязан быть самоделкой, но если вы склонны к исследованиям и решению задач, то инструменты программирования вроде Python, Raspberry Pi и программного обеспечения, подобного Edublocks,

весь язык. Универсальность имеет место отчасти благодаря высокому уровню языка, что позволяет любому взять и попробовать его. Например, Arduino обычно используют для обучения электронике и программированию, но его язык многие находят слишком сложным на первый взгляд.

ГвР: Вы тратите больше времени на пререкания с компилятором. Приятно слышать, что Python удобен и прост не только в освоении, но и в преподавании.

ОТ КОДИРОВАНИЯ К ДИЗАЙНУ ИНТЕРЬЕРА!

«Вы можете проводить время, собирая детали, и охнуть не успеете, как изумительная, невероятная программируемая гирлянда светодиодов уже будет висеть в вашей спальне.»

просто делают такие занятия намного увлекательнее, поскольку те приобретают гораздо больший масштаб. Вы можете проводить время, собирая детали, и охнуть не успеете, как изумительная, невероятная программируемая гирлянда светодиодов уже будет висеть в вашей спальне!

LXF: Использование Python в качестве первого языка кодирования показывает его универсальность. Человеку можно показать, как включить/выключить светодиод, и эта логика пронижет

LXF: Поскольку Python вызывает больший интерес у преподавателей, есть ли необходимость в создании учебной программы для преподавателей при обучении Python?

ГвР: Я думаю, что правильнее задать этот вопрос в различных рабочих группах и группах сообщества, которые занимаются образовательными ресурсами.

LXF: Значит, существует образовательная рабочая группа?

ГвР: Ну, ежегодно на PyCon проходит EduSig, однодневный семинар для педагогов, где обсуждается применение Python в образовании и ресурсах для педагогов.

LXF: Сегодня суббота PyCon, и выступления продолжают уже два дня. Какое вам больше всего понравилось?

ГвР: Мне на самом деле невероятно трудно попадать на выступления, поскольку я постоянно встречаюсь с людьми, желающими поговорить со мной, и мне нравится разговаривать с ними не меньше, чем смотреть выступления.

Сегодня я был на выступлении Lady Red [«Красная дама», настоящее имя — Крис Бичем [Chris Beacham], — прим. пер.] про опыты со светодиодными полосками с использованием Raspberry Pi и Python, и это было действительно здорово. Еще одно выступление, на котором я побывал перед обедом — выступление Дэвида Бизли [David Beazley] об изобретении нового парсер-генератора. Его тема гораздо ближе к моему профессиональному опыту, потому что это про написание парсеров для языков, подобных Python. Дэвид — изумительный оратор, он умеет сделать интересной любую тему, а данная тема весьма нуждается в таковом!

LXF: Ранее мы говорили об универсальности Python, и во вступительной речи мы услышали от Дэна Каллахана (Dan Callahan) (Mozilla), что «Python является вторым лучшим языком для изучения».

ГвР: Да, хотя, если вас интересует мое абсолютно честное мнение, я подумал, что это странно. Может быть, некоторые люди так и думают про Python. С моих позиций трудно видеть мир таким же, каким его видят другие. Вообще-то я подозреваю, что Дэн главным образом хотел немного поддразнить сообщество Python, чтоб оно пришло в восторг от портирования Python в web-браузеры.

LXF: Когда разработчик или кодировщик напишет код, он будет супер-критически оценивать его.

А по отношению к своему собственному коду вы испытываете такие чувства?

ГвР: Это зависит от аудитории, для которой я пишу. Когда я пишу для Python или другого открытого проекта, типа туру, я выполняю тесты, пишу комментарии и строки документации, даю правильные имена переменным, придерживаюсь разметки и стиля — делаю всё возможное, но это требует времени. Мы неоднократно инспектируем код, и я заставляю других участников проекта придерживаться таких же высоких стандартов.

С другой стороны, иногда я пишу фрагмент кода, который намереваюсь использовать только один раз, чтобы выполнить задание, и эта программа Python и отступ будут выключены, не будет функции main, поскольку код одноразовый. Худшее случается, когда нечто предназначенное для одноразовой задачи внезапно снова употребляется для аналогичной задачи, и я обнаруживаю ошибку в своем скрипте, и должен выполнить



рефакторинг кода, и не успеешь оглянуться, как это становится целым проектом.

Но иногда отношение к написанию такого кода остается: «Ну, это проект не важный, и я не собираюсь платить цену за все тонкости». Никто не пишет идеальный код всегда. Я старею, печатаю ужасно и медленно, и иногда мой мозг глючит.

LXF: Есть ли у вас советы для разработчиков и молодых программистов, делающих свои первые шаги?

ГвР: В зависимости от того, каков ваш первый шаг. Когда ты маленький, ты изучаешь, что собой представляет язык. Найди хороший учебник и достань хорошую книгу. Разных книг много, и они используют разные подходы к просвещению. Например, есть расширение Minecraft Python для Raspberry Pi, а есть другие вещи, такие как PyGame. Так что купи учебник, адресованный тебе, и работай с ним. Делай это с кем-то другим, если можешь, потому что вдвоем учиться быстрее, чем в одиночку.

Когда вы почувствуете, что «это легко, что мне теперь делать, как мне стать лучше?», начните читать код других проектов и поищите на GitHub проекты, нуждающиеся в помощи. Есть много хороших примеров работы больших кодовых баз. Отличный проект, который приветствует стороннюю помощь, называется Home Assistant. Он весь на Python 3, это реальное приложение, способное контролировать все устройства IoT в вашем доме. Это отличная кодовая база для обучения, а также отличный проект, чтобы внести свой вклад, если вы хотите сделать больше, чем просто изучить реальный код. Но это только один пример — проектов, которые делают полезные вещи и тоже нуждаются в помощи, намного больше. **LXF**



MINT 19 ПОШЕЛ

Джонни Бидвелл сделал отличную подборку хитростей и знаний Mint, чтобы вы могли по максимуму использовать этот мощный дистрибутив.

Наконец-то дождались! Свежая версия одного из самых популярных дистрибутивов Linux увидела свет, и она предлагает новый внешний вид, новые инструменты и, после того, как Mint был приобретен Microsoft, новых владельцев.

Ну ладно, последнее было неправдой, но зато привлекло ваше внимание, а данный релиз его стоит. Потому что он потрясающий. И, чтобы внести ясность, Linux Mint по-прежнему независимо разрабатывается и финансируется пожертвованиями, рекламой на сайте и спонсорами.

Ветераны Mint найдут, что всё так любимое ими стало еще лучше, а новички обнаружат мощную, простую в использовании и приятную на вид ОС. Используя солидную базу Ubuntu 18.04 и все чудесные новшества, Mint добавляет уровень изящества, одновременно современный

и традиционный. Имеются новые версии рабочих столов Cinnamon и Mate, новый инструмент резервного копирования, свежий подход к обновлениям и многое помимо этого.

Есть также небольшие дополнения, которые играют важную роль: удобный экран приветствия, *Redshift* (для

снижения напряжения глаз), удобная программа записи образов на USB.

За этим релизом стоит та же восхитительная приверженность сообществу, то же понимание, что людям не нравится, когда им велят использовать свой рабочий

стол так-то и так-то, и то же желание создать разновидность Linux, которую будут любить все, от зеленых новичков до суровых технарей.

Итак, давайте глубоко вдохнем все мятные метафоры и все сравнения со специями в этом восхитительном дистрибутиве Linux.

ПОЧЕМУ MINT ПРОСТО РАБОТАЕТ

«Используя солидную базу Ubuntu 18.04 и все чудесные новшества, Mint добавляет уровень изящества...»

Из-за чего расшумелись?

Прошло всего семь месяцев с последнего отличного релиза Mint, так что же нового на сей раз?

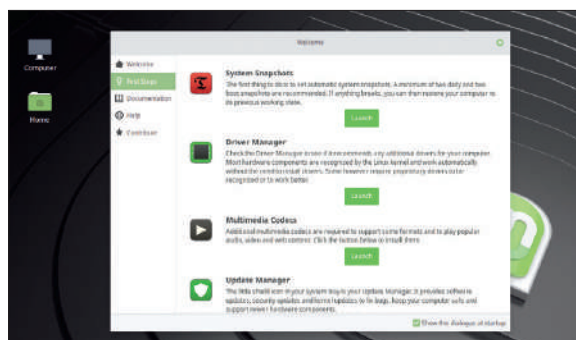
Основанный на апрельском релизе Ubuntu 18.04, Mint 19 (как и два предыдущих поколения) будет поддерживаться пять лет, и мы можем ожидать основные релизы через шесть, 12 и 18 месяцев. На сей раз у нас три варианта рабочего стола: Cinnamon, Mate и Xfce, и каждый предлагает что-то свое. Отсутствует вариант KDE, хотя пользователи всегда найдут KDE Plasma в репозиториях Ubuntu (или Gnome, Budgie, i3 или любой другой рабочий стол или менеджер окон, который вы назовете).

Мы получаем кучу вопросов от читателей насчет резервных копий, и не так-то просто предложить решение, годное для всех случаев. Конечно, мы можем посоветовать людям старый добрый способ — записать процедуры *cron*, которые используют дикие и безумные опции *rsync*; но удобными для пользователей их не назовешь. Кроме того, процесс восстановления довольно жуткий, и весьма вероятно нанести еще больший ущерб, в попытке что-то восстановить поставив гадкий обратный слэш не там, где надо.

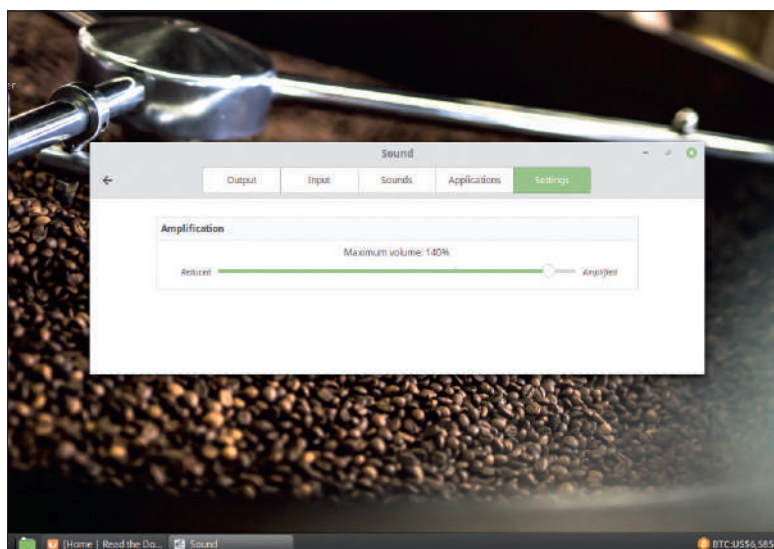
И очень кстати, что одно из самых заметных изменений в Mint 19 — как раз инструмент резервного копирования, *Timeshift*. *Timeshift* является аналогом *Time Machine* в macOS или точкам восстановления системы в Windows, поскольку больше заботится о создании моментальных снимков директорий системы, чем о файлах пользователей. *Timeshift* может делать и их резервные копии тоже, если его попросить, но для этой задачи больше подходит *Backup Tool*. Мы посвятили целых две страницы *Timeshift* (см. стр. 44 и 45), так что, пожалуйста, сообщите нам, было ли это полезно для вас.

В то время как Ubuntu рвется к созданию пакетов Snap, Mint больше склоняется к Flatpak. Поддержка Flatpak появилась в Mint 18.3 и в этом релизе улучшена. Приложения Flatpak можно устанавливать прямо из *Software Manager*, что обеспечивает легкий доступ к самым свежим версиям популярных программ, таких как *Spotify*, *Atom*, *Telegram* и *Slack*. Snaps не остались за бортом — их просто надо устанавливать из командной строки. Честно говоря, на момент написания вам придется делать то же самое и с Flatpaks, но это из-за ошибки в *Software Manager*, которую скоро исправят.

Здесь происходит масса интересного. Например, продолжает улучшаться поддержка HiDPI. Тема значков Mint-Y включает



Удобный экран приветствия поможет вам сделать первые шаги и ответить на все ваши вопросы о Mint.



Любители установить громкость на 11 оценят эту новую настройку громкости на глобальный максимум.

значки в высоком разрешении — они будут отлично выглядеть на 4K-дисплеях. Многие приложения теперь переключились на символные значки, лучше различимые в темных темах, которые (видимо, из-за мрачного политического климата) становятся всё популярнее.

Mint всячески изгоняет наследство от Ubuntu. Исчезли подборки данных из версии 18.04 и вернулась опция установки с зашифрованной в *ecryptfs* домашней директории. Mint продолжает обслуживать также пользователей 32-битных машин, и вы найдете на диске чудесную 32-битную редакцию с Mate.

» ИСКРОМЕТНЫЙ CINNAMON

Cinnamon 3.8, флагманский рабочий стол Mint, вышел в начале мая 2018 г. и предлагает впечатляющую тему Mint-Y. Следуя современным тенденциям дизайна, она гладкая и плоская. Mint-Y была доступна в Mint 18, хоть и не по умолчанию, но с тех пор подверглась более тонкой настройке и доведена до совершенства. Если вы предпочитаете классическую тему Mint-X, существующую с Mint 10, не расстраивайтесь — она по-прежнему доступна.

Внутри есть ряд важных улучшений производительности. Менеджер окон *Muffin* предлагает более быстрый рендеринг окон, а менеджер файлов *Nemo* быстрее передает файлы в сеть или на диски USB, а также должен заметно ускорить рендеринг содержимого больших директорий. Более того, теперь *Nemo* позволяет сохранять общие результаты поиска, которые потом будет индексироваться в фоновом режиме для более быстрого поиска в будущем.

Если вам не нравится его готовый внешний вид, Cinnamon можно переделать с помощью множества других тем, расширений, апплетов и десклетов (которые все вместе именуются *spices* [пряности]). Они могут всё, от возвращения колеблющихся окон до отображения погоды (которая здесь, в Блайти, против обыкновения хороша, что сильно вредит работе над журналом).

Исследуйте среду live

Если вы хотите познакомиться с Mint, не затрагивая своего жесткого диска, запустите дистрибутив прямо с нашего тщательно подготовленного DVD...

На DVD **LXF239/240** было две версии Mint: 64-битная редакция Cinnamon и 32-битная редакция Mate.

Даже если у вас 64-битный компьютер, вы можете спокойно познакомиться с релизом Mate. Помимо всего прочего, он сэкономит вам 1,9 ГБ загрузки по сравнению с 64-битным собратом [Ред.: — Ого, дистрибутивы ныне стали весьма увесистыми].

Если вы хотите попробовать редакцию Xfce (которая весит чуть меньше, чем Mate), вам придется скачать ее самостоятельно, однако мы полагаем, что читатель вашего уровня легко справится с этой задачей. Если вы уже знакомы с Cinnamon-редакцией Mint, вы увидите, что всё практически там же, где и раньше. Если же

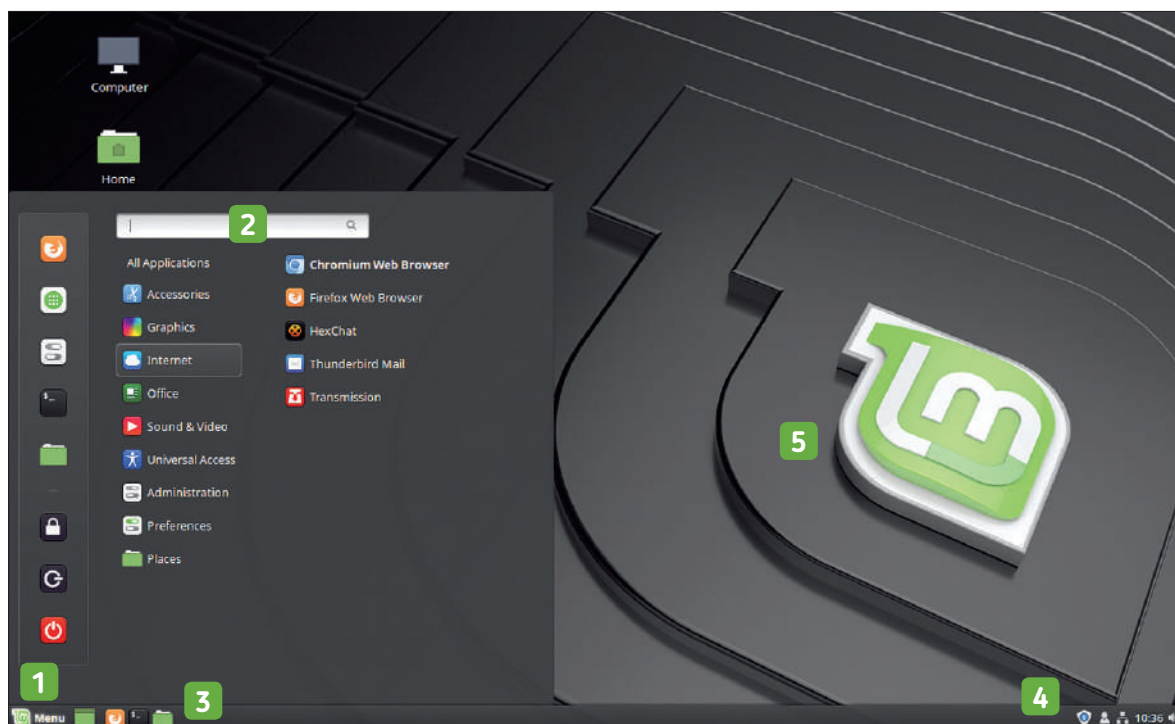
нет, удобная аннотация (внизу) покажет вам основные области Cinnamon.

Программы можно устанавливать в среду live, как и в обычную установку, но все они, естественно, исчезнут после выключения компьютера. Загляните в *Software Manager* (с зеленым кружком и белыми точками со стороны меню, и он также есть в Mate и Xfce), чтобы познакомиться со значительной предлагаемой подборкой.

Прелести Flatpaks

Обычная претензия к дистрибутивам LTS — что их набор пакетов актуален только на момент выхода. Говорят, время нас всех

Обходим рабочий стол Cinnamon



1 Меню

Как и у всех рабочих столов, которые вы помните, здесь есть меню в нижнем левом углу. Установленные вами программы доступны в этом меню, мило расставленные по категориям. Свежеустановленные программы выделяются жирным шрифтом, как *Chromium* на рисунке.

2 Поиск

Однако, как и во всех крутых рабочих столах, вы можете начать вводить слово для поиска по установленным программам. Это работает и для описаний программ, и для их имен, так что поиск по "xed" или "text" найдет вам текстовый редактор.

3 Значки быстрого запуска

Щелчок по зеленому значку рядом с меню минимизирует все окна и покажет рабочий стол, а нажатие средней кнопки выделит все активные десклеты. Другие запустят ваши любимые приложения. По умолчанию это *Firefox*, приложение *Terminal* и менеджер файлов *Nemo*, но вы можете перетащить всё, что хотите, из меню, и присоединить к ним.

4 Апплеты

Здесь вас уведомят о статусе всех внешних дисков, подключенных к вашей машине, или обо всех доступных обновлениях пакетов. Здесь также имеются функции управления, позволяющие

переключить пользователя, выйти из системы или вырубить компьютер. Рядом с ними — кнопки управления *Network Manager*. Щелчок правой кнопкой по пустой области панели откроет меню, откуда можно добавлять апплеты или настраивать пункты.

5 Рабочий стол

Щелкните по нему правой кнопкой — откроется меню, где можно изменить фон, добавить на свой рабочий стол виджеты («десклеты», на жаргоне Cinnamon) или выбрать, какие отображать значки. Cinnamon без колебаний добавит на рабочий стол ваши файлы. Но рой значков (это так, к слову...) закроет от вас чарующий фон.

делает глупцами — или, по крайней мере, состаривает некогда актуальные пакеты. Старая возможность обойти эту проблему заключалась в использовании сторонних PPA, но они бывают нестабильными — особенно когда о них забываешь или используешь несколько, конфликтующих между собой. Новая возможность — использовать интересное пространство имен ядра и функции контейнеров, которые позволяют разработчикам включать собственные программы и их зависимости в пакеты, способные работать в любом дистрибутиве.

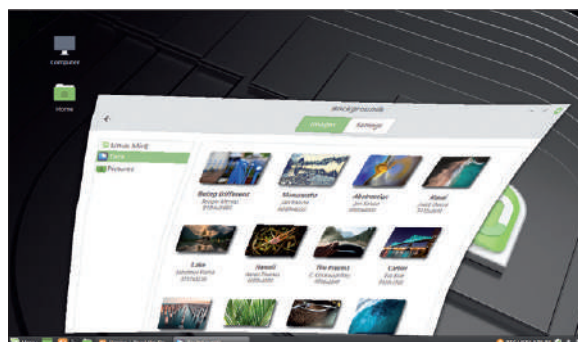
Есть как минимум три разных способа сделать это, и Mint предпочитает использовать Flatpak. Вы можете установить Flatpaks из командной строки, скачав файлы `.flatpakref` из доверенного источника (скажем, <https://flathub.org>), или через *Software Manager*. В последнем имеется целый раздел про Flatpaks. Итак, если вы опечалены отсутствием *GIMP 2.10* в репозиториях Ubuntu, именно здесь это можно будет исправить. В tandem с *Software Manager* идет *Update Manager*, куда вас направят, если Mint обнаружит новые версии установленных пакетов.

Индивидуальная настройка Cinnamon

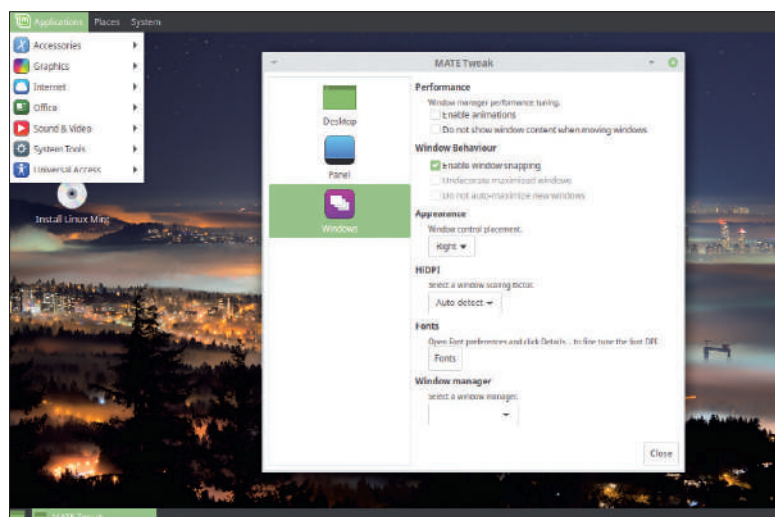
Ветераны Mint, вероятно, заметили новую тему Mint-Y. Очень стильная, а? Однако всё можно переделать, зайдя в раздел Themes [Темы] в меню Preferences [Предпочтения]. Вы можете переключиться на более старую тему Mint-X, поработать с цветовыми схемами и изменить весь вид дока и меню. В разделе Add/Remove [Добавить/Удалить] показаны темы, разработанные сообществом, с сайта Cinnamon Spices (<https://cinnamonspices.linuxmint.com>).

Помимо тем, рабочий стол Cinnamon можно приправить апплетами и десклетами, у которых есть собственные разделы в меню Preferences. Апплеты — это такие штуки, которые живут на основной панели (или на любой панели, если вам нравится несколько панелей). Часть их установлена по умолчанию, например, часы, календарь и меню пользователя. Gnome запретил программам классический системный лоток, где приложения могли добавлять собственные меню быстрого доступа и уведомления, однако в Cinnamon он по-прежнему существует. При желании вы тоже можете его заблокировать, но тогда вы не увидите значков в лотке, среди которых — удобные напоминания о появившихся обновлениях.

Десклеты — это штуки другого рода, они сидят вверху экрана. Классический десклет — программа для заметок Post-It, небольшой шаг вперед от наклеивания на монитор нацарапанных памяток. Если вам нужны эти желтые реликты на вашем рабочем столе Cinnamon, то скачайте Note из раздела Desklets. Вы также найдете десклеты для просмотра популярных web-комиксов Xkcd, отображения экстравагантных/интеллектуальных/псевдо-мистических цитат и сообщений о том, какая настанет чудесная погода.



Помните, как круто было иметь окна, которые вытворяли такое? Ну вот, они и сейчас могут так делать, причем не удушая ваш компьютер.



Установите Mate tweak tool и воссоздайте аутентичный вид Gnome 2 с триадой каскадных меню в верхней панели.

Еще больше трансформаций рабочего стола возможно с Extensions [Расширениями]. Они меняют внешний вид Cinnamon, в основном по части украшения. Фанаты эффектов рабочего стола старой школы найдут такие сокровища, как Wobbly Windows [Колеблющиеся Окна], Desktop Cube [Рабочий стол-куб] и разные опции прозрачности и размытия элементов рабочего стола.

Если у вас разгорелся аппетит, во врезке внизу вы найдете советы по установке. Главное здесь — осторожность: пожалуйста, не вините нас, если вы случайно удалите Windows, и не нажимайте на кнопку Go, если у вас есть сомнения. Финальное замечание: на момент написания только что был завершён путь обновления Mint 18.3. Вы можете прочитать официальный пост на <https://blog.linuxmint.com/?p=3615> и плавно обновить свой заботливо настроенный 18.3 до следующей версии.

» УСТАНОВКА MINT

Mint использует установщик Ubuntu, *Ubiquity* — у него хорошая репутация, и он почти наверняка не нанесет урона вашей системе. Подробное руководство вы найдете в нашей статье про Ubuntu в **LXF237**, но установщик и сам сообщит всё, что надо знать. Если вы планируете сделать Mint своей единственной операционной системой, то вам вообще не о чем беспокоиться. Разве что перед началом процедуры проверьте, что вам действительно не нужны данные, оставшиеся на целевом диске. Если вы планируете на том же диске наличие двойной загрузки с Windows, то здесь вам придется изменить объем разделов, поскольку *GParted* со товарищи имеют скверную привычку повреждать новые файловые системы NTFS. По возможности рекомендуем сделать установку на отдельный диск. Помимо всего прочего, это даст вам некий простор.

В любом случае, сделайте резервные копии всех важных файлов и убедитесь, что у вас есть всё необходимое для возврата системы в ее предыдущее состояние (например, установочный носитель для ОС, ключи активации/регистрации), если что-нибудь пойдет не так. В наших экспериментах установка Mint (редакция Cinnamon) заняла около 6,5 ГБ, включая 1-ГБ файл подкачки. Поскольку всячески поощряется наличие нескольких моментальных снимков (см. наш раздел о *Timeshift*), и как минимум один моментальный снимок будет включать большую часть файловой системы (если вы используете *rsync*), для установки Mint вам понадобится минимум 20 ГБ, а в идеале намного больше. Возможно, вы захотите разместить `/home` или `/timeshift` на отдельном разделе, что приведет к меньшему объему системного раздела. Но чем больше вы будете узнавать о возможностях Mint, тем больше вам захочется его установить, и тем сильнее он будет разрастаться.

Timeshift и обновления

Почти как говорили Нейт Догг [Nate Dogg] и Уоррен Джи [Warren G] в 1994 г. — «Backup, backup, backup cause it's on.» И это не о развороте автомобиля...

Одно из самых больших отличий после Mint 18 — работа с обновлениями. Признавая, что у обновлений ядра и драйвера отображения имеется скверная привычка разрушать пользовательскую систему, предыдущая версия Mint по умолчанию не устанавливала их. Она (и ее предшественники) предлагали для обновлений пятиуровневую систему, варьирующуюся от 1 — «не ломать мою систему» до 5 — «обновлять всегда и всё».

Установленный по умолчанию уровень 3 обновлял большую часть области пользователя, но не трогал ничего потенциально способного оставить пользователя наедине с пустым экраном. Это было спорно, но не бессмысленно. Критики заявляли, что таким образом пользователи не получают важных обновлений безопасности, но очень мало найдется опытных линуксоидов, ни разу не пострадавших от того, что обновление сломало им всю систему. Глава проекта Mint, Клеман Лефевр [Clement Lefebvre], говорит об этой дихотомии: «Так же, как невозможно говорить об обновлениях и не затронуть проблему безопасности, невозможно говорить об обновлениях и не затронуть вопрос стабильности. Речь шла о двух противоречащих и очень важных проблемах, и обсуждать их было очень сложно».

Кажется, что это игра с нулевой суммой, однако в Mint 19 дилемма стабильность/безопасность, похоже, разрешилась. С этого релиза все обновления получают по умолчанию. Что согревало бы душу, будь причиной данного изменения уверенность команды Mint в состоянии драйверов отображения Linux или механизма обновления ядра Ubuntu; и, возможно, эта уверенность у них есть. Однако сейчас есть страховка

от регрессий, вызванных обновлениями, в виде инструмента резервного копирования *Timeshift*. Как говорит Клем: «*Timeshift* изменил всё, превратив в обычное дело возврат назад во времени и устранив таким образом воздействие любой регрессии и, следовательно, опасения с ней столкнуться... Это позволило нам сделать автоматические обновления обычной рутинной». *Timeshift* появился в Mint 18.3, «уже в конце цикла», по словам Клема. Но в Mint 19 он занял центр сцены: «Теперь *Timeshift* полностью интегрирован как официальное решение, страхующее не только от ошибок и критических проблем, но и от программных регрессий».

Теперь обновления делятся на четыре уровня, от Minimal [Минимальных] до Sensitive [Требующих особого внимания], и сильно нервные люди могут, как и раньше, отказаться от обновлений ядра. Кстати о ядрах: теперь они устанавливаются в виде мета-пакета *linux-generic*. Этот пакет зависит от самого последнего реального пакета ядра, и как только появляется обновление ядра, старый пакет ядра остается не у дел, и его легко удалить с помощью `apt autoremove`. Надо надеяться, это положит конец проблеме старых и изношенных ядер/модулей, напрасно занимающих место и захламляющих меню *Grub*. Mint 19 также предоставляет через *Software Manager* легкий доступ к ядру с малой задержкой, и это великое благо для аудиофилов и всех тех, кому обычное значение задержки способно испортить песню.

Бэкапы и снимоты

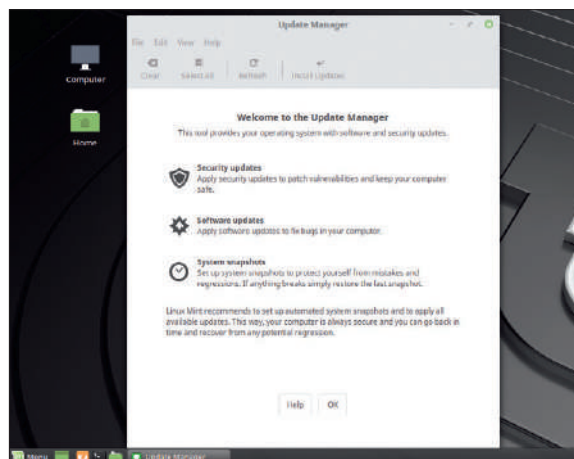
Опция включения автоматических обновлений — новинка Mint 19, но в интересах безопасности она включается только после того, как вы включите моментальные снимки [snapshot] через инструмент резервного копирования [backup] *Timeshift*. Тогда, если автоматическое обновление сломает вашу систему, вы сможете ее восстановить до «последнего нормального» состояния без особого труда. Даже если ваш компьютер окажется

» АППАРАТНОЕ РАЗБЛОКИРОВАНИЕ

Ядро 4.15, которое шло с Ubuntu 18.04 (то же, что поставляется с Mint 19), было довольно новым, когда появился Mint 19. Со временем будут применяться отладки безопасности, и важные новые функции будут обратно портироваться из будущих релизов ядра. Однако не все функции и поддержка оборудования будет поставляться таким образом, поэтому Ubuntu предлагает своим пользователям (и, следовательно, пользователям Mint) стек Hardware Enablement (HWE).

Потребность в новых функциях в основном относится к графике, поэтому помимо нового ядра HWE предлагает более новые версии компонентов дисплея области пользователя (Mesa, libdrm и все части X). С Ubuntu 16.04 модель релиза HWE, которая раньше представляла новый стек с каждым доработанным релизом Ubuntu (и поддерживала эти стеки параллельно), изменилась. Теперь один HWE выходит с доработанным релизом «с двумя точками» и обновляется с каждым новым главным релизом — по сути, это модель возобновляемого релиза. Доработанные релизы планируются каждые четыре месяца после каждого релиза Ubuntu, так что стек HWE 18.04 появится в феврале 2019 г. Это означает меньше работы командам поддержки и чуть меньше работы пользователям, поскольку стек HWE придется установить только один раз, а потом он будет обновляться сам. В любом случае, в феврале, если у вас есть новое оборудование или вам нужны новые функции, стек HWE будет вашим, с помощью

```
$ sudo apt install linux-generic-hwe-18.04 xserver-xorg-hwe-18.04
```



Обновления делятся на категории этими стильными значками, так что «кого любишь, выбирай». Однако официальная рекомендация — обновлять всё.

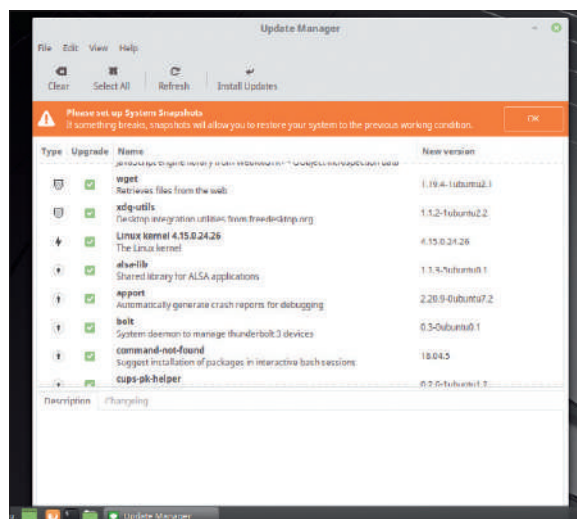
незагружаемым, вы всё равно сможете загрузиться, используя live-носитель Mint, и оттуда восстановиться с помощью *Timeshift*. Нет нужды использовать *chroot* или переживать по поводу стран-ных правил *rsync* насчет обратных слэшей.

С самого начала, с экрана приветствия, Mint настоятельно рекомендует вам настроить моментальные снимки. Существует множество теорий о стратегиях резервного копирования, ко-торые мы не будем здесь подробно разбирать. Достаточно сказать, что простой доступ ко вчерашней резервной копии может и не решить все ваши проблемы. Очевидной проблемой будет потеря вами всего, что изменилось со вчерашнего дня. Другая, не столь очевидная проблема — то, что, возможно, собой приключился задолго до вчерашнего дня, и будет намного проще решить это при наличии у вас резервной копии из более отдаленного прошлого.

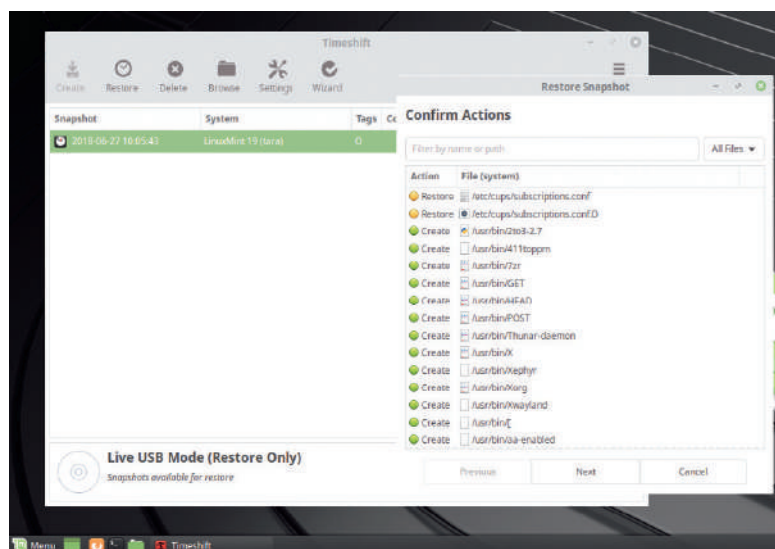
В результате стандартной практикой будет создавать и де-лать ротацию резервных копий в разные сроки. В *Timeshift* нам предлагается сколько угодно моментальных снимков — при за-грузке [Boot], каждый час [Hourly], ежедневно [Daily], еже-дельно [Weekly] и ежемесячно [Monthly], без малейшего намека на *cron*tab. Важно отметить, что *Timeshift* — это скорее инстру-мент резервного копирования системы, чем файлов пользова-теля. Вы можете приказать *Timeshift* делать резервные копии домашних директорий, однако есть лучшие способы управлять резервными копиями личных документов в зависимости от важ-ности этих документов (например, зашифровать их и отправить на удаленный сервер Nextcloud, или копировать в NAS по дости-жении некой вехи).

Моментальные снимки обрабатываются очень четко. Есть выбор между «родными» моментальными снимками *btrfs* (если ваш раздел *root* отформатирован в *btrfs*) и *rsync*. *Btrfs* и его магия CoW работают с неизменными файлами прозрачно на уровне файловой системы и хранят моментальные снимки в подтоме, тогда как *rsync* сохраняет моментальные снимки в директории */timeshift*. В *rsync* первый моментальный снимок будет полной копией файловой системы, и поэтому займет много места (около 6 ГБ для чистой установки по умолчанию), но последующие мо-ментальные снимки будут для неизменных файлов использо-вать жесткие ссылки на предыдущие.

По умолчанию *Timeshift* поддерживает пять моментальных снимков при загрузке и пять ежедневных. Такое может показаться чрезмерным, но если вы вносите много изменений на уровне системы, то это обеспечит вам подстраховку; а если не вносите,



Несравненное большое оранжевое напоминание мотивирует вас настроить моментальные снимки, потенциально способные спасти вашу систему. В Mint это легко.



Щелчка за три наша убиенная директория */usr/bin* вернулась из небытия. Но это не делает удаление системных директорий чем-то похвальным.

то каждый новый моментальный снимок места практически не займет. Помимо домашних директорий, не делаются также резервные копии файлов подкачки. Также обычно исключаются директории, используемые для монтирования внешних носите-лей, как и те, что используются для хранения контейнеров Docker

ПРЕИМУЩЕСТВА TIMESHIFT

«Если автоматическое обновление сломало вашу систему, вы сможете ее восстановить до „последнего нормального“ состояния без особого труда.»

и *chroot*, плюс (чтобы исключить жуткие кошмары рекурсий) сама директория */timeshift*.

Наш первый моментальный снимок занял 6,2 ГБ (лишь немногим меньше полной установки). Те, кто хочет уменьшить размеры моментальных снимков, могут сэкономить, исключив */var/cache/apt*, которая хранит скачанные пакеты *.deb*. Сценарий, когда вам понадобятся старые версии этих файлов, будет отчасти притянутым за уши, однако в возможности доступа ко всем пакетам, когда-либо установленным в вашей системе, есть некий шарм.

Откат, при наличии моментального снимка, очень прост. Мы даже провели деструктивный эксперимент в VM, удалив там директорию */usr/bin*. В результате у нас оказалась незагружаемая система, поэтому мы запустили средство установки, затем *Timeshift*, и примерно за три щелчка (включая одно согласие на пред-упреждение) восстановление запустилось. Штук 2000 удаленных нами файлов вернулись на свое законное место, и после переза-грузки с нашей системой всё опять было хорошо.

Кроме восстановления файлов, *Timeshift* также весьма разумно берет на себя переустановку *Grub*. Дело в том, что при установке в MBR образ *Grub* указывает на определенный сектор жесткого диска, где должна находиться остальная часть *Grub*, и если *Timeshift* восстанавливает только часть *Grub*, то, из-за причуд файловой системы, в конечном итоге они ока-жутся в другом разделе жесткого диска, и указанная система станет незагружаемой.



По Mint'овски

Как и его ботанический тезка, Mint [англ. мята] успокаивает желудки пользователей, пострадавших от неудобоваримых дистрибутивов...

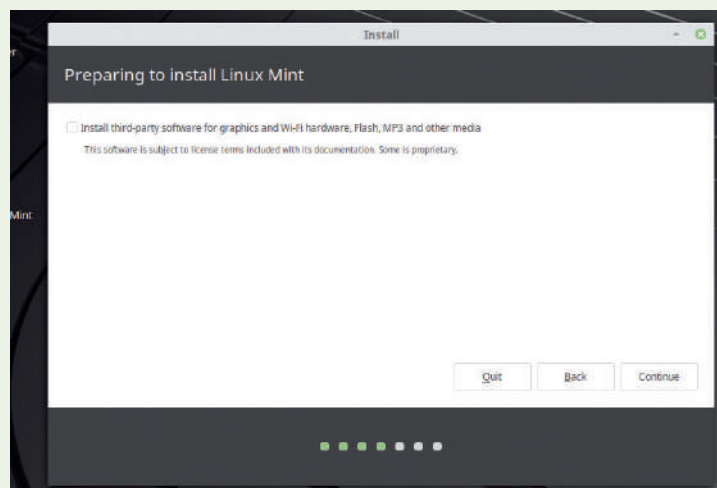
Двенадцать лет назад, когда Gnome 3 был просто зло-радным огоньком в глазах вредного разработчика, хакер и автор статей про Linux по имени Клеман Лефевр выпустил, без особых ожиданий или воодушевления, респин Kubuntu. Он назвал его в честь домена, где размещал свои статьи, Linux Mint, и дал этому релизу кодовое имя Ada.

» ЧТО В (СТОРОННЕЙ) КОРОБКЕ?

Во время установки у вас спросят, хотите ли вы установить сторонние программы. Если вы похожи на нас, то всегда будете отвечать отрицательно, а через некоторое время жалеть о содеянном, обнаружив какой-то сайт или что-нибудь еще, где они понадобятся для воспроизведения видео. Если во время установки вы отвечаете на этот вопрос утвердительно, то устанавливается мета-пакет *mint-meta-codecs*. В противном случае вы можете либо установить этот пакет сами, либо использовать удобный ярлык в разделе Sound and Video [Звук и Видео] главного меню. Однако помните, что заодно установится ужасный плагин Adobe Flash (хотя в Firefox он будет отключен).

Помимо воспроизведения web-видео (которым занимаются *Gstreamer* и *libav*), этот мета-пакет устанавливает инструменты для работы с несвободными форматами архивов (*unshield*, *cabextract* и *unrar*) и воспроизведения DVD. Достаточно сказать, если вас интересуют только некоторые из этих вещей, выясните зависимости мета-пакета с помощью `apt-cache show mint-meta-codecs` и просто установите то, что нужно.

В Ubuntu и предыдущих версиях Mint используется сторонняя опция third-party для установки пакета *ubuntu-restricted-extras* со многими такими сторонними пакетами, включающего также *Fluendo* и *LAME* (для воспроизведения и кодирования MP3) и программу установки основных шрифтов Microsoft. В былые дни предлагался также web-плагин Icedtea Java, однако сейчас очень мало сайтов используют на клиентской стороне Java. А если подумать, так никто не использует и MP3 или DVD...



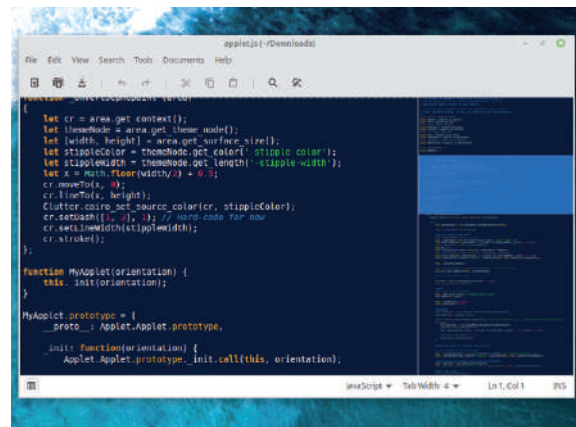
Не делайте так, если только ваш Wi-Fi или подборка мультимедиа этого не требуют.

В его скромном сообществе фанатов возник интерес, и к 2007 г. (с выходом Mint 2.2, "Bianca") любительский дистрибутив превратился в нечто более серьезное. К этому времени устоялась традиция давать релизам Mint женские имена, заканчивающиеся на "a" и в алфавитном порядке, совпадающем с порядковым номером основной версии. В базу Ubuntu были добавлены разные новые приложения, простой инструмент настройки, укрощающий монстров беспроводного Интернета и управления питанием. И самое примечательное — его уникальная поддержка мультимедиа: Flash, воспроизведение DVD, Quicktime и даже Realplayer работали по умолчанию. Автор Distrowatch, Ладислав Боднар [Ladislav Bodnar], описал его в **LXF94**, используя волшебный эпитет «просто работает» — оно и сегодня-то редкость, а уж в те времена — сущий Святой Грааль.

Единство взглядов

Когда в 2011 г. Ubuntu запустил свой рабочий стол Unity, интерес к Mint взлетел до небес. Это был дистрибутив с традиционным рабочим столом, дружелюбным сообществом и (скандал!) разработчиками, которые прислушивались к пользователям. Экосистема рабочего стола Linux уже больше не была, как раньше, оплотом двух: немалую часть отобрал рабочий стол Mint, Cinnamon. Он становился всё лучше и лучше, а последний релиз — вообще фантастика. Однако Mint является — и всегда являлся — не просто Ubuntu с дружелюбным рабочим столом (на самом деле, с тремя рабочими столами), хотя бы уже потому, что имеется редакция Debian.

Linux Mint Debian Edition существует в основном как резерв на случай, если Ubuntu вдруг исчезнет (что выглядит маловероятным) или окажется привязан к определенному рабочему столу (что маловероятно, потому что большей частью своей популярности он обязан наличию идентичных пакетов для Ubuntu Server). Вдобавок то, что это возобновляемый релиз на базе Debian Testing, нравится пользователям-экстремалам. Поэтому когда выйдет LMDE 3 "Cindy" (LMDE традиционно называется женскими именами, заканчивающимися на букву "y"), ждите длинных



Xed — традиционный текстовый редактор с меню и, с недавнего времени, чудесной мини-картой открытого файла справа.



Блог Mint приводит несколько изображений в Xreader старых PDF LXF; это — одна из наших любимых обложек.

номеров версий и шероховатостей. Мы не колеблясь рекомендуем основную версию и начинающим, и опытным пользователям, однако явно существуют те, кто (раздраженные ее удобством для пользователя) при ее упоминании воротят носы. LMDE понравится как раз таким, да, собственно, и прочим пользователям — хотя бы потому, что этот дистрибутив компактнее и быстрее своего основного собрата.

В Mint 18 появились X-Apps, основные утилиты, созданные взамен основных предложений Gnome, которые в последнее время всё больше привязываются — как по стилю, так и функционально — к рабочему столу Gnome. Бригада X-Apps состоит из текстового редактора (*Xed*), просмотрщика изображений (*Xviewer*), видеоплеера (*Xplayer*), читалки PDF (*Xreader*) и органайзера фото (*Pix*). Все они — ответвления более традиционных приложений Gnome со времен до меню в виде гамбургеров и строк заглавия. Фактически, *Xed* и *Xreader* — это ответвления *Mate Pluma* и *Atril* соответственно, которые, в свою очередь, ответвились от их эквивалентов Gnome, пока всё не пошло в новом стиле.

Отмечаем крестиком

X-Apps существуют отнюдь не для того, чтобы снабжать основными утилитами Mint или Cinnamon. Они созданы на благо любого дистрибутива или рабочего стола. Существует также вопрос о будущем Linux Mint. Инструментарий *GTK3* не зависит от рабочего стола, но остается вероятность, что *GTK4* (который выйдет не скоро) будет в основном инструментарием чисто для Gnome. Может, так и не будет, а может, даже если и будет, то неважно. *GTK3*, на котором основаны Cinnamon, Mate и Xfce, вероятнее всего, будет поддерживаться еще долго по истечении пятилетнего срока поддержки Mint 19.

Когда *GTK3* станет нерабочим, возможно, появится созданное сообщество ответвление, а то и новый блестящий инструментарий. Как знать. Однако нам известно, что по причине недовольства Gnome рабочий стол Solus OS, Budgie, в следующем релизе не-Gnome перейдет на *Qt5*. *GTK3* прошел через сложную ситуацию, разгневав многих разработчиков поломками API и ABI с каждым новым релизом. Mint предлагает самую последнюю версию, 3.22, и на ответвление *GTK3* есть только планы на очередной основной релиз. Так что после 3.24 всё опять должно стать стабильным.

Возможно, вы помните из нашей статьи об Ubuntu, что Ubuntu 18.04 поставлялся «в основном» с Gnome 3.28, с некоторыми компонентами, основанными на предыдущей версии, 3.26. Например, *Files* (бывший *Nautilus*), вышел в виде Snap на среде исполнения

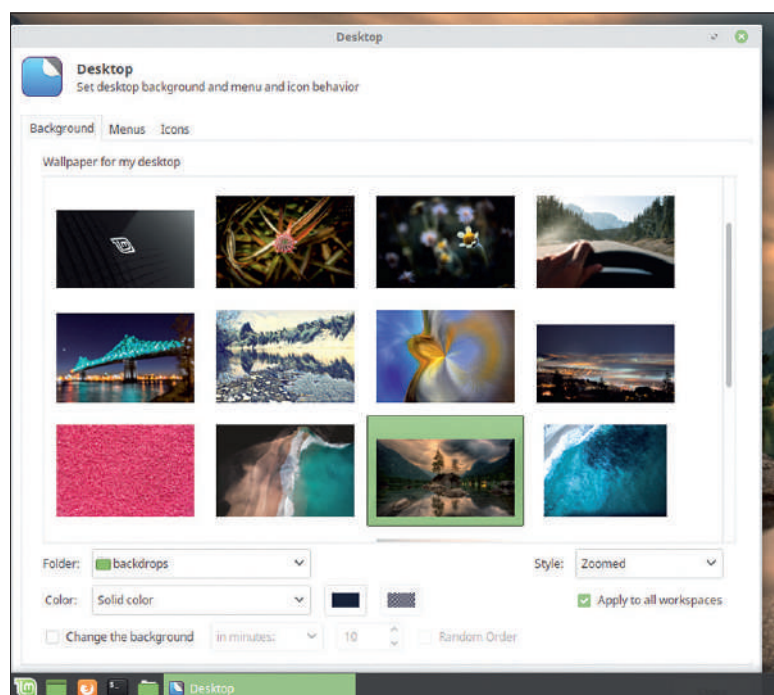
3.26 — для удобства пользователей, желавших побаловать себя вульгарнейшей практикой: размещением значков на рабочем столе. Уже некоторое время стало известно, что Gnome 3.26 станет последним с поддержкой значков на рабочем столе, так что пользователям Ubuntu после 18.04 придется сменить привычки... или рабочий стол.

Cinnamon, для сравнения, уважает все привычки своих пользователей, без предубеждений. Допустим, в наши дни весьма непросто реализовать рабочий стол как директорию (<https://gitlab.gnome.org/Gnome/nautilus/issues/158>), и это может привести к ужасной дезорганизации рабочего процесса, но пользователи настольных ПК (зло)употребляют им именно в таком виде начиная с Windows 95 (и даже дольше, если учитывать инструменты вроде *Norton Desktop*, см. <http://toastystech.com/guis/ndw.html>); а пользователи Mac могут позлорадствовать, что у них эта функция существует с 1980-х. Как и в Gnome, всё, что связано с поддержкой значков на рабочем столе, передано файловому менеджеру, в данном случае — *Nemo*. Точно так же в Mate этим занимается *Caja*.

ЧТОБЫ РАБОЧИЙ СТОЛ БЫЛ ЖИВЫМ

«X-Apps не для того, чтобы снабжать утилитами только Mint или Cinnamon. Они созданы на благо любого дистрибутива или рабочего стола.»

Fedora продвигает Wayland-по-умолчанию, но в Mint вы не найдете упоминания графической подсистемы следующего поколения, API композиции и т.п. Пакеты доступны, но Mint не намерен навязывать пользователям то, чего они не хотят или то, что не работает. LXF



А вот обязательный скриншот с красивыми новыми фонами рабочего стола. Каждый найдет здесь что-нибудь себе по вкусу — полагаем, вы с этим согласитесь.

FreeBSD стукнуло 25

Валентин Сеницын в первую очередь является специалистом по Linux, но по этому случаю он находит в своем багаже некоторые навыки в BSD.

В 2016 г. Linux исполнилось 25. А сейчас, пока я пишу эти строки, FreeBSD тоже празднует четвертьвековой юбилей. Некоторые из нас предпочитают считать FreeBSD младшим братом Linux (а некоторые — наоборот), и если это не доказательство, то чего вам больше?

Возможно, вы также слышали, что BSD — это «настоящий Unix», тогда как Linux всего лишь «Unix-подобен». Это не совсем правда. Список официальных Unix ведет The Open Group, и туда не включен ни Linux, ни BSD; однако macOS в него входит. Но при этом никто не будет спорить, что BSD уходит своими корнями глубоко в историю Unix.

» ДЕЛАЙТЕ КОД, А НЕ ВОЙНУ

В июне 2018 г. мы отмечали 25-ю годовщину FreeBSD, и я здесь шучу насчет жарких баталий в сообществе Свободного ПО. Однако обязательны ли противостояния? Не знаю, как вы, а я по большей части перестал читать комментарии на технических ресурсах, куда я захожу. Причина в том, что, похоже, никому нет дела до самой истории, а только до имен собственных в ней. Остаток дискуссии сводится к тому, нужен ли нам <вставьте имя проекта>.

Если вы считаете, что это обычный интернет-троллинг, не могу вполне с этим согласиться. По неким причинам наше сообщество склонно поощрять подобную «конкуренцию», по крайней мере, вблизи от меня. Сам я ничего не имею против конкуренции как таковой, ведь она ведет к появлению лучших продуктов; но это не тот тип конкуренции. Неужели миру, изобилующему серверами Linux, не обойтись без FreeBSD?

Несколько лет назад ответом было уверенное «да». FreeBSD был единственным реальным выбором для желающих попробовать ZFS или *DTrace* вне Solaris. Что обеспечивалось лицензией BSD, по крайней мере, на некоторое время; вспомните об этом в следующий раз, когда кто-нибудь затеет спор о том, достаточно ли она свободна, чтобы иметь право на жизнь. Теперь, когда *DTrace* официально получил лицензию GPL (**LXF236**), и после недавних изменений в файловой системе, ответ уже не кажется столь очевидным, но я уверен, что он по-прежнему «да».

Может, FreeBSD и старая ОС, с отчасти устаревшим кодом и т.д. Linux, быть может, технарский, недружелюбный к пользователю (или, как минимум, слишком разборчивый в выборе друзей) и занимает всего несколько процентов в базе данных пользователей настольных ПК. В конечном итоге, всё зависит от вашего выбора: вокруг меня разновидности Unix вот уже несколько лет как побили всех остальных. Нам вовсе не нужно кольцо всевластия: пусть расцветают все цветы.

Если кто не в курсе, BSD означает Berkeley Software Distribution, а ведет он начало из 1970-х. Те, кто сейчас подумал: «да ладно, это и дети знают!», пусть обратятся к менее известным фактам (и мы имеем в виду программные сокет Berkeley). *Vi* пришел из BSD: его (вместе с *ex*) разработал Билл Джой [Bill Joy], тогда выпускник Университета Беркли для Second Berkeley Software Distribution (2BSD), выпущенного в 1979 г. Теперь вы поняли причину долгосрочного противостояния *Vi* и *Emacs*: это, по сути, противостояние Linux и BSD, только наблюдаемое под другим углом. И второе: вы когда-нибудь задумывались, почему образ ядра Linux называется «vmlinuz»? Ну, «z» — это фактически стандартный суффикс сжатия. Распакуйте его, и вы получите «vmlinux», явное перефразирование «vmunix». Последнее переводится как «virtual memory Unix» — напоминание о временах, когда студенты-«берклианцы» реализовывали поддержку виртуальной памяти для компьютера VAX, приобретенного Университетом Беркли в 1978 г.

Так что произошло 19 июня 1993 г.? Как и 25 августа 1991 г., эта дата связана с рассылкой по электронной почте. Тогда ответвление 4.4BSD принимало решение о названии. Цитируя Фила Карлтона [Phil Karlton], «в информатике есть только две сложности: аннулирование кэша и достойное именование». В данном случае труднее всего казалось втиснуть в название слово «free», чтобы выложить на стол нечто вроде BSDFree86 и Free86BSD (кстати, кто-нибудь помнит XFree86?). Потом Дэвид Гринман [David Greenman] спросил: «А может, просто FreeBSD?».

И вы знаете, это сработало! 25 лет спустя FreeBSD стоит у корней кустистого дерева ОС со множеством «вариантов» BSD. NetBSD и OpenBSD находятся ближе к верхушке. Относительные новички, такие как DragonflyBSD, а теперь еще и TrueOS (ранее PC-BSD), уже доказали свою жизнеспособность. Bravo, и с днем рождения, FreeBSD!



19 июня отныне будет официальным Днем FreeBSD Day. Узнайте, как правильно его отмечать, на www.freebsd.foundation.org/national-freebsd-day.



**НАШ
ЭКСПЕРТ**

Д-р Валентин Сеницын
Разработчик
облачной инфра-
структуры
в Yandex днем,
активный участ-
ник сообщества
открытого кода
ночью, интере-
сующийся всем,
от АН до X509.

Scapy: Разработка пакетов в Python

Говорят, всему свое время. Но сегодня не время сажать или вырывать посаженное; время создавать и контролировать пакеты.

По части перехвата пакетов стоит обратиться к *tcpdump* или *Wireshark*. Эти инструменты употребляют библиотеку *libpcap*, чтобы вытащить пакеты из сети или прочитать и сохранить их в так называемых файлах PCAP. Механизм фильтров, например, BPF, отсекает искомые пакеты. *tcpdump* в некотором роде низкоуровневый; *Wireshark* приятнее для глаз и добавляет больше визуальных подсказок — например, выделение сегмента TCP, выбивающегося из порядка. Несмотря на мелкие различия, вкратце, все эти инструменты контролируют пакеты, декодируют протоколы и представляют их вам для анализа.

А если пассивного перехвата пакетов недостаточно? Например, вам надо имитировать сетевую функцию — хотя бы шлюз SNAT. Или — вы тестируете брандмауэр, чтобы убедиться в правильности его работы. В любом случае, вам требуется средство для формирования пакетов и отправки их в сеть. Чем и занимается Scapy (<https://scapy.net/>).

Взламываем оболочку

Как вы, вероятно, догадались по суффиксу, *Scapy* относится к Python. Точнее, это библиотека и инструмент командной строки, который предоставляет среду Python для скриптов, связанных с сетевыми задачами. Однако *Scapy* активно использует волшебные методы Python, чтобы помочь вам сосредоточиться в большей степени на сети и в меньшей — на Python. В том и заключается философия *Scapy*: предоставить вам стройматериал, достаточно гибкий для создания инструмента обработки пакетов, который вам нужен, всего несколькими строками кода. Более того, *Scapy* пытается не навязывать определенной интерпретации того, что видит: он просто кодирует или декодирует байты и рассчитывает, что вы сами уясните их значение.

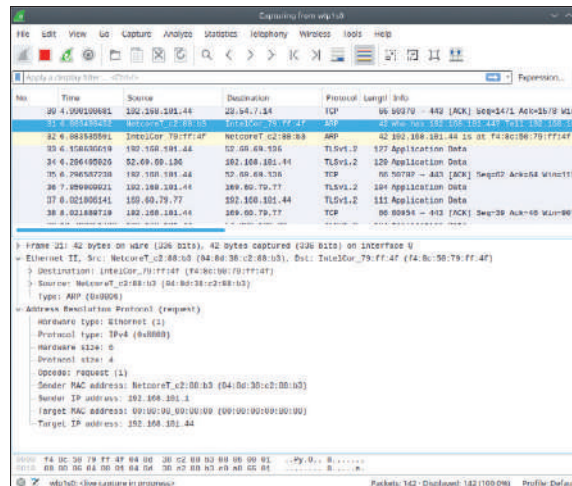
С учетом происхождения *Scapy* от Python, неудивительно обнаружить, что он устанавливается из PyPI с помощью *pip* (возможно, в виртуальной среде [virtualenv]). Однако это не единственный способ: код доступен на GitHub, и вы, вероятно, найдете *Scapy* в своем менеджере пакетов. Ubuntu называет его *python-scapy* или *python3-scapy*. Да, поддерживаются и Python 2.7 и 3; однако Ubuntu (даже самый последний и самый превосходный Bionic Beaver) предлагает *Scapy* 2.3, вышедший в 2014 г. У последнего есть ошибка в адресном анализаторе IPv6, способная не дать вам запустить интерактивную оболочку. В таком случае подумайте об использовании вместо него версии *pip*.

Чтобы запустить интерактивную оболочку *Scapy*, просто введите

```
$ sudo scapy
```

Перемалывание байт в Linux не является привилегированной операцией, а вот захват сырых пакетов из сети или их возврат — является. Поэтому обычно вам понадобится *sudo* для ваших скриптов *Scapy* или интерактивных сессий. Если вы установили *Scapy* в *virtualenv*, помните, что *sudo* для каждого пользователя свой, и соответственно настройте *sudo*.

Некоторым функциям *Scapy* могут понадобиться добавочные пакеты, от скромных *matplotlib* до целого дистрибутива LaTeX. Обнаружив их отсутствие при запуске, *Scapy* печатает предупреждение; пока что такие предупреждения можно спокойно игнорировать, поскольку основная функциональность этим не затронута. Имеет смысл установить IPython, поскольку он обогащает вашу



Wireshark — свободный (как слово) дружелюбный к пользователю анализатор пакетов, доступный в версиях на GTK и Qt.

интерактивную работу со *Scapy* выделением синтаксиса, автозаполнением, журналом и прочими благами.

Формирование нескольких пакетов

С учетом того, что *Wireshark* выглядит всезнающим, вы можете заинтересоваться, сколько сетевых протоколов (известных как «слои [layers]» или «уровни») понимает *Scapy*. Вот ответ:

```
>>> ls()
AN : AN
```

» ФИЛЬТРЫ BPF

Сегодняшние сети — довольно болтливые создания. Мой простаивающий ноутбук с Linux наблюдает несколько сотен пакетов каждые пять секунд. Представьте же себе, что видит активный сервер, подключенный к высокоскоростной сети центра обработки данных!

Захват всех пакетов в подобной ситуации выглядит довольно непрактичным. Вот почему большинство программ захвата [sniffer] реализуют некий язык фильтров, чтобы сузить поиск. Фактическим стандартом здесь служит т.н. фильтр BPF. Имя отчасти вводит в заблуждение: на самом деле, Berkeley Packet Filter (BPF) — низкоуровневый язык типа Assembler. Однако термин 'BPF filter' обычно применяют к высокоуровневому мини-языку, реализованному в *libpcap*, как подсказывает map-страница 'pcap-filter(7)'. Тем не менее, фильтры BPF — не то же самое, что фильтры отображения в *Wireshark*, хотя те тоже поддерживаются (они называются фильтрами захвата [capture filters]).

Недавние версии 'Scapy()' принимают фильтры BPF в качестве аргумента ключевого слова 'filter' к функции 'sniff()'. За сценой вызывается *tcpdump -ddd*, чтобы скомпилировать фильтр в низкоуровневое представление. Скомпилированные фильтры исполняются внутри ядра и часто транслируются в родной код, что делает их высокопроизводительными. BPF и eBPF (см. **LXF231**) различаются, хотя первого можно считать предшественником второго.

Большинство фильтров BPF не сложнее, чем *host 192.168.0.1* или *tcp port 80*, но это только вершина айсберга. Вы можете индексировать произвольно выбранные байты в пакете, использовать арифметические и булевы операции, и т.д. А вы знали, что запуск выражения *tcpdump -i any -d filter* даст вам человеко-читаемое представление скомпилированного кода BPF?

ARP : ARP

Каждый протокол — по сути, класс Python: вы создаете экземпляр для формирования пакетов и получаете эти экземпляры назад, делая захват из сети с помощью `sniff()`. Атрибуты объектов проецируются прямо на поля заголовков протокола. Так же и `ls()` работает на уровне классов, и в этом случае показывает все поля протокола и их типы.

Сетевые протоколы обычно группируются поверх друг друга (отсюда и «слои»). В *Scapy* вы это делаете с помощью знака деления:

```
>>> IP(dst="1.1.1.1") / UDP(dport=53) / DNS(rd=1,
    qd=DNSQR(qname="www.linuxformat.com", qtype="A"))
```

Говорит само за себя, верно? Создается запрос DNS для рекурсивного разрешения www.linuxformat.com с помощью одного из открытых DNS-серверов. Параметры сетевого и транспортного уровней тоже говорят сами за себя. Однако заметьте, что исходный IP-адрес не указан: *Scapy* достаточно умен, чтобы угадать его самому при отправке пакета — он даже поддерживает свою таблицу маршрутизации! Вы же занимаетесь анализом, а не рутинной работой. Параметры DNS могут показаться загадочными, но на самом деле являют собой просто поля протокола, в основном описанные в RFC 1035. И опять *Scapy* предлагает помощь, но не пытается умничать: протоколы интерпретируете вы, так что знание их внутренностей является обязательным.

Помните, что созданный пакет — просто объект Python, так что вы можете хранить его как переменную и вообще делать с ним что угодно. Например, можно индексировать пакет именем уровня:

```
>>> p = _
>>> p[UDP]
<UDP sport=domain dport=domain l<DNS rd=1 qd=<DNSQR
    qname='www.linuxformat.com' qtype=A l> l>>
```

Символ `_` относится к последнему результату в оболочке Python. Слои нумеруются от самого внешнего к самому внутреннему, поэтому `p[2]` и `p[DNS]` в данном случае одно и то же. Если вы индексируете пропущенный уровень — скажем, `p[TCP]` — *Scapy* выдаст `IndexError`. Чтобы этого избежать, проверьте наличие уровня с помощью метода `haslayer()`:

```
>>> p.haslayer(TCP)
0
```

Особенным уровнем является Raw. Это байты, которые *Scapy* не распознает, и иногда бывает полезно поработать с протоколами уровня приложения, которые *Scapy* (пока) не поддерживает.

Полубуйтесь, как красиво интерактивная оболочка *Scapy* печатает для вас пакеты. Желая самому добиться таких же результатов, просто введите `repr(p)`, `str(p)`, что даст вам двоичное представление пакетов на уровне байтов.

Есть ряд других способов визуализировать уровни пакетов. Во-первых, это метод `show()`, который работает также и над списками пакетов (см. ниже). По сравнению с `repr()`, он печатает (или возвращает по `dump=True`) пакет, как структурированную иерархию. `show2()` делает то же самое, но заодно заполняет автовычисляемые поля, такие как контрольные суммы. `summary()` предоставляет вам однострочную сводку по пакету со всеми пропущенными деталями. Еще один однострочник, `command()`, возвращает выражение Python/*Scapy*, пригодное для ручного формирования пакета:

```
>>> p.command()
IP(dst='1.1.1.1')/UDP(dport=53)/DNS(rd=1, qd=DNSQR(qtype=1,
    qname='www.linuxformat.com'))
```

Получается отличный обучающий инструмент: вы захватываете пакеты из сети — или делаете PCAP файла с помощью `rdpcap()` — и смотрите, как воссоздать их вручную.

Возможности *Scapy* простираются далеко за пределы текста. Установив *PyX* (<http://pyx.sourceforge.net>) и LaTeX, вы можете визуализировать пакеты в виде диаграмм. Это, опять же, отличное обучающее средство, поскольку вам показывается, как определенные байты пакетов отражаются в полях протокола; но это также и мощное вспомогательное средство для создания презентаций в *LibreOffice Impress* (или любой другой программе).

В приведенных примерах нас волновал только один пакет. *Scapy* также способен генерировать серии пакетов. Каждое поле в пакете может быть набором (`dport=[80, 443]`), и вы можете использовать специальные символы или префиксы вместо простого сетевого адреса (например, сработают и `192.168.0.*`, и `192.168.0.0/24`). В порядке альтернативы можете задействовать `fuzz()`, которая устанавливает в поля протоколов случайные значения.

Отправка и получение

Итак, у вас есть представление о том, как формировать пакет в *Scapy*, и пора предпринять несколько настоящих сетевых путешествий. Здесь вам снова понадобятся привилегии суперпользователя, поэтому перезапустите сессию *Scapy* от имени root, если вы этого еще не сделали.

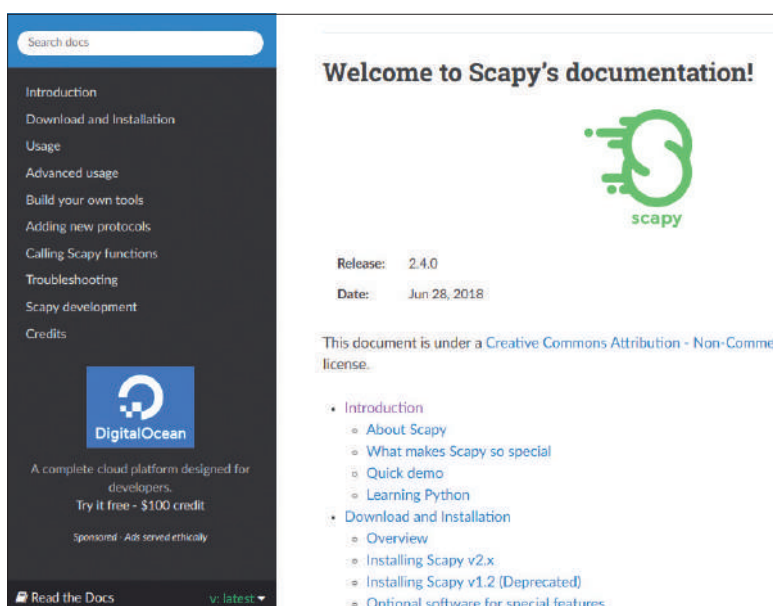
Самый простой способ закинуть несколько пакетов — функция `send()`:

```
>>> send(p)
```

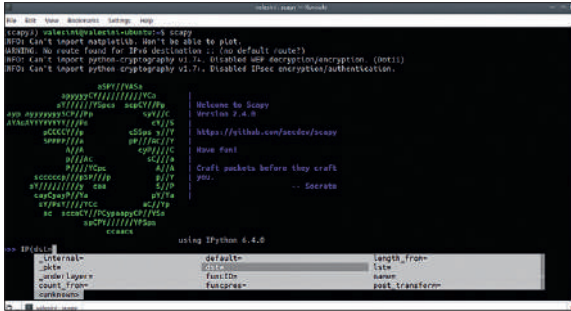
```
Sent 1 packets.
```

Функция работает на уровне Layer3: например, она добавляет заголовок Ethernet и выполняет маршрутизацию. Функция-аналог, `sendp()`, работает на Layer2 и рассчитывает, что вы предоставите Ethernet- или другой заголовок L2. `send()` принимает несколько аргументов ключевых слов, таких как счетчик пакетов, или отправлять ли пакеты в бесконечном цикле. Она также печатает точку после каждого отправленного пакета, так что вы можете оценить темп и количество отправленных пакетов.

Если вы что-либо отправляете, есть вероятность, что вам требуется ответ. Функция `sr()` делает именно это: она отправляет пакет, ждет ответа — время ожидания настраивается — и возвращает ответ вам. Точнее, она возвращает в ответ несколько соответствующих пакетов вместе с пакетами, оставшимися без ответа. `sr1()` — ее вариант, который захватывает один ответ на один пакет.



Как и во многих нынешних проектах с открытым кодом, самая свежая документация *Scapy* доступна онлайн на сайте Read The Docs.



IPython – не жесткое требование для Scapy, но помогает интерактивным сессиям работать ровнее, как в IDE.

```
>> sr(p)
```

```
Begin emission:
```

```
Finished sending 1 packets.
```

```
.....
```

```
Received 7 packets, got 1 answers, remaining 0 packets
```

```
(<Results: TCP:0 UDP:1 ICMP:0 Other:0>,
```

```
<Unanswered: TCP:0 UDP:0 ICMP:0 Other:0>)
```

Функция отправляет пакет, затем следит, пока не получит в ответ нечто подходящее.

Когда вам надо только слушать пассивно, подумайте о `sniff()`. Вы можете настроить ее для хранения захваченных пакетов для дальнейшего анализа, или простой их обработки и удаления. Можно также указать Scapy, что именно захватывать с помощью фильтра BPF. По умолчанию `sniff()` работает «бесконечно», пока вы не нажмете Ctrl+C, но вы можете задать собственное «определение готовности». Самое простое — задать для `sniff()`, сколько пакетов следует обработать. Для более расширенных сценариев должен сработать 'stop filter' — это вызываемая функция Python, возвращающая 'True', когда `sniff()` должна остановиться.

Теперь рассмотрим такие два примера:

```
>>> captured = sniff(filter="ip", store=True, stop_filter=lambda p: p.haslayer(UDP))
```

```
<Sniffed: TCP:3 UDP:1 ICMP:0 Other:0>
```

```
>>> sniff(store=False, prn=lambda p: p.summary())
```

```
Ether / IP / UDP 77.88.6.88:1199 > 192.168.101.44:36626 / Raw
```

```
...
```

Первая команда захватывает IP-пакеты, сохраняет их в буфер и останавливается, как только получает датаграмму UDP. Чтобы получить эту датаграмму, вы индексируете результат следующим образом: `captured[UDP][0]`. Вторая команда ничего не сохраняет, она просто печатает обзор пакетов, схожий с `tcpdump`. `sniff()` в этом случае по-прежнему возвращает полученный в результате объект, но он всегда пуст.

Обратите внимание на аргумент `prn`, который получает вызываемую функцию, как `stop_filter`. Несмотря на название, печать он не ограничивается. Например, вы можете контролировать пакеты MPLS, выудить заголовок и затем поместить внутренние пакеты в устройство настройки, превратив свой компьютер с Linux в простейший роутер Provider Edge (PE). Чтобы это сработало, вам нужен один недостающий фрагмент: Scapy не поддерживает MPLS по умолчанию.

Хорошая новость: исправить это легко. Вам надо просто сделать подкласс `Packet`, дать своему уровню имя и создать атрибут `field_desc`, который хранит список полей заголовков протоколов. Для MPLS стандартные описатели поля, такие как `BitField` или `ByteField`, должны отлично подойти. Последнее, что надо сделать — вызвать `bind_layers()`, сообщив Scapy, что MPLS обычно следует заголовку Ethernet с EtherType, установленным на 0x8847,

» ОДИН SCAPY ПРАВИТ ВСЕМ

Обычно люди используют для решения своих сетевых проблем множество инструментов: *Ping* для простой проверки соединения, *traceroute* и *tracert* для выявления нерабочих переходов, *tcpdump* в качестве анализатора. Является ли Scapy универсальным инструментом, способным заменить их все?

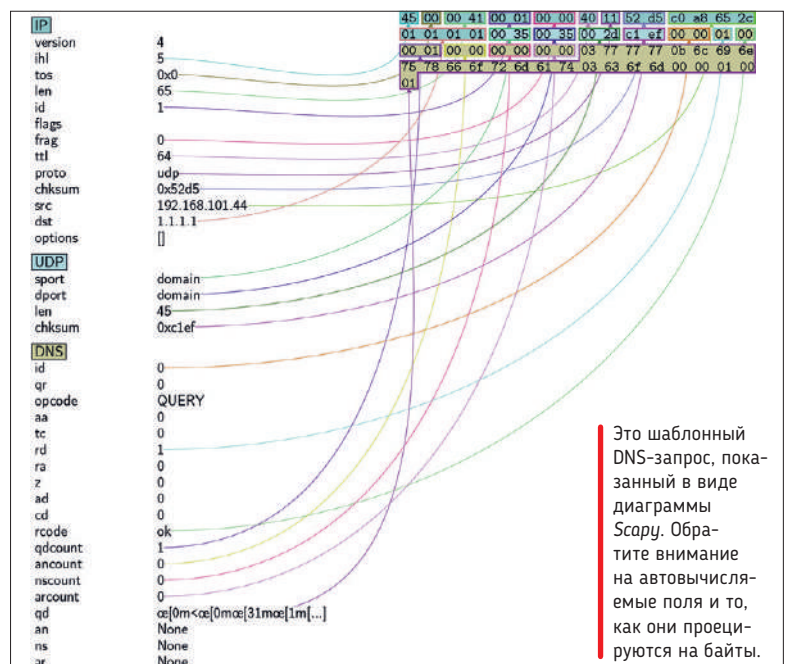
Ну, не совсем, но попробовать можно. Наличие Scapy в вашем арсенале не означает, что вы не должны использовать другие инструменты, но выдавать subprocess.Popen() каждый раз, когда вам нужен *traceroute*, и затем анализировать результаты тоже не очень здорово. Вместо этого Scapy заново реализует (или оборачивает) множество популярных сетевых инструментов и предоставляет их готовыми к использованию.

Возьмем `scapy.utils.tcpdump()`. Эта функция принимает список пакетов (захваченных, созданных или иных), отправляет его утилите *tcpdump* (или *tshark*) и возвращает вам результат. Внутри список пакетов конвертируется во временный файл `pcap`. Это очень удобно для периодического декодирования протоколов, которые Scapy по умолчанию не понимает. Имея внутри *tshark*, вы даже можете дать указание `tcpdump()` вернуть вам JSON для дальнейшей обработки. Чтобы узнать детали, запустите `help(tcpdump)`.

Еще одна интересная команда — `traceroute()`. Как подсказывает ее название, она вызывает *traceroute* (TCP) между вами и указанным набором хостов. В отличие от множества инструментов, `traceroute()` сканирует все хосты параллельно. Команда реализована с помощью `sr()` и фильтров BPF, и является хорошим примером того, как программировать «в Scapy». По умолчанию она выводит простой текст, но если установить соответствующие пакеты Python, сможет создавать графики или даже 3D-визуализации (хотя на момент написания последнее, похоже, не работало).

или UDP, предназначенным порту 6635 (это MPLS через UDP, см. RFC 7510).

Возможно, вы не хотите делать всё это из интерактивной оболочки — хотя это и возможно. Если так, просто введите `quit()` или нажмите Ctrl+D и запустите собственный отдельный инструмент Scapy! MPLS — простой протокол, так что мы оставляем точную реализацию на ваше усмотрение в качестве упражнения. **LXF**



Это шаблонный DNS-запрос, показанный в виде диаграммы Scapy. Обратите внимание на автовычисляемые поля и то, как они проецируются на байты.

» ПОДПИШИТЕСЬ НА ЖУРНАЛ LINUX FORMAT на www.linuxformat.ru/subscribe!

СКАЖИ УЧЕБЕ

ДА!



**СЕРВЕРНЫЕ РЕШЕНИЯ
LINUX
SOLARIS
ХРАНЕНИЕ ДАННЫХ
JAVA
ANDROID
БЕЗОПАСНОСТЬ
ЗАЩИТА ДАННЫХ**

Санкт-Петербург,
ул. Яблочкова, 12, литер Ц
(812) 611 1575
unixedu.ru

UnixEducationCenter

Учебники



Евгений Балдин
Подтвердивший
свою квалификацию
физик.

» УЛИЧНАЯ ТОПОЛОГИЯ ГОРОДОВ

Все вокруг — геометрия!
Ле Корбюзье

Джефф Боинг [Geoff Boeing], постдок из лаборатории городского планирования Калифорнийского университета в Беркли, предложил простой метод визуализации топологии улиц в городах посредством радиальных гистограмм. В поддержку своего метода он выложил диаграммы нескольких десятков крупных городов обычным людям на забаву.

Если при строительстве города планировщики хоть немного думали об удобстве жителей, то топология городских улиц представляет собой ориентированную прямоугольную решетку. На диаграмме это выглядит как крест, с пиками вдоль осей симметрии. Города с долгой историей или просто плохо спроектированные не имеют какой-либо выделенной структуры, и их диаграмма похожа на круг без выделенных направлений.

Исследователь выложил исходный код для создания такого рода картинок на GitHub под лицензией MIT в виде файлов Jupyter Notebook (интерактивная среда и IDE для Python). Для получения информации по улицам и их направлениям использовались карты свободного проекта OpenStreetMap.

Почему это интересно? Во-первых, потому, что картинки действительно занимательные. Но самое главное заключается в том, что исходный код может взять относительно неопытный в программировании человек (Jupyter Notebook действительно удобен для обучения) и построить картинку для своего города. Что, в частности, продемонстрировал руководитель отдела «Шапито» сайта Meduza Султан Сулейманов, опубликовавший диаграммы крупнейших российских городов. E.m.Baldin@inp.nsk.su

В этом месяце вы научитесь...

ТЕРМИНАЛ

Предсказывать погоду 54
Правда, не более чем на пять дней вперед, предупреждает **Шашанк Шарма**, запуская замечательную утилиту *Wego* (с тремя движками и интерфейсами).

СТРУКТУРА СИСТЕМЫ

Лазать по деревьям 56
Linux организует файлы в древовидную структуру, а **Ник Пирс** осуществляет по ней навигацию с помощью менеджера файлов *Nautilus*. Пока ни разу не заблудился.

ОБРАБОТКА ВИДЕО

Кодировать видео 58
Без **Джона Найта** вам это будет непросто: программа *HandBrake* создана для опытных пользователей, кодеки там всякие... А вот с ним у вас появится шанс!

ПЕРИФЕРИЯ СИСТЕМЫ

Подключать Bluetooth 62
Ник Пирс взялся примирить Linux с традиционно капризным Bluetooth. Игра стоит свеч, ведь это идеально для подключения беспроводных устройств ввода к вашему ПК.

ЭЛЕКТРОННЫЕ КНИГИ

Писать книги 66
Осваивать редактор электронных книг *Sigil* **Джону Найту** показалось трудновато. Но потом он с изумлением понял, что зато получаемый продукт уже готов для магазина.

ЛЮБИТЕЛЬСКОЕ РАДИО

Ловить радиостанции 70
Получите у **Майка Бедфорда** отбивать морзянку и настраиваться на любительские радиостанции со всего мира. Нет, коротковолновое радио отнюдь не умерло.

АКАДЕМИЯ КОДИНГА

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

МАШИННОЕ ОБУЧЕНИЕ

Бот ученый 74
Системные журналы, в поисках попыток взлома, за **Дэна Фроста** просматривает бот! Но это не чудо из чудес, такого бота вы и сами можете поднатаскать.

WEB-РАЗРАБОТКА

Смотреть везде 78
Один и тот же сайт может быть одинаково приятен для просмотра и на компьютере, и на смартфоне — это если он адаптивный, как у **Кента Ельчука**.

ЯЗЫК ПРОГРАММИРОВАНИЯ

Пробежка со Swift'ом 82
Михалис Цукалос рассматривает особенности версии языка программирования Swift 4. Видимо, новшества в первую очередь обрадуются профессионалы.

TERMINAL: WEGO

Прогноз погоды через терминал

Иные фанаты Bash не хотят покидать терминал, даже чтобы проверить прогноз погоды. **Шашанк Шарма** не из таких, но знает прекрасный инструмент.



**НАШ
ЭКСПЕРТ**

Шашанк Шарма — судебный адвокат в Дели и рыный пользователь Arch Linux. Он всегда в поиске экзотических технарских сувениров.

Наряду со статистикой системного мониторинга использования ЦП и ОЗУ, многие пользователи очень любят проверять прогноз погоды для своего города. В большинстве настольных дистрибутивов есть для этого графический инструмент; *Weather* — одно из таких популярных приложений, предназначенное для рабочего стола Gnome, но есть и другие. Для пользователей, предпочитающих видеть эту информацию на отдельной вкладке терминала *Bash*, нет лучшей альтернативы, чем *Wego*.

Как следует из названия, утилита написана на языке Go, и *Wego* не имеет других зависимостей. Для отображения значков прогноза погоды, таких как «облачно», «влажно» и «в основном облачно», она использует стиль ASCII. Более того, *Wego* не требует установки, а настройка занимает всего пару минут.

Для начала запустите терминал и выполните команду `sudo apt install golang`, если вы используете Ubuntu, Debian или производный дистрибутив. Пользователи Fedora могут в данном случае запускать команду `sudo dnf install golang`.

Приложения Go устанавливаются в каталог `$GOPATH/bin`, поэтому прежде всего мы должны определить рабочее пространство `$GOPATH` для хранения всех таких приложений Go:

```
$ mkdir ~/gocode
$ export GOPATH=~/gocode
```

Теперь можно загрузить *Wego* командой `get -u github.com/schachmat/wego`, которая автоматически поместит *Wego* в каталог

`$GOPATH/gocode/bin`. В нашем случае *Wego* загружался прямо в каталог `/home/linuxlala/gocode/bin`.

Чтобы запустить утилиту, перейдите в каталог `$GOPATH/bin` и запустите команду `./wego`. Если вы предпочитаете запускать *Wego* так же, как любую другую утилиту командной строки — из любого места в терминале, вы должны обновить соответствующую информацию в файле `~/.bashrc`:

```
$ echo 'export GOPATH=$HOME/gocode' >> ~/.bashrc
$ echo 'export PATH="$PATH:$GOPATH/bin"' >> ~/.bashrc
$ source ~/.bashrc
```

Благодаря этим трем командам вам больше не надо будет перед запуском *Wego* переключаться командой `./wego` в каталог `$GOPATH/bin`. Достаточно команды `wego` из любого места терминала.

При первом запуске *Wego* вместо отображения текущей погоды для вашего города утилита выведет двухстрочное сообщение:

```
2018/07/23 01:16:43 No forecast.io API key specified.
You have to register for one at https://developer.forecast.io/register
[Вам надо зарегистрироваться на https://developer.forecast.io/register]
```

Быстрая настройка

Wego может получать информацию о погоде из трех разных источников, которые называют движками. У вас обязана существовать учетная запись либо на [forecast.io](https://developer.forecast.io/register) (Dark Sky), либо на [OpenWeatherMap](https://openweathermap.org) (openweathermap.org), либо на [WorldWeatherOnline](https://worldweatheronline.com) (worldweatheronline.com).

Инструмент по умолчанию использует [forecast.io](https://developer.forecast.io/register), что и объясняет сообщение, появляющееся при первом запуске *Wego*. Хотя услуги движков имеют еще и коммерческую подписку, для создания бесплатной учетной записи вам достаточно указать действительный адрес электронной почты. Dark Sky ([forecast.io](https://developer.forecast.io/register)), например, отводит на бесплатную учетную запись до 1000 запросов в день, чего должно быть достаточно для большинства пользователей настольных компьютеров. А вот WorldWeatherOnline поддерживает 500 запросов в день и предоставляет 60-дневную бесплатную пробную версию.

Каждая из этих служб после создания учетной записи предоставляет уникальный ключ API, который выглядит примерно как `b7ca7b57d8d10702c01b075b1dfe287b`.

[forecast.io](https://developer.forecast.io/register) не поддерживает географические названия, и вы не можете указать свое местоположение как `location=New York` или `location=Somerset`. Вместо этого вы должны выяснить



Окно терминала по умолчанию слишком мало для правильного отображения в стиле ASCII. Не забудьте расширить окно терминала по горизонтали, ради лучшего результата.

широту и долготу вашего города и указать их в файле `~/wegorc`. Например, широта и долгота Нью-Дели — 28,6139°N, 77,2090°E. Вы вводите эту информацию в файл `~/wegorc` в формате `location=28.6139,77.2090`. Если вы хотите использовать с *Wego* движок по умолчанию `forecast.io`, указывайте широту и долготу для вашего города через запятую.

Создав учетную запись на `forecast.io`, получив ключ API и выяснив широту и долготу своего города, пора будет поместить эту информацию в файл настройки `~/wegorc`, поэтому откройте этот файл в своем любимом текстовом редакторе. Пока нас интересуют только три строки:

```
backend=forecast.io
forecast-api-key=
location=
```

Добавьте ключ API в строке `forecast-api-key=`, а координаты своего города — в строке `location=`. Сохраните файл, чтобы запускать *Wego* командой `wego`.

По умолчанию прогноз дается на 3 дня, но вы можете изменить его максимум на пять дней, отредактировав строку `days=3` в файле `~/wegorc`.

Другие услуги движков, такие как OpenWeatherMap, более гибки в работе с местоположением и даже умеют работать с названиями городов. Если вы решили вместо `forecast.io` по умолчанию использовать OpenWeatherMap или WorldWeatherOnline, измените строку `backend=forecast.io` в `~/wegorc`, отобразив правильное имя движка. Вам также необходимо предоставить ключ API для вашего сервиса движков. Файл `~/wegorc` предусматривает отдельную строку для приема ключей API разных служб. Отредактируйте строку `owm-api-key=` или `wwo-api-key=`, вставив соответствующий ключ API.

Инструмент по умолчанию метрический, но вы можете изменить это на систему Британской империи, изменив строку `units=metric` на `units=imperial` в файле `~/wegorc`.

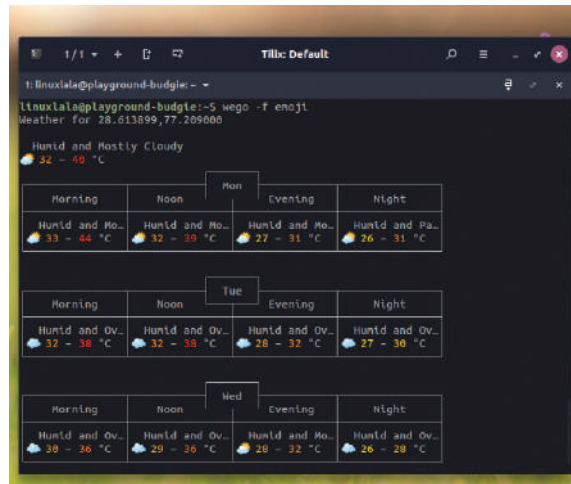
Хотя значение по умолчанию `~/wegorc` не предоставляет такого указания, *Wego* также поддерживает два дополнительных интерфейса, кроме стандартного `frontend=ascii-art-table`.

Чтобы увидеть, как выглядит прогноз с интерфейсом эмодзи, запустите команду `wego -f emoji`. Он понятней, чем интерфейс в стиле ASCII, но не предоставляет важных деталей, таких как скорость ветра или осадки. Если вам нужны все эти данные, придется вернуться к интерфейсу в стиле ASCII.

Третий интерфейс, JSON (JavaScriptObject Notation), удобен разве что для экспорта вывода в другие службы и приложения. Частичный вывод JSON с *Wego* выглядел бы следующим образом:

```
$wego -f json
{
  "Current": {
    "Time": "2018-07-23T17:30:00+05:30",
    "Code": 14,
    "Desc": "clear sky",
    "TempC": 31.56,
    "FeelsLikeC": 32.55,
    "ChanceOfRainPercent": 38,
    "PrecipM": 0.0012471,
    "VisibleDistM": 3110,
    "WindspeedKmph": 11.195999,
    "WindGustKmph": null,
    "WinddirDegree": 118,
    "Humidity": 89
  }
```

Как видно из блока, это вывод только для текущей погоды. Вывод JSON дает аналогичные блоки для каждого слота, а также предоставляет дополнительную информацию, такую как «По ощущению» и «Расстояние видимости», «Вероятность дождя» и т.д. Кроме того, вывод JSON также предоставляет



астрономические подробности, такие как время восхода и захода луны и солнца. Однако вся эта информация зависит от выбранного движка.

Хотя все три движка давали одинаковый прогноз погоды, разница становится очевидной, когда вы используете интерфейс JSON. В наших тестах `forecast.io` предоставил наиболее точные сведения о видимости, вероятности дождя и астрономических данных.

Если вы хотите узнать прогноз погоды в другом месте или хотите переключиться на другой интерфейс или движок, вам не надо постоянно редактировать файл `~/wegorc`. Вы можете предоставить всю информацию *Wego* из терминала, вызвав все соответствующие параметры команды. Запустите `wego -h` для получения списка таких параметров.

Например, следующая команда указывает *Wego* использовать интерфейс JSON с движком `forecast.io`: `wego -f json -b forecast.io -l 28.6139,77.2090 -forecast-api-key=b7ca7b57d8d10702c01b075b1dfe287b`.

Было бы замечательно, если бы будущие выпуски этого отличного инструмента могли расширить его интерфейсы эмодзи и `ascii-art-table`, чтобы предоставить информацию, доступную в выводе JSON. **LXF**

На данный момент *Wego* поддерживает только три движка и интерфейса, но список задач проекта обещает больше в будущих выпусках.

СКОРАЯ ПОМОЩЬ

При внесении изменений в файл настройки `~/wegorc` помните: ожидается, что каждая строка будет в формате `KEY=VALUE`. Так что не ставьте кавычки при указании местоположения, единиц, движка и т.д.

» СЛИШКОМ МНОГО РАБОТЫ!

Для уважающих себя ниндзя *Bash* нетипично жаловаться на задачу или инструмент, требующий слишком много работы или настройки, но если *Wego* вас отталкивает множеством настраиваемых параметров, у нас есть альтернатива.

Служба `wtr.in` требует наличия Python и использует *Wego* и API WorldWeatherOnline для получения данных о погоде для вашего местоположения. А если вы не хотите проходить процесс настройки, можете получить данные о погоде на своей машине с помощью только утилиты `curl`.

Команда `curl wtr.in/Delhi` получит прогноз погоды и представит его в ASCII-стиле так же, как и *Wego*.

Вы также можете использовать `wtr.in`, чтобы проверить погоду в любом аэропорту, указав трехбуквенный код аэропорта вместо местоположения, или погоду в популярном месте. В последнем случае вы должны поставить перед названием `~`, чтобы `wtr.in` его распознал.

Команда `curl wtr.in/~Taj+Mahal` получит погоду в Тадж-Махале, Агра, штат Уттар Прадеш, Индия, тогда как `curl wtr.in/~Stonehenge` получит погоду в Стоунхендже, Авеню, Эймсбери, Ларкхилл, графство Уилтшир, Юго-Западная Англия, Великобритания.

Не забывайте употреблять `+` вместо пробела, если название места состоит из двух или более слов.

ДЕРЕВЬЯ ФАЙЛОВ

Освоим навигацию по файлам в Linux

Запутались в структуре? **Ник Пирс** разоблачит деятельность файловой системы Linux плюс лучший способ ее просмотра: встроенный менеджер файлов Ubuntu.



НАШ
ЭКСПЕРТ

Ник Пирс — журналист-фрилансер, специализирующийся в технологических вопросах и в *A-ha!*.

СКОРАЯ ПОМОЩЬ

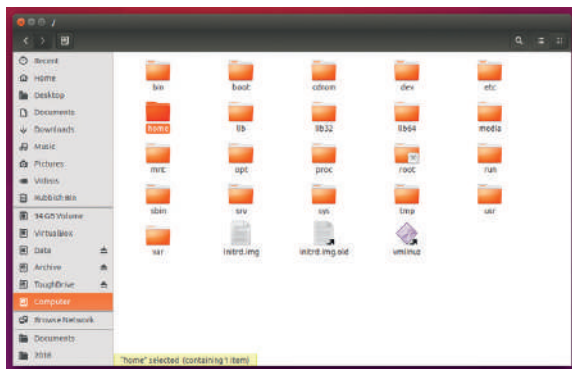
Если вам не нравится то, как *Nautilus* использует хлебные крошки для отображения пути текущей папки, выберите Go > Enter Location... [Перейти > Введите местоположение], чтобы заметить их на более традиционный вид и вручную ввести путь для перехода куда надо.

Без сомнения, вам известно, как Linux определяет подключенные к вашему ПК диски, а также то, что Linux считает всё файлом — в том числе папки (которые являются файлом, отображающим файлы, составляющие содержимое папки). Но как эти файлы организованы на вашем жестком диске, и как получить к ним доступ через графический менеджер файлов Ubuntu, *Nautilus*?

Давайте начнем с основ. Linux организует файлы в древовидную структуру, которая немного похожа на Windows. Основное различие в том, что если Windows физически разделяет диски на отдельные деревья с буквой диска наверху, Linux объединяет всё в одно дерево файлов с корневой [root] директорией (/) на самом верхнем уровне, а всё остальное — включая диски — размещается относительно этой директории.

Это может показаться непонятным, но в реальности Linux предоставляет логичное, организованное отображение всего, что имеется в вашем распоряжении. Поняв, где всё хранится, вы будете относительно легко снова всё находить. А по сути, в повседневном использовании вы ограничиваетесь своей персональной папкой пользователя, которая всегда находится внутри директории */home*, и именно здесь хранятся ваши персональные документы, фото, настройки и прочие данные.

Еще больше всё упрощает тот факт, что Ubuntu поставляется с дружелюбным к пользователю менеджером файлов *Nautilus*, который доступен через ярлык Files в программе запуска. *Nautilus* сразу же покажется знакомым всем, кто использовал собственный менеджер файлов Windows, *Windows Explorer*, и у него множество похожих характеристик.



Перейти в директорию верхнего уровня файловой системы Ubuntu очень просто.

Файлы и папки представлены значками, и открываются двойным щелчком. Вы можете щелкнуть правой кнопкой, чтобы получить доступ к большему числу опций, а удалить файлы можно в Корзину или удалить мгновенно (выделите файл, удерживайте Shift и нажмите Delete) точно так же. Врезка внизу справа стр. 57 рассказывает об основах использования *Nautilus* для доступа и поиска файлов в файловой системе Linux.

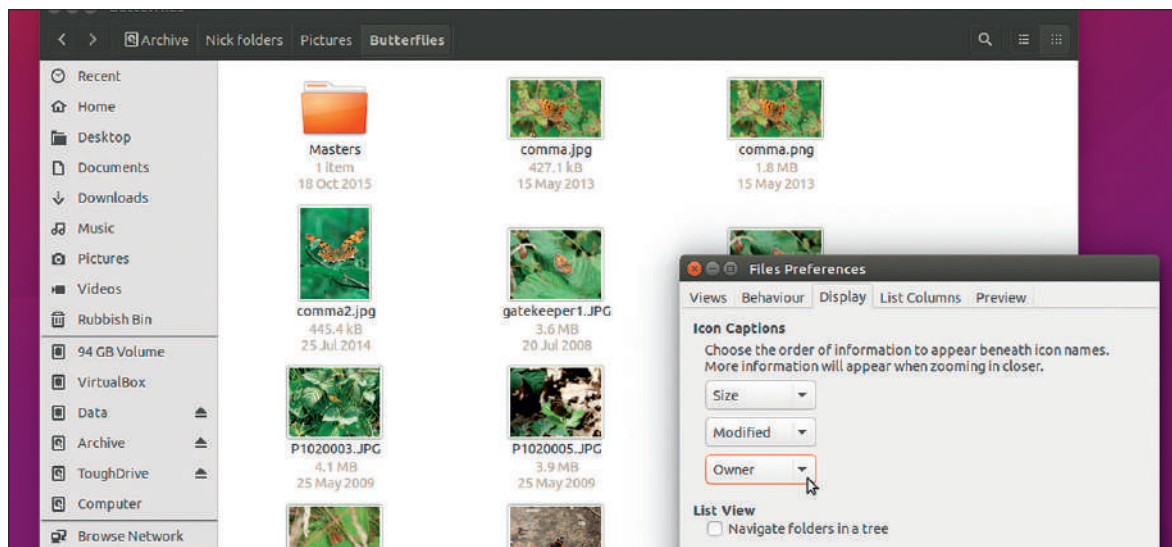
Что лежит в основе

Давайте воспользуемся *Nautilus* для определения основных папок в вашей системе и выясним, для чего они. Запустите его с помощью ярлыка Files в программе запуска, и по умолчанию он откроется в вашей персональной домашней папке — */home/<username>* — в дереве файлов Linux. Здесь вы найдете папки знакомого вида — отражение папок, имеющихся в других операционных системах; все они вполне понятны, за исключением, возможно, **Public** — папки, внутри которой вы можете хранить файлы, доступные для других, когда они входят в систему из своих профилей пользователя — и **Examples**. Это просто ярлык папки, содержащей бесплатные обои, музыку и видео от Ubuntu Free Culture Showcase.

Далее нажмите на ярлык Computer в левой панели навигации, и вы окажетесь в корневой директории, где увидите немалое количество папок. Некоторые можно и не объяснять, но вот некоторые, которые объяснить нужно: папка */bin*, где установлены обычные программы, доступные для употребления всеми пользователями (включая суперпользователя-root и саму систему) — сюда входят все программы Терминала. Файлы запуска — включая ядро Linux — хранятся в */boot*, и мы уже видели, что */dev* содержит ссылки на оборудование вашей системы, включая все подключенные устройства хранения.

Папка */etc* содержит множество файлов настройки, а разные папки */lib* содержат данные настройки программ — подобно тому, что хранится в Реестре Windows. Внешние диски монтируются внутри папки */media*, а */mnt* служит «временной» точкой монтирования для таких устройств, как сетевые папки.

/root — домашняя директория для суперпользователя-root. Она хранится вне основной папки */home* на случай проблем с загрузкой, и доступ к ней можно получить только через команду *sudo*. Загляните в папку */usr*, и вы найдете там папки *bin* и *lib* — именно здесь хранятся программы, установленные вами под вашим собственным профилем пользователя. И, наконец, папка */var* содержит изменяемые данные, включающие потенциально полезные пункты, такие как журналы, для решения проблем.



Настройте то, что будет отображаться под файлами при просмотре в виде значков – увеличьте масштаб, чтобы увидеть подробности.

СКОРАЯ ПОМОЩЬ

Не удивляйтесь, обнаружив множество скрытых файлов – даже внутри вашей домашней директории – после того, как выберете View > Show Hidden Files. Это нормально, и лучшее, что можно сделать – снова выбрать эту опцию, чтобы скрыть их.

Всё это интересно и полезно знать, но вряд ли нужно вам для повседневного использования.

Исследуем Nautilus

Вернитесь в свою домашнюю директорию. Теперь переведите мышь вверх, в строку меню, где вы увидите собственный выбор меню *Nautilus*. Посмотрите на них — File [Файл], Edit [Редактировать], View [Вид], Go [Перейти], Bookmarks [Закладки] и Help [Помощь] — и вы увидите список знакомых команд; однако некоторые из них выделяются.

Начните со вкладки File > New Tab [Файл > Новая Вкладка], которая позволяет открыть несколько местоположений, не загромождая ваш рабочий стол окнами. Вы увидите, что вкладки появляются над панелью папки, и вы можете перемещать (или копировать, удерживая Ctrl при перетаскивании) файлы между ними, выбрав их в одной вкладке, перетаскивая через другую вкладку и затем ожидая, пока эта вкладка откроется, чтобы уронить их туда.

Еще один удобный инструмент можно найти, выбрав Edit > Select Items Matching... [Редактировать > Выбрать соответствие...]. Отсюда вы можете применять фильтры, которые обеспечат выбор только определенных файлов, соответствующих этим фильтрам (например, типу файла — PNG, JPG и т.д.). Пока вы находитесь в меню Edit, выберите Preferences [Предпочтения] для настройки работы *Nautilus* — особенно интересна вкладка Display [Показать], где вы добавляете под значком дополнительные описания (например, размер файла или владельца).

Здесь вы также увидите ссылку на масштабирование — по умолчанию можно отображать до трех подписей под значками в зависимости от степени увеличения значков. Вы увидите эти кнопки управления в меню View, или можете использовать Ctrl++ и Ctrl+- и Ctrl+0 для быстрой установки уровня масштаба.

Скрытые файлы

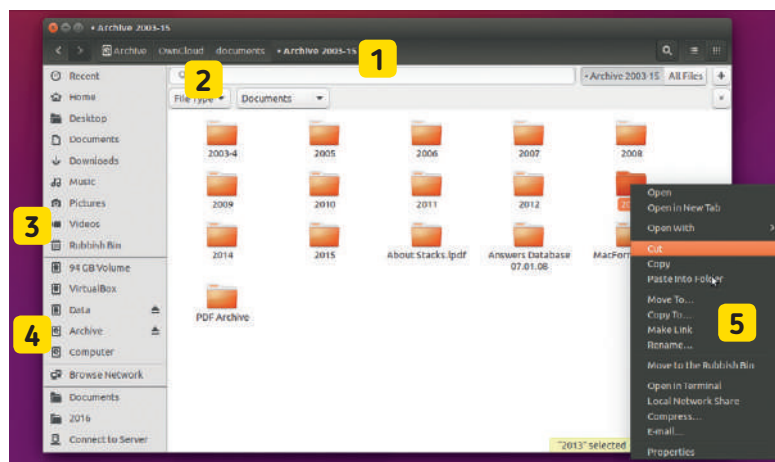
По умолчанию Ubuntu скрывает все файлы, перед которыми стоит точка (.) — чтобы увидеть, как это работает, щелкните правой кнопкой по файлу в *Nautilus*, выберите Rename [Переименовать], припишите точку, затем нажмите Ctrl+R, чтобы обновить вид... и, о ужас, файл исчезнет. Не паникуйте — выберите View > Show Hidden Files [Вид > Показывать скрытые файлы], и он магическим образом появится, позволяя вам снова переименовать его, чтобы удалить метку точки и оставить его видимым постоянно.

Отыскиваете определенный файл? Нажмите на кнопку с увеличительным стеклом, и появится панель поиска. Просто введите параметры поиска, и появится список подходящих файлов.

Желая расширить свой поиск на всю файловую систему, нажмите на кнопку All Files [Все Файлы].

Вы также можете применить к своему поиску фильтры типа файлов, чтобы ограничить результаты документами, фото или по иным критериям. Нажмите на кнопку +, оставьте выбранным File Type [Тип Файла], и затем либо выберите тип из выпадающего меню (например, Illustration [Иллюстрация], Spreadsheet [Таблица] или Pdf/Postscript, или укажите в фильтре расширение файла (например, .png) вручную. *Nautilus* поддерживает разные фильтры, и вы сможете, например, легко искать фото .png, .jpg и .tif, не включая .gif или иные типы изображений. **LXF**

Исследуем интерфейс Nautilus



1 Хлебные крошки

По мере углубления в дерево папки имеет смысл использовать эти значки папки «хлебные крошки», чтобы увидеть, где вы находитесь. Нажмите на нее, чтобы перепрыгнуть в эту папку.

2 Поиск

Нажмите на значок с увеличительным стеклом, чтобы открыть мощный инструмент поиска – нажмите на кнопку + для отображения фильтра по типу файлов, чтобы уточнить поиск.

3 Панель навигации

Она предоставляет удобные ярлыки для основных частей вашей

системы – верхняя часть вполне понятна, и включает удобный список недавно просмотренных файлов.

4 Прикрепленные диски

Все распознанные диски – внутренние и внешние – перечислены здесь. Ярлык Computer предоставляет вам ссылку на корневую папку вашей файловой системы Linux.

5 Щелчок правой кнопкой

Щелкните правой кнопкой по файлу или папке, чтобы увидеть больше опций для взаимодействия с ней. Особенно полезны функций 'move to [перейти в]' и 'copy to [скопировать в]'.

HANDBRAKE

Кодируем видео быстрее и лучше

Пытаясь создать идеальную нарезку видео, **Джон Найт** скрежещет зубами и укрощает дикого зверя — *HandBrake*. Держись, ковбой!



НАШ
ЭКСПЕРТ

Джон Найт

Если Джон не играет в компьютерные игры на французском, то громко колотит по большому барабану.

Как сказано на сайте *HandBrake*, «это инструмент для конвертирования видео практически из любого формата в ряд современных широко поддерживаемых кодеков». И хотя *HandBrake* определенно стоит изучить, мы должны признать, что понять его будет непросто, если вы — новичок: программа явно создана для опытных пользователей. Тем не менее, по некоторым сценариям обучения мы надеемся превратить его в нечто более удобоваримое, и интерфейс уже не будет так пугать.

Да, интерфейс *HandBrake* немного непривычный и сначала может запутать, но не отступайте: некоторые части работают прекрасно и реально ставят *HandBrake* на особицу. У нас нет места для рассказа обо всей программе целиком, но мы, надеемся, сможем провести вас по достаточной части интерфейса, чтобы вы могли начать кодировать собственное видео с удобством и уверенностью.

Итак — вперед, в GUI

С целью превратить GUI в нечто более удобоваримое, мы пройдем через несколько простых сценариев, чтобы его понять. Начнем мы с однофайлового сценария низкого уровня сложности, и затем перейдем к нескольким файлам в очереди, после чего добавим более расширенные опции для получения действительно высококачественных рипов. Начнем с превращения большого в малое.

H.265 предназначен стать новым стандартом онлайн-видео, при значительном уменьшении размера файла по сравнению с его предшественником, H.264. И сжатие видео H.264 до куда меньшего по размеру видео H.265 выглядит разумным. Поэтому, пожалуйста, подготовьте большой файл H.264 (обычно они называются *.mkv*), и мы сделаем его мини-вариант.

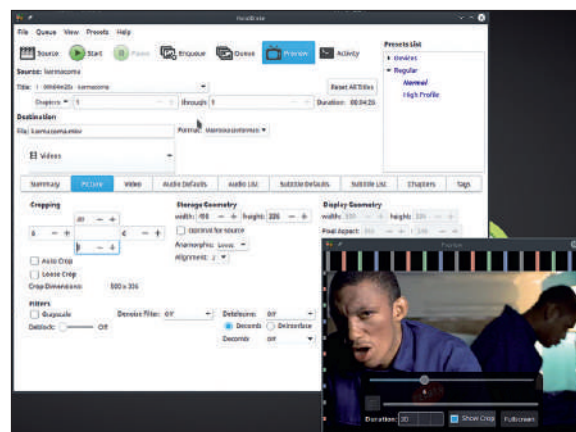
Если ваш компьютер достаточно мощный, чтобы обработать H.265, следуйте дальше; в ином случае попробуйте заменить эти кодеки вашим собственным примером того, как более старый источник может быть сжат более новым кодеком.

Кодирование одного файла — процесс довольно простой:

- 1 Нажмите на кнопку **Source** и выберите ваш файл или диск.
- 2 Выберите опции форматирования и результирующего кодека.
- 3 Нажмите **Start**.

Однако GUI, кодеки и форматирование файлов могут потребовать некоторого разъяснения.

Начните с нажатия большой кнопки **Source** [Источник] вверху слева и выберите свой большой исходный файл H.264. Далее



Окно Preview HandBrake — это просто спаситель жизни, и оно появляется из пакета обработки видео. Управление монтажом показалось нам просто блестящим.

выберите место хранения кодированного видео. По умолчанию *HandBrake* сохраняет все видео в папке **Videos** вашей домашней директории. Если вы предпочитаете другую папку назначения, можете изменить это в выпадающем окне под заголовком **Destination** [Назначение].

Теперь вам надо выбрать **Format** [Формат], который тоже находится под заголовком **Destination**. Вы можете выбирать между MPEG-4 или Matroska; если ваше видео для соцсетей, выберите MPEG-4, а для полнометражного фильма лучше Matroska. Разные программы и разные аудитории будут рассчитывать на разные форматы, так что выбирайте наиболее подходящий для конкретной задачи.

Чтобы выбрать видеокодек, нажмите на вкладку **Video** внизу. В меню **Video Encoder** выберите кодек H.265. Если H.265 в этом меню отсутствует, вероятно, он не установлен в вашей системе. Тогда вам, возможно, придется заглянуть в свой менеджер пакетов и установить соответствующие пакеты.

Далее мы не будем менять никаких опций — оставим всё по умолчанию и изучим вывод. Чтобы приступить к нарезке, просто нажмите на большую зеленую кнопку **Start**. Внизу появится расчетное время — потерпите, ожидание может оказаться долгим!

По завершении риппинга посмотрите на получившийся файл, который должен быть в папке **Videos**.

СКОРАЯ ПОМОЩЬ

Кодирование видео сильно нагружает ваш CPU, который мгновенно нагреется, причем, возможно, до предела. Чтобы избежать перегрева, закройте все ненужные программы (в частности, браузер); а если у вас хватает умения — при выключенном компьютере очистите радиатор от пыли и мусора.

При настройках по умолчанию мы сжали видео H.264 объемом 16,6 ГБ до видео объемом 2,3 ГБ с отличным качеством. Особо радоваться не стоит — снижение качества картинки всё равно неизбежно (хотя оно и отличное), и аудио микшировано из DTS в стерео. Так или иначе, результаты весьма приятные для обычного просмотра; но к этим проблемам мы еще вернемся в разделе для продвинутых.

Мы сделали файл компактнее, но как насчет обратного процесса? Не считайте за глупость: H.265 очень сильно нагружает процессор — для старых компьютеров это чересчур. Наш компьютер достаточно мощный, но ему уже несколько лет, и когда мы попытались воспроизвести техническое демо LG 4K в H.265, компьютер чуть не треснул.

Используя более старый кодек, вы гарантируете большую совместимость со старыми компьютерами. Итак, если мы конвертируем видео 4K в H.264, воспроизведется ли оно?

Следуя тем же шагам, что и ранее, мы выбрали в качестве исходного техническое демо LG и в качестве формата — Matroska, но на сей раз во вкладке Video использовали H.264 — настройку системы по умолчанию. И как оно работало? Превосходно.

H.265, возможно, не слишком хорошо работал в 4K, но 4K — уже большая нагрузка на любую систему. Как насчет H.265 в более реалистичном разрешении, например, 1080p?

Уже загрузив 4K демо, мы вернулись на вкладку Video и просто изменили Video Encoder [Видеокодировщик] на H.265; но чтобы изменить разрешение, вам нужно открыть вкладку Picture. Кнопки управления под заголовком Storage Geometry [Геометрия Хранилища] установят ширину и высоту.

Когда вы слышите о 1080p, 720p или 480p, имеется в виду высота. Так что для увеличения или уменьшения масштаба просто установите нужный показатель управления высотой, а ширина задастся в соответствии с этим. Для справки, в формате 16:9, 1080p — это 1920×1080, 720p — 1280×720, а 480p — 854×480.

И как же компьютер работал в 1080p вместо 4K? Отлично, без всяких проблем. Отсюда возникает вопрос: а нужно ли потребителям 4K? Или это такой маркетинговый ход от производителей телевизоров?

Много файлов, легкая работа

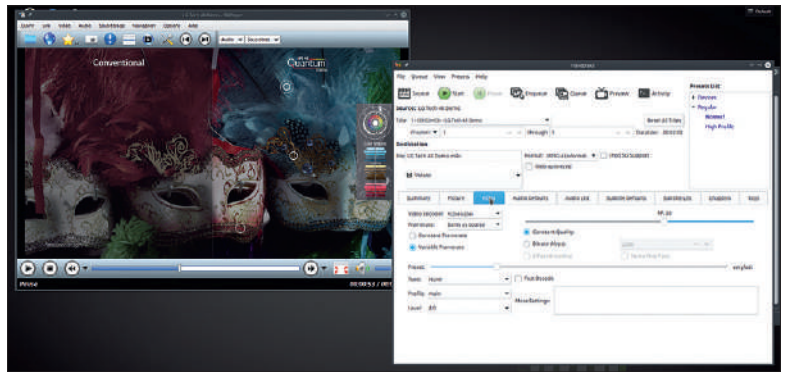
И вот где HandBrake выступает во всей красе. Функция Queue позволяет выполнить целый список задач вместо одной, что идеально для запуска группового копирования, пока вы можете отойти от компьютера — особенно ночью.

Когда вы разберетесь с тем, какая кнопка за что отвечает, процесс для нескольких файлов станет довольно простым делом:

- 1 Нажмите на кнопку Source и выберите файл или диск.
- 2 Выберите опции форматирования/вывода.
- 3 Нажмите Enqueue [Поставить в очередь], что поместит задачу кодирования в список.
- 4 Повторите со следующим файлом, затем нажмите Enqueue.
- 5 Нажмите Queue [Очередь]; чтобы просмотреть имеющиеся задачи; нажмите на кнопку Edit [Редактировать] (значок с карандашом), если хотите внести какие-то изменения.
- 6 Нажмите Start.

Если вы обнаружите, что все ваши настройки исчезли, а вам надо внести изменения, GUI по-прежнему находится в режиме Queue — просто еще раз нажмите на кнопку. Нам особенно нравится, что у каждого видео в Queue имеются собственные настройки, и изменения, примененные к одному видео, не затрагивают остальных.

Если в силу каких-либо причин надо что-то изменить, нам особенно понравились четыре опции, которые вы получаете по нажатию на большую красную кнопку Stop: Cancel Current and Stop [Отменить текущее и остановить]; Cancel Current, Start Next [Отменить текущее, Запустить следующее]; Finish Current,



Техническое демо компании LG 4K было слишком сложным для нашей не новой машины, чтобы запустить H.265, но переход на более старый кодек обеспечил отличную работу.

then Stop [Завершить текущее, затем остановить]; и Continue Encoding [Продолжить кодирование]. Может, это и мелкий штрих, но сделан заботливо, и может сэкономить вам часы впустую потраченной работы!

Настройка качества

Именно это отличает новичков от ветеранов — когда вы вникаете в детали кодирования видео. Хотя наши предыдущие рипы были значительно меньше исходников, они всё равно достаточно велики, как минимум несколько гигабайт каждый. А если вы хотите сделать рип еще меньше?

Первое, на что надо обратить внимание — это битрейт. Откройте вкладку Video, и вы увидите ползунок, окно-флажок для Constant Quality [Постоянное Качество], и еще одно для Bitrate (kbps). Опция Constant Quality использует ползунок: чем дальше он влево, тем выше качество и размер файла, а чем дальше вправо — тем меньше файл и ниже качество. Это простой способ.

Если вы предпочитаете более прямой контроль, используйте опцию Bitrate (kbps). По умолчанию она установлена на 2,500, но мы думаем попытать счастья и установить ее на 1,000 kbps. В сочетании с низким битрейтом, более низким разрешением

СКОРАЯ ПОМОЩЬ

А нужно ли вам более высокое разрешение видео? Больше 720p нередко вам ни к чему, а для скучной офисной работы приемлемо даже 480p.

» О КОДЕКАХ В ДВУХ СЛОВАХ

Видеокодеки — это форматы, которые сжимают исходное видео в нечто меньшее и более удобное для обработки. Хотя существовала масса их видов, реально всё началось с MPEG (Moving Picture Experts Group) в 1988 г. Увеличение мощности обработки позволило усложнить алгоритмы сжатия видео, дав возможность втиснуть видео лучшего качества в меньший по объему файл. Большим шагом вперед по сравнению с MPEG-1 стал DVD MPEG-2, и в начале 2000-х делиться своими видео стали во множестве, ведь DivX/MPEG-4 позволили уместить видео, по качеству близкое к DVD, на один CD.

Однако каждый важный шаг в сжатии видео требует соответствующего улучшения оборудования. Вначале для MPEG-4 требовался достаточно мощный компьютер, но в конечном итоге аппаратное декодирование MPEG-4 вошло в обычные потребительские устройства, такие как DVD-плееры и даже автомобильные магнитолы. Опять же, у оборудования ушло некоторое время на то, чтобы догнать кодек H.264, который в основном и используется в наши дни, однако даже такое скромное устройство, как Raspberry Pi, может воспроизводить H.264 видео, не растекаясь малиновой лужицей.

Следующий важный пункт — H.265, ему нужно примерно в 3–10 раз больше производительной силы, чем для H.264, но преимуществом является файл на выходе объемом примерно в половину меньше. Как только потребительское оборудование нагонит этот новый стандарт, H.265 обещает обеспечить настоящее видео с высоким разрешением даже через скромное интернет-соединение.

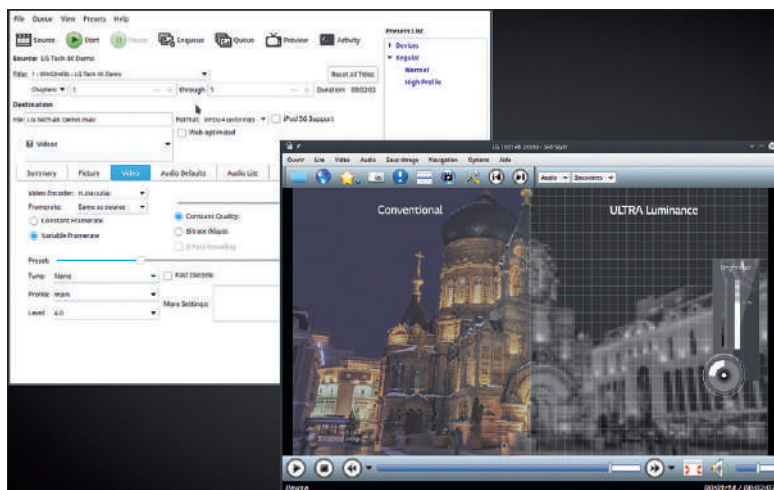


720p и стереозвуком у нас получился файл, который мы сжали с 16,6 ГБ всего до 1 ГБ!

Естественно, по качеству видео и рядом не стояло с оригиналом, и ухудшение качества из-за сжатия становится заметным, но для случайного зрителя качество всё же приемлемое. И на этом не обязательно останавливаться. По мере снижения битрейта появляются уродливые визуальные артефакты, но у *Hand-Brake* припасено пара трюков для борьбы с ними.

Во вкладке **Video**, если вы используете ручную опцию **Bitrate** (kbps), вы можете включить **2-Pass Encoding** [кодирование в два прохода]. Оно рипует видео дважды и использует данные из первого рипа для улучшения точности картинки во втором рипе. Однако это будет в два раза дольше!

Чтобы еще улучшить изображение, откройте вкладку **Picture**. Внизу находится раздел **Filters**. Начните с **Deblock**, устраняющего тот ужасный эффект, когда изображения низкого качества превращаются в кирпичи. **Denoise** устраняет проявления шума, такие как зернистость, усложняющая рип, и может уменьшить объем файла. **Detelicine** устраняет неприятный эффект гребенки, возникающий вследствие процесса конвертирования пленки в видео.



Снижение разрешения 4K до 1080p позволило видео отлично работать. Вы и вправду видите разницу?

» ПОГОДИТЕ, ЕСТЬ ВЕЩИ ПОКРУЧЕ!

Рип DVD *HandBrake* не просто делает рип из отдельных видеофайлов, он также поддерживает DVD. Вы можете найти свой диск, если посмотрите из окна **Source**; или выберите диск в меню **File**. Конечно, мы в *Linux Format* уважаем копирайт, и предполагаем, что у вас есть права на всё вами рипуемое!

Отметки глав На вкладке **Chapters** [Главы] вы можете добавлять, настраивать и даже отмечать главы, что придаст вашему файлу весьма профессиональный вид.

Градации серого [Grayscale] Причем в классическом английском это 'greyscale'. Отметьте эту опцию, и ваше видео конвертируется в черно-белое — идеально, чтобы добавить настроения в ваши фильмы или помочь молодым кинодеятелям быть претенциозными!

Преднастройки устройств Цитируем список функций на сайте: «Начните работать с *HandBrake* сразу же, выбрав профиль, оптимизированный для вашего устройства, или выберите универсальный профиль для стандартных или высококачественных конверсий. Легко, просто, быстро. Для тех, кому нужно больше выбора, перепробуйте разные базовых и расширенных опций, чтобы улучшить ваши кодирования». Мы не знали, что «кодирование» здесь должно стоять во множественном числе, ну да ладно. Похоже, что у *HandBrake* есть предустановки для разных устройств, таких как iPhone, планшеты Android, PlayStation 3 и даже AppleTV!

И, наконец, у нас есть **Decomb** и **Deinterlace**. Если вы когда-либо смотрели что-нибудь снятое на видео и выглядящее «морщинистое», это называется перемежение [interlacing]. **Deinterlacing** исправляет его, но это настолько общая проблема, что методов деинтерлейсинга изобретено множество. Это означает, что вам просто надо поэкспериментировать с каждой настройкой — уж извините. **Deinterlace** тупо применяет эффект ко всему видео целиком, тогда как **Decomb** пытается сначала обнаружить эффект перемежения и избежать ненужного снижения качества.

У включения всех этих опций есть один большой недостаток. Если базовый рип можно сделать в реальном времени или даже быстрее, продвинутый рип займет больше времени. Причем намного.

На тестовый рип двухчасового фильма со всеми включенными функциями ушло около 16 часов. Рип в течение двух часов — это одно, и эй — даже рип на всю ночь еще ничего... но 16 часов? Это выводит на новый уровень непрактичности, и вам придется задуматься, так ли вы любите качественный рип!

Каков аспект?

Хотя на первый взгляд кнопки управления картинкой неинтуитивны, они превосходны, особенно в сочетании с окном **Preview**. Если у вас есть картинка, которую надо обрезать или изменить пропорции согласно правильному отношению сторон [aspect], отлично. А если у вас есть один из этих раздражающих файлов, которые нужно исправить в обеих областях, это еще лучше! Откройте вкладку **Picture**, мы начнем с нее.

Помимо устранения черной каймы, обрезка удобна тем, что рендеринг этих черных областей съедает немалую часть объема файла. Если вы собираетесь делать обрезку, с этого и начните, поскольку **Cropping controls** [Управление обрезкой] к вашей вящей злости переустановит настройки **Display Geometry**.

HandBrake по умолчанию включает функцию **Auto Crop**, и она в большинстве случаев обнаружит и уберет черную кайму автоматически. Интересно то, что она обрежет неиспользуемые границы у всех видеофайлов, а не только у тех, где имеется разница в пропорции кадра. Тем не менее, всегда проверяйте окно **Preview**, чтобы убедиться в его нормальной работе. Некоторые файлы настолько плохо рипованы, что управится с ними только человеческий газ!

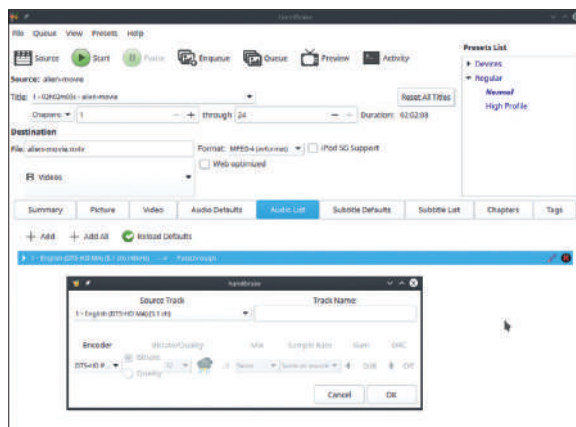
К счастью, настроить обрезку можно на ходу, и кнопка **Show Crop** очень удобна. Прежде чем продолжить, поведите ползунком предпросмотра с целью убедиться, что с остальной частью файла всё в порядке.

При желании можете делать обрезку вручную. Сначала отключите **Auto Crop** — включится управление **Cropping** — и откройте окно **Preview**. По мере регулировки размеров обрезки, окно **Preview** будет обновляться в реальном времени.

Переходим к изменению размеров и отношению сторон: **GUI** здесь непонятный и требует поразмыслить. Заглянув в **Storage Geometry**, вы увидите, что всё здесь связано с выпадающим меню **Anamorphic**: оставьте его как **Loose**, если хотите изменить размер видео при прежнем отношении сторон, или переведите в **Custom**, если хотите изменить и то, и другое.

При установленном **Loose** вы можете задавать ширину или длину, но и та, и другая сохраняет те же самые числовые пропорции. С **Custom** вы можете задать ширину и высоту, и окно **Preview** увеличится или уменьшится вместе с ними.

Желая узнать, какое соотношение сторон у видео сейчас, уберите отметку с **Keep Aspect** в соседнем поле **Display Geometry** [Геометрия Дисплея] — и показатель **Display Aspect** [Пропорции дисплея] его продемонстрирует. Продолжайте изменять размер, пока он не станет 16:9 (или иным нужным вам значением). Это управление геометрией очень мощное, но в использовании бывает кошмарным.



Невозможно уйти от того факта, что аудиодialogи *HandBrake* кошмарны. Однако сразу переходите во вкладку *Audio List*, и у вас появится шанс!

Слушайте, черви!

Изначально мы здесь сосредоточились на видео, однако звук является половиной формулы, и может значительно повлиять на объем файла. Хотя в *HandBrake* можно делать весьма крутые вещи, настройка аудио — это место, где GUI его подводит...

Для вывода стерео всё в основном хорошо. В *Audio Defaults* [Аудио по Умолчанию] или во вкладке *Audio List* [Список Аудио], вы обязательно найдете настройки для выбранного кодека, хотя вкладка *Audio List* определенно проще для понимания.

А вот если вы хотите получить объемный звук, работы потребует больше. Здесь мы расскажем о DTS, DTS-HD и Dolby Surround. Учитывая, что DTS/DTS-HD является основным выбором для объемного звука в Blu-ray, сначала мы расскажем о нем. Однако Dolby Surround верховодит в DVD, так что потом мы расскажем и о нем.

Начав с DTS и его родича, HD, мы обнаружили, что лучше полностью избежать вкладки *Audio Defaults*, открыв вместо нее вкладку *Audio List*. В списке ниже вы, возможно, найдете листинг вывода по умолчанию, отображающий нечто вроде "English (DTS) (5.1 ch) (48kHz) > AC3 (Dolby Pro Logic II) (48kHz)".

Нажмите на кнопку *Edit* [Редактировать] со значком карандаша справа. Сплывет новое окно настройки, практически с теми же функциями управления, что и во вкладке *Audio Defaults*, но урезанное до такой степени, что в этом можно разобраться.

Выберите свой аудиотрек из списка *Source Track* [Исходный Трек]. Теперь под заголовком *Encoder* выберите *DTS-Passthrough* (или *DTS-HD Passthrough*, если это есть в вашем файле), в этот момент остальные опции станут серыми. Нажмите *OK*, и аудио должно настроиться должным образом на объемный звук, с указанием в списке, что оно конвертируется из DTS в "Passthrough". Есть и другие способы заставить это сработать, но мы сочли, что этот — самый простой, и позволит избежать потери качества.

Микшируем

Если ваш усилитель не поддерживает DTS, но поддерживает Dolby Surround, мы обнаружили, что можно сделать подходящий ремикс сигнала. Выберите *AC3* из списка *Encoder*, затем — *5.1 Channels* из списка *Mix*. В нашем усилителе это определилось как *Dolby Surround* — и сработало. Ирония в том, что когда мы выбрали *Dolby Surround* из списка *Mix*, он выводился только как стерео! Черт.

Теперь перейдем к сохранению исходника *Dolby Surround*. Если у вас объемный звук в исходном аудио, в списке ниже вы,

вероятно, найдете опцию по умолчанию, нечто вроде: "English (AC3) (5.1 ch) (48kHz) > AAC (avcodec) (Dolby Pro Logic II) (48kHz)".

Нажмите на значок *Edit*, как раньше, и снова выберите свой *Source Track*. На сей раз в меню *Encoder* выберите *AC3 Passthrough*; в этот момент остальные функции опять посереют.

Этот способ — лучший и простейший, если вы хотите оставить свой объемный звук нетронутым. Но если вы хотите изменить такие опции, как битрейт или усиление, можете сохранить *Dolby Surround*, выбрав из *Mix* вместо этого *AC3 с 5.1 Channels*. И снова, выбрав *Dolby Surround*, вы, вероятно, получите стерео. Постарайтесь сдержаться и не ломать табуретки.

Хотя использование *passthrough* для аудио проще и лучше по качеству, помните, что оно солидно увеличит объем файла. Наш взятый для примера рип был всего 2,3 Гб в стерео, а с объемным звуком стал 6,6 Гб — сильное увеличение объема, однако всё равно на 10 Гб меньше исходного!

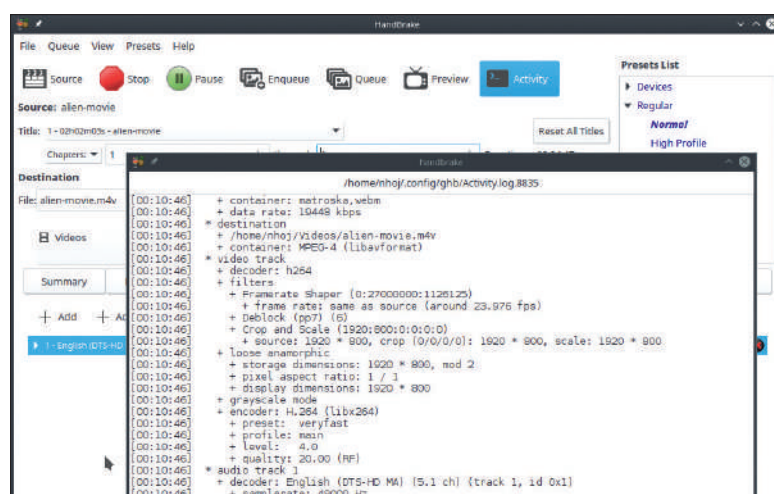
Если вы упорно хотите сохранить объемный звук в вашем рипе, при этом сохраняя небольшой объем файла, рекомендуем поэкспериментировать с кодеками и битрейтом, пока вы не найдете подходящего решения.

Взвешиваем

В конечном итоге, для рипа главное — баланс и компромисс: размер против качества, сжатие против совместимости, тщательность против практичности. На чем он может работать? Он прекрасно сжат — не работает только на новом оборудовании, или вы хотите пожертвовать некой долей совершенства — чтобы его мог воспроизводить кто угодно? Всё сводится к решению, без чего вы не можете прожить, потому что всегда приходится идти на уступки.

Более того, есть практические доводы, о которых стоит задуматься. Ждут ли от вас, что вы будете выкладывать новые видео быстро и регулярно, или вы можете не спеша создавать нечто высококачественное и с небольшим объемом файла? И если вы планируете регулярно делать рипы видео, не приобрести ли вам еще один компьютер специально для этой задачи?

Все эти факторы определяют, какой вы риппер, и *HandBrake* предназначен для более профильных пользователей. Хотя кое-какие его функции управления GUI бесят, есть и другие, действительно блестящие, которые обеспечивают функции, делающие программу незаменимой для любого серьезного кодировщика видео. **LXF**



Окно *Activity* выглядит сущей абракадаброй, но когда что-то неизбежно пойдет не так, именно здесь вы найдете решение проблем.

СКОРАЯ ПОМОЩЬ

Используйте более старый кодек, если хотите получить совместимость с большим диапазоном устройств. Если вы хотели бы, чтобы ваше видео работало на таких устройствах, как автомобильные системы или DVD-плееры, вашим выбором должен стать *MPEG-4*. Более подробные рекомендации по оптимизации устройств вы найдете на сайте *HandBrake* на www.handbrake.fr.

BLUETOOTH

Заставим Bluetooth работать с Ubuntu

Ник Пирс открывает, как добиться в Linux распознавания и работы вашего оборудования Bluetooth. Послушаем, чтобы заполучить беспроводное соединение!



НАШ
ЭКСПЕРТ

Ник Пирс имеет десятилетний опыт любительской работы с Linux вообще и с Ubuntu в частности.

СКОРАЯ ПОМОЩЬ

Подключили общий адаптер Bluetooth и хотите узнать его возможности? Просто введите `btmgmt` в окно терминала, чтобы всё выяснить. Обратите внимание, что номер версии соответствует его версии HCI.

Проблемы с тем, чтобы заставить работать адаптер? Используйте аппаратные инструменты Linux, чтобы проверить, распозналось ли ваше устройство.

Когда речь идет о беспроводном соединении с вашим ПК, Bluetooth очень легко упустить из вида. Однако низкое энергопотребление делает его идеальным для всех беспроводных сценариев, и не в последнюю очередь — подключения беспроводных устройств ввода к вашему ПК. Зачем перегружать порты USB беспроводными адаптерами для клавиатур и мышей, если можно подключить их через Bluetooth?

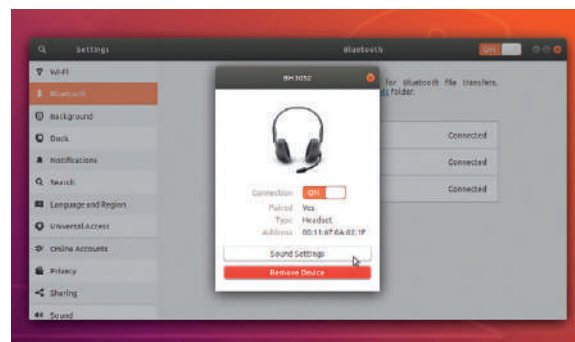
Это не означает, что Bluetooth идеален — вовсе нет, особенно когда речь идет о Linux. Реализация Bluetooth улучшается и становится проще в управлении, но она по-прежнему готова рухнуть при малейшем намеке, в частности, при попытках выполнить более сложные задачи, вроде потокового воспроизведения аудио или отправки файлов с и на мобильные устройства. Мы не только покажем вам, как заставить Bluetooth работать, но и откроем некоторые секреты в помощь решению проблем, если они возникнут.

Bluetooth встроен во многие современные ПК и ноутбуки, но вы можете добавить функцию Bluetooth всего за £2 с адаптером USB. Ищите адаптер, который поддерживает Bluetooth 4.0 или 4.1. Даже если он не поддерживает Linux напрямую, большинство всё равно должно работать, но попытайтесь найти тот, который успешно используется другими — введите в Google название и свой дистрибутив или 'Linux', и посмотрите, что получится.

По заказу начинающих

В наши дни большинство адаптеров Bluetooth — plug-and-play. Вероятно, основные пакеты для использования Bluetooth уже установлены, но для безопасности запустите следующие команды в окне терминала:

```
nick@nick-ubuntu:~$ sudo rfkill list
0: hci0: Bluetooth
   Soft blocked: no
   Hard blocked: no
nick@nick-ubuntu:~$ hciconfig -a hci0
hci0:
  Type: Primary
  BD Address: 00:15:03:EE:11:40 ACL MTU: 310:10 SCO MTU: 64:8
  UP RUNNING PSCAN ISCAN
  RX bytes:676148 acl:3019 sco:0 events:2352 errors:0
  TX bytes:232068 acl:1942 sco:0 commands:494 errors:0
  Features: 0xff 0xff 0x8f 0xfe 0xdb 0xff 0x5b 0x87
  Packet type: DM1 DM3 DM5 DM1 DM3 DM5 HV1 HV2 HV3
  Link policy: RSMITCH HOLD SNIFF PARK
  Link mode: SLAVE ACCEPT
  Name: 'nick-ubuntu'
  Class: 0x1c0104
  Service Classes: Rendering, Capturing, Object Transfer
  Device Class: Computer, Desktop workstation
  HCI Version: 4.0 (0x6) Revision: 0x22bb
  LMP Version: 4.0 (0x6) Subversion: 0x22bb
  Manufacturer: Cambridge Silicon Radio (10)
```



После подключения вы найдете удобные ярлыки соответствующей части Settings для настройки ваших устройств Bluetooth.

```
$ sudo apt-get update
```

```
$ sudo apt-get install bluetooth bluez bluez-tools rfkill
```

Затем удостоверьтесь, что ваш адаптер Bluetooth определился:

```
$ sudo rfkill list
```

В списке должен выводиться ваш адаптер Bluetooth, а **SOFT** и **HARD** должны быть установлены в **Unblocked**. Если это не так, скомандуйте

```
$ sudo rfkill unblock Bluetooth
```

Это должно обеспечить корректную работу адаптера Bluetooth. Если адаптер опять не появился, попробуйте

```
$ lsusb
```

Есть ваш адаптер в этом списке? Если да, введите `hciconfig` и нажмите Enter. Эта команда должна вывести устройство как `hci0`; тогда введите следующее, чтобы выяснить производителя:

```
$ hciconfig -a hci0
```

Посетите <https://wiki.ubuntu.com/HardwareSupportComponents-BluetoothUsbAdapters>, чтобы увидеть, есть ли ваше устройство в списке, и выполните инструкции, которые помогут вам ввести его в действие.

Всё еще не помогло? Попробуйте такую команду:

```
$ dmesg | grep Bluetooth
```

Проверьте наличие **Direct firmware load failure**, что будет означать отсутствие прошивки. Вам нужно будет отследить это и скопировать в указанную папку в `/lib/firmware/` — всё это вместе с внутренними Wi-Fi/Bluetooth адаптерами с чипсетами Broadcom должно запустить поиск, например, на <https://github.com/winterheart/broadcom-bt-firmware>.

Если не получится, запишите имя и задайте поиск в Google вместе с 'Ubuntu', чтобы получить больше информации и выяснить, известна ли эта проблема. Мы, например, столкнулись с тем, что более новые чипы могут не распознаваться ядром,

которое в данный момент работает. И если обновление Ubuntu не работает, а вы набили руку в Linux, подумайте об обновлении до самой последней версии ядра. Если это кажется вам сложным, попробуйте *Ubuntu Kernel Update Utility*, инструмент GUI:

```
$ sudo apt-add-repository -y ppa:teejee2008/ppa
$ sudo apt-get updateshare files bluetooth linux
$ sudo apt-get install ukuu
```

Запустите это, и вы увидите список всех имеющихся версий. Выберите более новую версию и установите ее — если Linux не сможет загрузиться, вы сможете вернуться к изначальному ядру при загрузке через меню *Grub* (ищите в меню *Advanced Options* [Расширенные опции]), затем попробуйте более старую версию. Если ваш адаптер упрямо продолжает отказываться работать, попробуйте другой — как мы уже заметили ранее, вас это не разорит.

Как только ваш адаптер Bluetooth определится, проверьте состояние сервиса Bluetooth:

```
$ sudo service bluetooth status
```

Если он не работает —

```
$ sudo service bluetooth start
```

Настройка устройств Bluetooth

Теперь можете закрыть окно терминала. Если всё хорошо, вы теперь можете делать всё, что хотите, в рабочем столе Gnome. Пошаговое руководство (см. стр. 65) откроет вам простой путь к соединению, подключению и настройке устройств с помощью встроенного инструмента Settings.

В большинстве случаев это должно работать, но хотя определенные устройства, скорее всего, будут работать отлично — мыши и клавиатуры, например — другие могут стать причиной головной боли, особенно звуковые устройства. Главное — не терять терпения: не спешите нажимать на кнопки. Иной раз вам может понадобиться перезагрузить ПК и попробовать заново подключить их или переподключить.

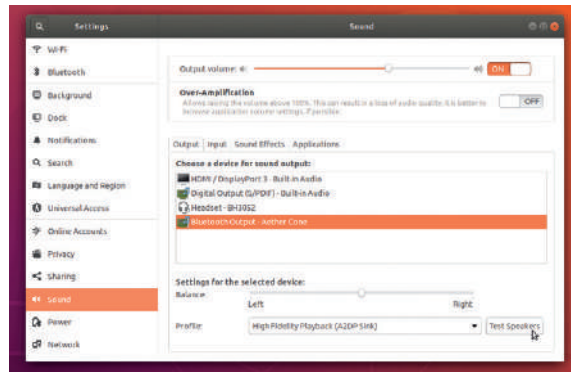
Стоит также отметить окно *Blueteman*, альтернативного инструмента, который обеспечит вам больше контроля за соединением, подключением и процессом управления, а также выдаст более подробные сообщения в случае ошибок. Кроме того, не забудьте поискать ваше устройство в Google вместе с 'Linux' или 'Ubuntu', чтобы выяснить, не прошел ли уже кто-то этим путем.

Аудио на беспроводные устройства

Использование Bluetooth для потокового аудио — популярное применение беспроводной технологии, особенно с учетом быстрого размножения разнообразных дешевых наушников, гарнитур и колонок Bluetooth. И опять же, поищите в Google потенциальную покупку, чтобы посмотреть, как у других получается с ними работать. После соединения и подключения загляните в Settings > Sound [Настройки > Звук], где вы должны увидеть нужное устройство в Output (если это гарнитура, она должна появиться также и в Input).

Вы увидите, что вашему устройству присвоен профиль — обычно Hi-Fidelity Playback (A2DP Sink) для динамика или Headset Head Unit (HSP/HFP) для гарнитуры — и, возможно, вы увидите, что через выпадающее меню поддерживается несколько профилей. Однако пока используйте настройки по умолчанию и нажмите на *Test Speaker*, чтобы проверить его работу (возможно, вам придется нажать несколько раз или даже переключить вывод, чтобы откликнулось подключение и вы увидели некий ответ).

Итак, устройство настроено и на Output [Вывод], и — если это применимо — на Input [Ввод], и вы можете попытаться использовать свое устройство в сочетании с приложением вроде *Rhythmbox*. Если звука нет, попробуйте временно переключиться опять на колонки вашего ПК в Settings > Sound, а потом снова



нажмите на свое устройство Bluetooth — теперь всё должно работать корректно.

Музыка на ваш ПК

Еще одно достойное использование Bluetooth — потоковая передача аудио с вашего телефона или иного аудиоустройства на более мощные колонки вашего ПК. Сначала убедитесь, что ваше устройство настроено на вывод на ПК — в Android, например, переход в Bluetooth Settings и нажатие на значок с шестеренкой Settings рядом с ярлыком вашего ПК должно вывести опции для вывода аудио с телефона или другого устройства на ПК — выберите последнее для своей музыки.

Теперь перейдите в Settings > Sound, выберите вкладку Input и проверьте, появилось ли ваше устройство как опция, показав, что оно поддерживается. Нажмите выпадающее меню Profile, для просмотра, что поддерживается. Типичная опция — High Fidelity Capture (A2DP Source).

Если у вас низкое качество — например, прерывается звук — есть несколько средств, которые можно попробовать. Сначала придвиньте свое устройство поближе к адаптеру Bluetooth вашего ПК (что особенно актуально для более старых моделей).

Настройка аудио бывает сложновата, но как только ваш динамик или гарнитура распознаются, настройте их через Settings > Sound.

СКОРАЯ ПОМОЩЬ

В процессе написания этой статьи мы начали с более старого адаптера Bluetooth 2.1 — он работал, но малость глючил. И наоборот, нам очень повезло с более новой моделью Bluetooth 4.0, особенно с ее качеством аудио.

» BLUETOOTH В КОМАНДНОЙ СТРОКЕ

Если вы бы хотели иметь возможность контролировать Bluetooth через терминал или другую оболочку, тогда все нужные вам инструменты уже установлены с помощью пакетов *bluez* и *bluetooth-tools*. Хотя традиционно для сканирования, соединения и подключения устройств используется *hcitool*, более простым инструментом является *bluetoothctl*. Вызовите в командной строке пакет:

```
$ bluetoothctl
```

Отсюда введите *help*, где вы найдете множество команд знакомого вида — например, 'scan on' для перехода в режим сканирования. Устройства будут появляться по мере распознавания, или введите 'devices' для получения обновленного списка. Как только вы найдете устройство, подключите его, вот так:

```
$ pair <device ID>
```

Device ID — это шестизначный шестнадцатеричный код (1A:2B:3C:4D:5E:6F), который появляется рядом с описанием устройства. Для экономии времени введите первую цифру ID и нажмите Tab для автозаполнения.

После соединения введите *trust <device ID>*, чтобы подготовить устройство к подключению, и затем *connect <device ID>* для завершения подключения.

Желая передать файлы по Bluetooth из командной строки, выйдите из *bluetoothctl* и введите

```
$ obexctl
```

И снова введите *help*, чтобы вывести полный список команд, которые охватывают подключение к удаленному устройству, отправку файлов, мониторинг и управление процессом передачи файлов. Полное руководство по инструментам и прочим компонентам *bluez* вы найдете на <https://docs.ubuntu.com/core/en/stacks/bluetooth>.



Введите `sudo nano /etc/bluetooth/main.conf` для доступа к главному файлу настройки Bluetooth. Большинство опций здесь закомментировано, но они сопровождаются объяснениями. Настройка аудио Bluetooth – через `sudo nano /etc/bluetooth/audio.conf`.

Отправка файлов

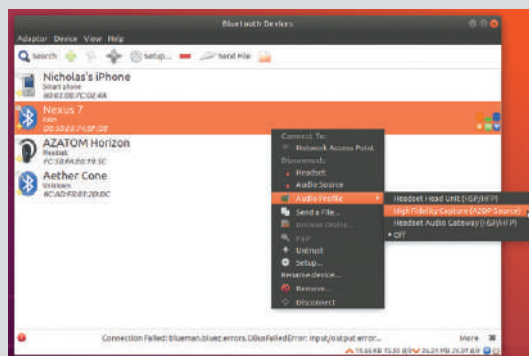
Технически вы уже должны быть в состоянии передавать файлы с вашего телефона на ПК. Откройте приложение, выберите файл и нажмите кнопку Share. Выберите Bluetooth из выпадающего меню, проверьте, чтобы ваш ПК был в списке как подключенный, и затем нажмите Bluetooth. Снова перейдите на ваш ПК, где вы должны увидеть уведомление о входящем

Технически также возможно переслать файлы с вашего ПК на мобильный. Нажмите на устройство в **Settings > Bluetooth** и выберите **Send Files...** Найдите и выберите свой файл, затем нажмите **Select**. Обратите внимание, что разрешены только определенные типы файлов — например, изображения. И снова вам надо будет принять соединение, но на сей раз через уведомления вашего мобильного. Строка состояния будет информировать вас в том числе и о скорости передачи. По завершении файлы сохранятся в специальной папке **Bluetooth** на вашем планшете или телефоне.

По завершении вернитесь к значку Bluetooth и нажмите Connect to Internet. Если всё пройдет успешно, вы сможете работать в Сети через интернет-соединение вашего телефона. Отключайтесь из того же диалога. Пользователи *Blueman* также могут инициировать соединение прямо из самого приложения — просто щелкните правой кнопкой по своему телефону и выберите Network Access Point для подключения и доступа к Интернету.

Вам не всегда понадобятся столь суровые меры. Иногда проблема решается простой перезагрузкой, или вам надо будет просто отключить устройства и потом подключить заново. Если это не поможет, отключение Bluetooth на обоих устройствах (если возможно), перезагрузка ПК и попытка переподключиться заново может решить вам проблему. А если это не сработало, подумайте

Инструмент *bluetooth-manager* предлагает альтернативный способ настройки и управления вашими устройствами Bluetooth.



об удалении устройства и о том, чтобы начать всё с нуля, создавая соединение с ним, как с новым устройством.

Но что произойдет, если сам Bluetooth перестанет работать? Один пользователь сообщил, что Bluetooth больше не работает после того, как он перевел Ubuntu 18.04 в режим ожидания (у нас при этом не возникло никаких проблем, так что нельзя утверждать, что это проблема повсеместная). Перезагрузка, отключение и новое включение Bluetooth не возымело никакого действия, так же, как и перезапуск сервиса Bluetooth, переустановка ключевых пакетов или запуск *rfkill*. В конечном итоге ему удалось вернуть всё к работе после загрузки с live CD, отключения Bluetooth оттуда и потом перезагрузки — тогда всё снова заработало нормально.

Еще один пользователь создал обходной путь для этой проблемы, создав пустой скрипт:

```
$ sudo nano /lib/systemd/system-sleep/Bluetooth
```

Введите следующие строки:

```
#!/bin/bash
if [ "${1}" == "post" ]; then
    sleep 5
    service bluetooth restart
fi
```

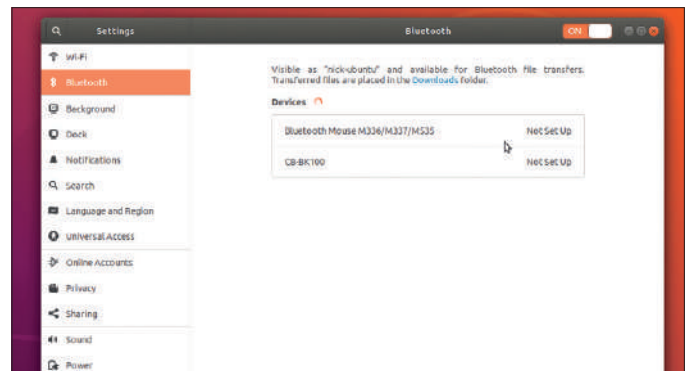
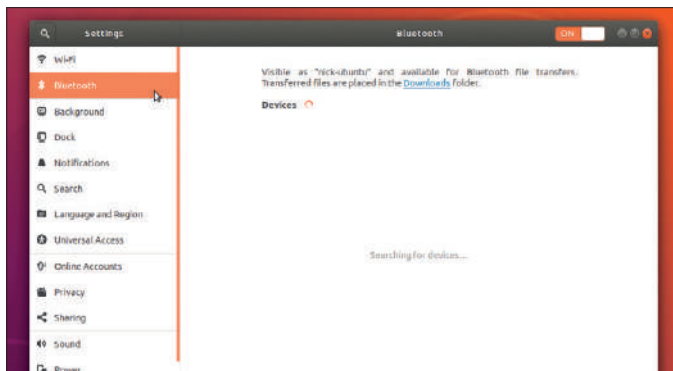
Сохраните и выйдите, затем сделайте скрипт исполняемым:

```
$ chmod +x /lib/systemd/system-sleep/Bluetooth
```

Помните, вам нужен этот скрипт, только если Bluetooth перестанет работать после перевода ПК в режим ожидания.

Жаждете найти решение для вашей конкретной проблемы? Форумы Ask Ubuntu (<https://askubuntu.com>) размещают множество вопросов от других пользователей Bluetooth — начните с 'Bluetooth Ubuntu 18.04' и сузьте параметры поиска отсюда. Или расширьте условия поиска в Google, чтобы найти также другие источники. **LXF**

НАСТРОЙТЕ УСТРОЙСТВА BLUETOOTH С ПОМОЩЬЮ ПК



1 Первые шаги

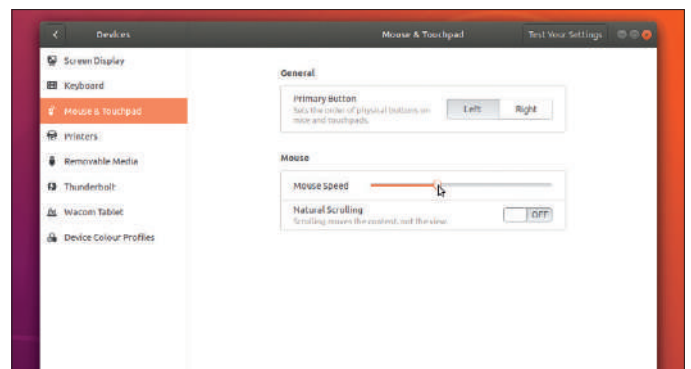
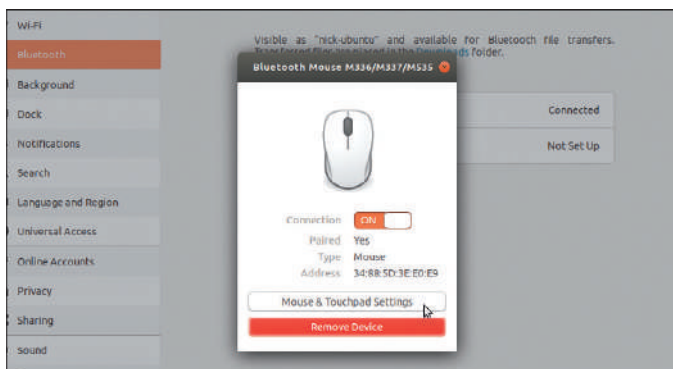
Нажмите на кнопку со значком сети в верхнем правом углу рабочего стола Nicholas's iPhone Network Gnome и затем нажмите на кнопку Settings. Выберите Bluetooth из левого меню (если его там нет, откройте терминал и введите

```
sudo apt-get install gnomebluetooth
```

чтобы добавить его). Если всё работает корректно, вы должны увидеть, как он ищет устройства и перечисляет их как видимые.

2 Соединение устройства

Теперь следуйте инструкциям, чтобы перевести ваше целевое устройство Bluetooth в режим соединения — обычно это поиск и удержание кнопки подключения, пока индикатор не замигает или оно не появится в Devices [Список устройств] как Not Set Up [Ненастроенное]. Нажмите на него и подождите, пока Ubuntu поменяет его состояние на Connected [Подключенное] или Disconnected [Отключенное]. Снова нажмите, если оно отключено, для его настройки.



3 Настройка устройства

Появится окно со списком устройств. Переведите соединение в On и подождите — через некоторое время оно должно измениться на включенное, и Paired покажет "Yes". Теперь вы должны быть в состоянии сразу же проверить устройство — например, пошевелив мышью — и оно должно работать.

Для дальнейшей настройки нажмите на соответствующую кнопку, чтобы перейти в соответствующий экран Settings.

4 Тонкая настройка устройства

Некоторые устройства подключаются автоматически — чтобы настроить их, просто щелкните по устройству, чтобы вывести его основной экран настройки, затем нажмите кнопку Settings, чтобы перейти в соответствующую область для дальнейшей тонкой настройки — например, настройки скорости курсора мыши. Кроме того, на этом экране можно временно отключить устройство или соединение с ним (для этого нажмите Remove Device [Удалить Устройство]).

Пишем свою электронную книгу

Джон Найт сперва не брался за написание электронных книг — не имея к тому ни склонности, ни документации; но обнаружил, что *Sigil* довольно удобен.



**НАШ
ЭКСПЕРТ**

Джон Найт опра-
вился от укуса
австралийского па-
ука, и теперь пишет
электронную книгу
о том, как играть
на барабанах.

3 адайте в Google поиск редакторов электронных книг [ebook], и там запрестит одно название: *Sigil* (<https://sigil-ebook.com>). На первый взгляд это озадачивает: несмотря на довольно ограниченный интерфейс, новые программы с похожими функциями, текучку разработчиков и частые реформы в управлении проектом, *Sigil* по-прежнему активен и популярен, и доступен напрямую из репозитория многих популярных дистрибутивов.

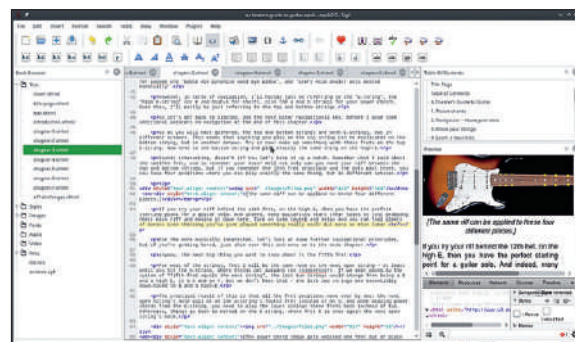
По информации на сайте, *Sigil* — многоплатформенный WYSIWYG-редактор электронных книг в формате epub с поддержкой UTF-16, проверки орфографии и полного ручного управления синтаксисом электронной книги. *Sigil* умеет импортировать файлы HTML, изображения и таблицы стилей с полной поддержкой Epub 2, хотя поддержка Epub 3 пока в разработке.

К сожалению, документация на сайте в основном касается установки и почти ничего не говорит о GUI или обычной работе. А значит, нам придется исследовать его самим...

Приступим к GUI

Открыв *Sigil* впервые, вы увидите панель инструментов и три панели окна. Начиная слева идет Book Browser, главным образом используемый для открытия всех файлов в вашей книге и управления ими. Сюда входят не только сам текст, но также изображения и даже мультимедиа аудио и видео. Более подкованные технически могут также управлять шрифтами и таблицами стилей CSS и редактировать такие файлы, как оглавление.

В центральной части окна находится основная область редактирования, предназначенная для создания текста или для ручного



Режим Code View — гвоздь программы *Sigil*. В процессе кодирования изменения в вашей электронной книге обновляются в окне предпросмотра в реальном времени. Супер!

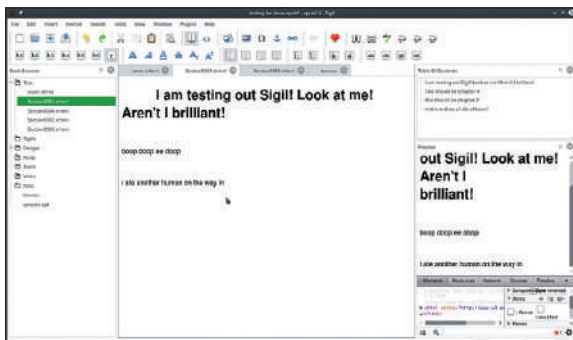
редактирования кода, но об этом чуть позднее. Очень удобно, что она использует вкладки, как браузер. Справа имеется панель Table of Contents [Оглавление], которая вступит в действие только позднее, когда вы создадите свое оглавление. Тем не менее, в предварительно подготовленной электронной книге она обычно действует как гиперссылки страниц, чтобы у вас было живое оглавление для быстрого доступа к любой главе.

К сожалению, одна из лучших функций *Sigil* не включена по умолчанию: функция Preview [Предпросмотр]. Прежде чем идти дальше, пожалуйста, включите ее, нажав View > Preview [Вид > Предпросмотр] или F10. Поверьте, она вам пригодится. По мере работы над вашей электронной книгой панель Preview будет обновляться в реальном времени, показывая вам, как она выглядит в реальном виде электронной книги. Это особенно полезно, если вы будете вносить какие-то изменения в коде; но мы оставим это на потом.

Кнопка вверх

Если мы посмотрим на основную панель инструментов, две самых важных кнопки *Sigil* размещаются там, в середине панели инструментов, и вся программа сосредоточена вокруг этих двух режимов: Book View [Режим книги] (отмеченный значком с книгой) и Code View [Режим кода] (отмеченный значком <>). При первом открытии *Sigil* по умолчанию используется Book View.

Book View — это место, где за дело берется редактор What You See Is What You Get (WYSIWYG) [Что видите, то и получаете], и хотя кажется, что вы просто просматриваете электронную книгу, все ваши действия — например, нажатие на пробел — превращаются



Режим *Sigil* Book View предлагает отличный редактор WYSIWYG для электронных книг. Элементы управления довольно минималистские, но на то есть причина...

СКОРАЯ ПОМОЩЬ

Не берите для проверки своего форматирования хорошие читалки электронных книг — они вам будут лить и скрывать ваши недочеты. Лучше тестируйте на ужасных читалках: если ваша электронная книга хорошо смотрится на них, то будет хорошо выглядеть где угодно!

в форматирование, так что будьте осторожны! Впрочем, функция WYSIWYG относительно ограничена, поэтому для управления чем-то более сложным вам придется перейти в Code View.

Открываете ли вы новый файл или заново запускаете программу, вы должны быть в Book View WYSIWYG-редактора — если нет, и вы находитесь в Code Editor, нажмите на значок с книгой, и мы начнем.

По новой странице и мигающему курсору вам не промахнуться... здесь всего минимально, в мире *Sigil*! Количество кнопок управления текстом уместно смотрелось бы в начале 1990-х, но скоро вы поймете, что на это есть причина.

Электронные книги ужасно капризны, и их рендеринг очень различается на разных устройствах. Поэтому вам нужно сохранять свое форматирование как можно более простым — если вы хотите что-то сделать с интересной графикой и раскладкой, лучше использовать нечто вроде *Scribus* и нечто вроде PDF в качестве формата.

Если вы побалуеете нас вводом названия главы в виде своего рода заголовка — на данный момент подойдет что угодно — и нажатием Enter для начала нового абзаца, а потом начнете вводить текст, следовать нашему уроку будет намного проще.

Возможно, вы заметили странные кнопки управления слева от главной панели инструментов, помеченные от h1 до h6 и p — они означают, является ли текст заголовком или частью обычного абзаца. Как и в обычном текстовом редакторе, h1 будет крупным шрифтом с жирным заголовком, h2 будет больше обычного шрифта и жирным, но не больше h1, и т.д. (думается, принцип вы поняли).

Пожалуйста, выделите название главы, который вы ввели, и нажмите h1, сообщая *Sigil*, что вы создали свой первый заголовок. Чтобы и дальше придерживаться правильной практики в электронных книгах, используйте h1 для названия каждой новой главы, а затем h2 для основных тем в главе. Кнопки h3 и далее используются для создания заголовков низших уровней, и хотя этих уровней допускается сколько угодно, лучше всегда помнить о простоте, и не рекомендуется выходить за пределы уровня h3.

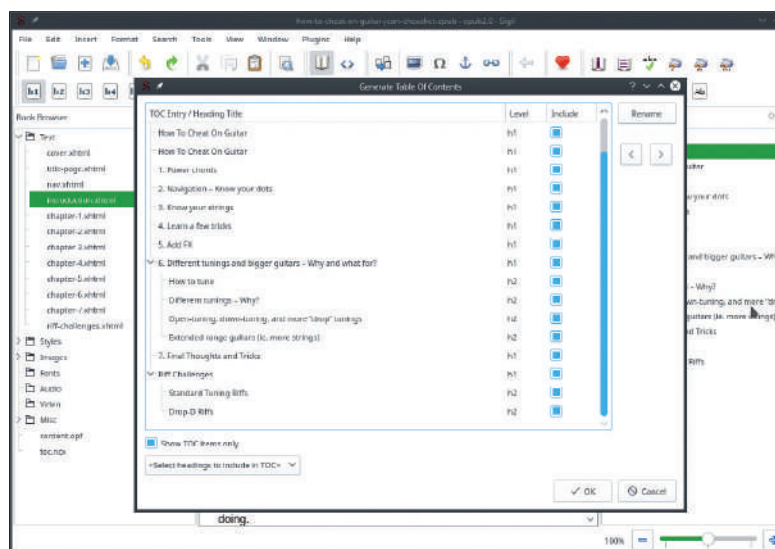
Не считайте это за занудство педанта — поверьте, позднее на нашем уроке всё это пригодится. Используйте h1 для названий глав и h2 для новых подразделов — вот и всё, что вам надо помнить.

После того, как вы ввели целую главу текста и вам надо начинать новую, обратитесь к Book Browser слева. Чтобы начать новую главу, лучше всего создать новый файл. Если были оставлены настройки по умолчанию, должен быть файл с именем **Section0001.xhtml**, выделенный зеленым; щелкните по этому файлу правой кнопкой и в выпадающем меню выберите Add Blank HTML file [Добавить пустой файл HTML].

Создастся новый файл с именем **Section0002.xhtml**, который откроется в новой вкладке. Одна из крутых вещей в *Sigil* — возможность работать над разными главами в разных вкладках, как в браузере. В панели Book Browser вы можете щелкнуть и перетаскивать файлы, чтобы изменить их последовательность. Щелкнув же по файлу правой кнопкой, вы можете выбрать — удалить его, переименовать или объединить с файлом выше, чтобы скомбинировать оба текста в одну главу.

Add Existing files [Добавить существующие файлы] позволяет импортировать любые уже имеющиеся у вас HTML-файлы в вашу книгу. Однако будьте осторожны, потому что если они не оптимизированы для электронной книги, они могут легко сломать ваше форматирование!

Если вы собираетесь постоянно использовать одну и ту же структуру, можете выбрать Add Copy [Добавить копию], чтобы продублировать существующий файл; это отличный способ получить своего рода шаблоны.



Прежде чем идти дальше, не забудьте создать титульную страницу. Это не квантовая физика и требует только базовых функций управления, и мы предоставим вам справляться с этим самостоятельно. Не стоит благодарности!

Генератор оглавления вызывает *Sigil* над другими методами.

Контент правит

Вы применяете правильную практику с названиями глав? Если да, то ваше усердие будет вознаграждено, когда вы возьметесь создавать Оглавление [Table of Contents, ToC].

Чтобы создать оглавление, нажмите Tools > Table of Contents > Generate Table of Contents [Инструменты > Оглавление > Создать оглавление]. Откроется окно со списком всех названий глав из вашей книги. Рядом с каждым заголовком будет стоять его уровень (h1, h2, h3 и т.д.) и окно-флажок с пометкой Include [Включить], где вы можете выбрать, что войдет в ваше Оглавление.

Если в вашей книге много заголовков, есть очень удобная функция фильтра в выпадающем окне, которая говорит <Select headings to include in TOC [Выбрать заголовки для включения в оглавление]>. Вы можете выбрать Up to level 1 [До уровня 1], Up to level 2 [До уровня 2], None [Ни одного] и All [Все].

СКОРАЯ ПОМОЩЬ

Переместив курсор куда-нибудь в Book View, вы увидите соответствующий код, выбрав Format и затем Show Tag [Показать теги].

» УКАЗАНИЯ ПО ХОРОШЕЙ ПРАКТИКЕ

Рекомендации по электронным книгам довольно хаотичны. Из всего написанного, ближе всего к стандарту руководство по стилям Smashwords.

Для непосвященных, Smashwords — это компания, которая берет вашу электронную книгу и размещает ее на всех основных платформах за весьма скромную комиссию. Поэтому в ее интересах, чтобы вы сделали форматирование правильно и она могла выполнить свои обязанности, не занимаясь правкой вашей книги. У нее есть 117-страничное руководство по стилям (www.smashwords.com/books/view/52), охватывающее все основные вопросы форматирования, чтобы ваша электронная книга была принята на всех платформах. Чем примитивнее ваше форматирование, тем проще конвертировать книгу в разные форматы, и тем большего числа ошибок вам удастся избежать. Если вы сможете продаться через этот огромный список, ваша книга будет в наилучшей возможной форме.

Пара более практических советов: средний размер книги Kindle — около 2,6 МБ, так что старайтесь не вставлять картинок размером более нескольких сотен КБ. Перед представлением всегда пропускайте свою книгу через валидатор W3 (<http://validator.w3.org>): продавцы вроде Apple требуют 100% чистоты орфографии, чтобы книга была допущена в их магазины, а программа проверки *Sigil* не настолько мощная. И позаботьтесь, чтобы ваша цена заканчивалась на «,99» (например, \$0,99, \$9,99, и т.д.). Это стандартная практика и требование ряда магазинов.

СКОРАЯ ПОМОЩЬ

В HTML разобраться нетрудно, и в основном всё происходит между открывающим тегом типа `<html>` и закрывающим тегом типа `</html>`. Просмотр кода нескольких простых сайтов должен дать вам общее представление о его работе. Желая узнать больше, загляните на www.w3schools.com, там есть бесплатные уроки.

Наличие Оглавления электронной книги дает возможность перехода между главами. Иные магазины даже хотят, чтобы в начале книги вы вставили вторую HTML-версию.

При правильной работе с электронной книгой уровень 1 отображает только названия глав, а уровень 2 также предоставит вам подразделы в главе.

Если вы всё перепутали, Tools > Table of Contents > Edit Table Of Contents [Инструменты > Оглавление > Редактировать оглавление] позволит вам перемещать пункты вверх и вниз по Оглавлению или удалить ненужные пункты.

Превращаемся в meta с помощью метаданных

Прежде чем заваливать мир своей книгой, отредактируйте ее метаданные. Эти метаданные предоставляют такую же информацию, как в MP3, где они сообщают про исполнителя и длительность трека, но с очевидными эквивалентами для электронной книги, такими как ваше имя, ваш издатель и т.д. Эта информация важна для кроссплатформенного распространения и жизненно важна, если вы продаете свою книгу на свободном рынке. Не полагайтесь исключительно на имя файла или колонтитулы страниц, потому что это в лучшем случае ненадежно и выглядит исключительно непрофессиональным.

Нажмите Tools > Metadata Editor [Инструменты > Редактор метаданных], и окно покажет вам несколько с виду непонятных полей для ввода данных. Не паникуйте, разобраться с ними несложно. Строка `dc:language` сообщает вашим читателям, что книга написана на английском — оставьте это поле в покое; а если она на другом языке, мы расскажем об этом через минуту. Самая важная строка — `dc:title` [No data]. Отмените запись [No data] в поле Value и введите название своей книги. Не используйте одни заглавные буквы и старайтесь не применять спецсимволов, поскольку это может помешать вашей книге появиться в определенных точках распространения.

Однако эти поля — только минимум, достаточный лишь для случайного документа; а если вы продаете собственную книгу, нужно добавить больше информации. Нажмите на кнопку Add Metadata [Добавить метаданные] для просмотра всех имеющихся полей, таких как Author [Автор], Publisher [Издатель], ISBN и так далее. Милая функция — опция Contributor [Благодарности], где вы можете добавить тех, кто помог в создании вашей книги.

Если ваша книга не на английском, пора учесть это свойство. Выберите Language [Язык] в списке опций, и появится новое окно со списком языков, из которого можно выбрать. Выбор автоматически добавится в строку `dc:language` с правильным кодом языка. Если ваша книга включает несколько языков, этот процесс можно повторить. Любую информацию, не нужную вам в метаданных, можно удалить, выделив строку и нажав Remove.

Прикрой меня!

Чтобы назвать вашу книгу готовой, отнюдь не мешает обложка, а в некоторых магазинах она к тому же обязательна. Само по себе добавление обложки не представляет трудности. Просто нажмите

Tools > Add Cover [Инструменты > Добавить обложку], и появится новое диалоговое окно. Нажмите на кнопку Other Files [Другие файлы] справа, выберите изображение для обложки — и вуаля! Создается новый файл, `cover.xhtml`, и откроется в новой вкладке с изображением обложки вашей книги.

Итак, это было просто, но, как и во многом в мире электронных книг, здесь есть масса ситуаций и инструкций, которые надо выполнять. Smashwords рекомендуют делать обложку 1600 пикселей в ширину на 2400 пикселей в высоту — хотя здесь есть возможность проявить гибкость — но всё равно хорошо бы придерживаться соотношения 1×1,5.

Естественно, ваша обложка должна иметь форму книги, так что позаботьтесь сделать ее в виде вертикального прямоугольника и проверьте, что она четкая и читаемая при уменьшении до эскиза и что она не зернистая, не размытая и не пикселист.

В зависимости от издательства, на обложке может быть запрещен показ наготы или непристойности [Ред.: — Пингинов?]; гиперссылки или неуместная реклама тоже автоматически заблокируют вашу книгу. Помимо необходимости указывать название и имя автора, информация на обложке должна соответствовать информации в книге и в метаданных.

Режим кодирования

Именно здесь редактирование вашей электронной книги станет потруднее. Хотя вы можете написать целую книгу, не прибегая к Code View, ваше форматирование будет весьма рудиментарным. Если вам нужен больший контроль за форматированием, то именно в Code View вы его добьетесь — и большая его часть основана на HTML-коде. Хотя с технической точки зрения для создания электронной книги знать HTML не обязательно, стоит всё же хотя бы немного его изучить, если вас серьезно заботит успех.

Кодирование на HTML для среднестатистического обывателя звучит жутко, но, пожалуйста, не робейте — ведь это, вероятно, самый простой способ кодирования! Его нетрудно освоить, и даже час изучения сыграет важную роль в ваших навыках форматирования.

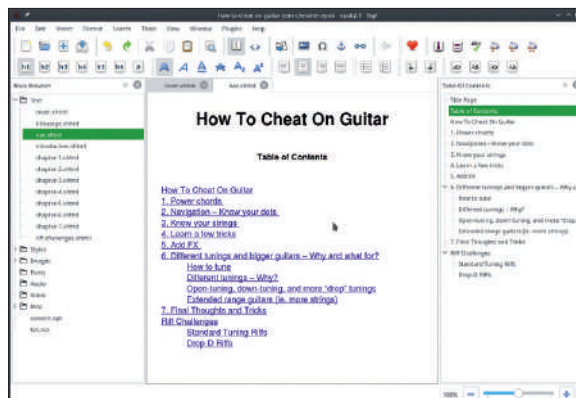
Итак, хотя электронные книги по большей части являются собой HTML, как сайты, устройства для их чтения очень сильно отличаются друг от друга по способу рендеринга контента. Следовательно, ваш код должен быть как можно более общим и не экзотичным, чтобы максимально повысить шансы на корректное отображение вашей страницы. Если вы мыслите, как дизайнер HTML, попытайтесь представить себе, что вам надо сверстать сайт в 1995 г., когда в браузерах было полно ошибок, а Всемирная Паутина была во младенчестве!

К счастью для вас, авторы *Sigil* прекрасно это сознавали при создании программы, и встроили в нее блестящие инструменты, чтобы сэкономить ваше время и уберечь вас от хронической головной боли.

Самой первой функцией должна бы быть сама панель Preview. Независимо от того, находитесь ли вы в Book View или Code View, панель Preview будет обновляться в реальном времени. Если у вас в коде есть ошибка, вы узнаете об этом сразу же, потому что появится розовое окно со следующим сообщением:

This page contains the following errors: error on line x at column y: expected 'z'
Below is a rendering of the page up to the first error.
[Данная страница содержит следующие ошибки: ошибка в строке x в столбце y: ожидается 'z'.
Ниже — рендеринг страницы до первой ошибки]

Естественно, реальные координаты мы заменили на xyz, но они означают, что вы и сразу же узнаете о наличии ошибки в коде, и сможете сразу перейти к ней, обойдясь без ручного пролистывания всего кода в поисках ошибки.



Возможно, одни из лучших компонентов *Sigil* — его валидаторы кода и опции чистки. Чтобы убедиться, что код вашей книги — в хорошем рабочем порядке и будет корректно отображаться на большинстве устройств, нажмите **Tools > Well-Formed Check EPUB** [Инструменты > Проверка правильности EPUB]. Ваш файл **.epub** просканируется на предмет наличия проблем. По завершении внизу окна откроется новая панель **Validation Results** [Результаты проверки], предупреждающая вас и о критических ошибках, и о плохом коде.

Учтите, что даже если вы не кодировали в HTML сами — вы, например, могли воспользоваться онлайн-конвертором — всё равно есть шанс, что в ваш файл вкрался плохой или неркомендуемый код. Автоматическое кодирование хорошо ровно настолько, насколько хорош создавший его кодер, поэтому всегда запускайте программу проверки, независимо от вашей уверенности в качестве своего кода.

Если вы решите доверить исправление указанного кода *Sigil*, а не заниматься этим самостоятельно, сделайте это, нажав на **Tools > Reformat HTML > Mend All HTML Files** [Инструменты > Переформатировать HTML > Исправить все файлы HTML].

Очистим город

Мы обнаружили, что *Sigil* весьма неплохо справляется с исправлением ошибок в файлах, и вы можете подумать о предоставлении *Sigil* возможности вычистить плохие фрагменты вашего кода, нажав **Tools > Reformat HTML > Mend And Prettify All HTML Files** [Инструменты > Переформатировать HTML > Исправить и улучшить все файлы HTML]. Однако делайте это скрепя сердце: мы обнаружили, что некоторые части форматирования из-за автоматического улучшения были выбиты, так что рекомендуем сначала сохранить работу!

Что касается практики кодирования — если вы уже работаете в HTML, вам придется несколько перестроиться. Между работой над сайтом и работой над электронной книгой есть разница, в частности, когда речь идет о чистом пространстве.

В частности, к незакрытию тегов в электронных книгах отношение весьма нетерпимое, и если, как и большинство людей, вы применяли для нового абзаца более вольный **<р>**, вам придется поднапрячься и не забывать ставить также и закрывающий тег. Так что вместо этого используйте **<р> </п>**, даже если вы просто добавляете пустое пространство.

Вам также придется отказаться от тега **
**, поскольку он тоже не годится. Вместо него надо применять **
, что очень странно выглядит для тех, кто привык к HTML двадцатилетней давности! Мы пытались применить тот же принцип к тегу абзаца, используя **<р/>, но когда мы сохранили файл, *Sigil* автоматически изменил его на **<р> </п>**, так что не советуем этого делать — сохраняйте оба тега!

Однако вы можете всё смешивать и находить совпадения, и весьма часто можно увидеть, что в электронных книгах при добавлении пустого пространства используется **<п>
 </п>**. К счастью, старый добрый ** ** по-прежнему годится для добавления односимвольного пробела, что очень удобно, поскольку доступ с помощью табуляции в электронных книгах не работает. Иными словами, на эту практику до сих пор смотрят косо, и рекомендуется попытаться как-то переосмыслить необходимость красной строки, потому что некоторые читалки электронных книг вообще ее игнорируют!

Эпилог

При таком количестве ограничений и поначалу пугающем интерфейсе будет трудно понять сразу, в чем прелесть *Sigil*. Однако

» КОНВЕРТИРУЕМ С PANDOC

Если интерфейс *Sigil* вас раздражает, вы, возможно, предпочтете просто писать в текстовом редакторе и затем конвертировать результаты в **.epub**. Вы можете обратиться к онлайн-конвертору, но мы любим инструмент *Linux, pandoc*, и доверяем ему: он справляется с конвертированием на отлично.

Если вы тоже хотите попробовать *pandoc*, следует использовать формат **.docx** MS Word, который, похоже, при конвертировании работает лучше всего. Издатели всё равно думают, что файл **.docx** у вас где-то есть, так что не будем раскисывать лодку.

Pandoc — это инструмент командной строки, но не бойтесь, он весьма прост в использовании. Обычная команда выглядит так:

```
$ pandoc original-wordprocessor-file.docx -t epub -o your-ebookfile.epub
```

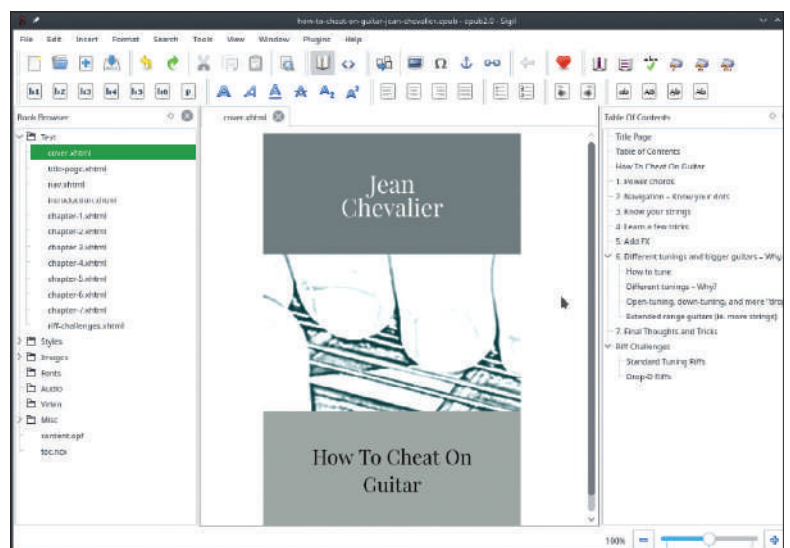
Естественно, *pandoc* — это команда, а далее идет файл **.docx**, который вы хотите конвертировать от **<i>** до **</i>**. Параметр **-t** — чтобы сообщить *pandoc*, в какой тип файла конвертировать, в данном случае это **epub**. И, наконец, параметр **-o** сообщает *pandoc* имя файла, который вы хотите получить в конце в **epub**.

Однако будьте осторожны с конверсией файлов. Велика вероятность, что она принесет в форматирование мусорный код, и справиться с ним будет настоящим кошмаром. Если вам всё же придется пойти по пути конвертирования, сохраняйте свое форматирование как можно более простым и понятным человеку. Если хотите всё сделать чуть интереснее, научитесь делать это в HTML в *Sigil*, а не в текстовом редакторе.

не опускайте рук, и ответ станет до боли очевиден: *Sigil* работает именно так, как надо работать.

Его «ограничения» оказываются истинным благословением, когда вы представляете свою книгу издателю и оказывается, что ее надо очень сильно упрощать. И хотя кодирование в HTML и функция проверки сначала сильно пугают, они прекрасно сотрудничают с редактором WYSIWYG, создавая почти идеальный баланс.

Иногда *Sigil* огорчает, но настойчивость, с которой программа пытается всё сделать правильно — а не легко и просто — означает, что ее культовая популярность среди создателей электронных книг по всему миру в обозримом будущем никуда не денется. **LXF**



Большинство магазинов электронных книг хотят, чтобы у вашей публикации была обложка. Сделайте ее привлекательной — это очень важно для продаж!

» ПОДПИШИТЕСЬ НА ЖУРНАЛ **LINUX FORMAT** на www.linuxformat.ru/subscribe!

КОРОТКОВОЛНОВОЕ РАДИО

Настройка на весь мир

Думаете, что коротковолновое радио умерло? Подумайте еще раз, пока **Майк Бедфорд** учит вас слушать и декодировать далекие радиостанции...



**НАШ
ЭКСПЕРТ**

Майк Бедфорд возился с радио еще в школе, аж с ламповыми приемниками, и уверяет, что владеет азбукой Морзе.

СКОРАЯ ПОМОЩЬ

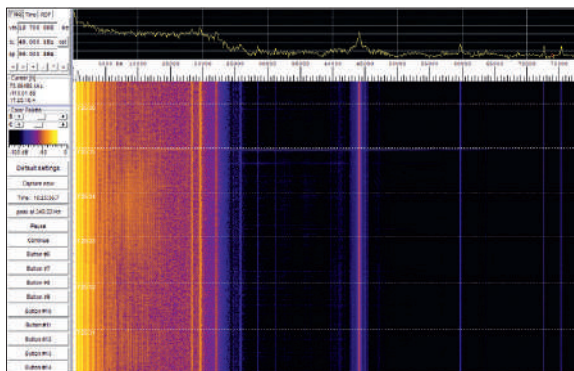
Несмотря на то, что на частоте ниже 1 кГц распознаются исключительно гармоники электросети, эти частоты делают возможным общение в морской воде. Когда-то у американского ВМФ была коммуникационная система для подводных лодок на частоте 76 Гц.



Ивья, как живем мы, в мире повсеместных коммуникаций, легко получить ложное впечатление о технологии радио, что лежит в основе всего этого. Хотя всё работает на Wi-Fi и сотовой технологии, обе не предлагают ничего, хотя бы близкого к глобальной коммуникации. Wi-Fi охватывает радиус, измеряемый метрами, и даже мобильный сигнал справляется только с несколькими километрами до ближайшего ретранслятора.

Обычные стандарты беспроводной коммуникации — очень высокие, или микроволновые частоты, на которых ширина полосы пропускания обеспечивает высокую скорость передачи данных ценой уменьшения диапазона. На более низких частотах, например, в диапазоне длинных и средних волн, охват достигает сотен километров, а коротковолновой диапазон способен охватить весь мир. Конечно, то, как сигналы наших мобильных телефонов маршрутизируются по всему миру через десятки коммутаторов — это современное чудо, как и проходящая по дну моря оптоволоконная сеть, обеспечивающая соединение на огромных расстояниях. Тем не менее, есть нечто волшебное в возможности получать сигналы со всего мира, не имея инфраструктуры многомиллионной стоимости, которая обеспечивает современные коммуникации.

Здесь мы узнаем, как получать глобальные радиосигналы. Мы узнаем, как получать низкочастотные радиосигналы (VLF), используя только звуковую карту вашего ПК, как настроиться на коротковолновые сигналы с помощью программного web-радио, и увидим, как конвертировать поток недоступных для понимания серий коротких и длинных сигналов,



Чтобы получать сигналы с нижнего конца радиодиапазона, вам понадобится только ПК и кусок проволоки.

встречающихся в коротковолновом диапазоне, в понятные сообщения или изображения.

СНЧ: Как низко вы готовы пасть?

Диапазоны VLF, ULF и SLF (см. врезку справа внизу на стр. 71) включают частоты, которые часто именуют аудиочастотами (AF). Это не значит, что мы в состоянии их услышать, поскольку радиоволны — это электромагнитное излучение, независимо от их частоты, а наши уши способны определять только механические колебания. Зато радиоволны можно получать с помощью звуковой карты в ПК. Но вместо микрофона для конвертирования звуковых волн в электрические сигналы используется антенна для конвертирования радиоволн в электрические сигналы. Это будет довольно примитивный приемник, так как короткие антенны неэффективны, а звуковая карта менее чувствительна, чем специальный низкочастотный приемник VLF, но это всё равно неплохо для начала.

Вначале мы рекомендуем вам протянуть 10 метров провода по дому и соединить его с 3,5-мм гнездом, которое можно подключить ко входу для микрофона вашей звуковой карты. Если у вас нет такого гнезда, можно разрезать звуковой кабель на две части и подсоединиться напрямую к двум центральным жилам.

С такой конфигурацией вы сможете просто слушать полученные сигналы, но по причинам, которые мы рассмотрим далее, вы услышите только постоянное гудение. Намного лучше использовать под WINE программу *Spectrum Lab*, скачав ее с <https://www.qsl.net/dl4yh/spectra1.html>. Почти единственная настройка, которая от вас потребуется — выбрать звуковую карту в качестве устройства ввода в Audio settings, I/O device selection [Настройки аудио/Выбор устройства ввода/вывода] в меню Options и указать частоту дискретизации. Максимальная частота, которую вы сможете получать составляет половину частоты дискретизации, поэтому, выбрав 192 кГц, вы сможете получать сигнал от 0 до 92 кГц.

Отображение по умолчанию показывает кривую силы сигнала относительно частоты вверх и спектрограмму [waterfall display] ниже. Спектрограмма показывает изменение сигнала определенной частоты во времени с цветным представлением силы сигнала. Если вам тяжело работать со *Spectrum Lab* в Wine, то попробуйте *Audacity*, но обязательно выберите вид Spectrogram и частоту дискретизации 192 кГц.

Самый заметный сигнал, который вы увидите, будет 50 Гц в Великобритании или 60 Гц в США, и есть также сигналы, кратные 50 Гц или 60 Гц. Они генерируются электросетью, поскольку

провода в вашем доме выступают в роли хорошей передающей антенны. 50 или 60 Гц — это частота электросети, а кратные — гармоники, и они заметны до 22 кГц. Это источник гудения, которое вы бы услышали, если бы слушали сигнал ухом. Помимо сигналов электросети, вы можете услышать и другие мощные сигналы — они будут созданы вашим ПК. Однако один «настоящий» радиосигнал, который вы, возможно, увидите, будет на 60 кГц. Сигналы на этой частоте часто слышны по обе стороны Атлантики — позывной MSF в Великобритании и WWVB в США, и их цель — передача сигналов точного времени.

Имея антенну получше, вы, вероятно, увидите больше искусственных сигналов, а также, ниже диапазона 9 кГц могут оказаться природные радиосигналы, называемые атмосферными помехами [sferics]. Они генерируются при взаимодействии молнии со слоями атмосферы, чаще всего на рассвете и закате, и принимают разные формы. Например, так называемые свистящие атмосферники уменьшаются в частоте через одну-две секунды и появляются на спектрограмме в виде кривых. Более длинная и высокая антенна, размещенная на улице, будет лучше, но знайте, что высокие антенны могут накопить достаточно высокое напряжение, чтобы спалить видеокарту, так что нужна защита — вы найдете подробности онлайн.

Широковещательные станции

Нижняя часть радиодиапазона весьма интересна, поскольку доказывает: всё, что вам нужно, чтобы принимать радиосигналы — это ПК и кусок провода. Но если мы перенесем внимание в коротковолновой диапазон, там нас ждут более интересные сигналы. Не так давно вам требовался радиоприемник, но сейчас, благодаря web-приемникам SDR, вы можете слушать длинно-, средне- и коротковолновые диапазоны, вообще не тратясь на оборудование.

Традиционные радиоприемники содержат много аналогового оборудования для извлечения сигнала на одной определенной частоте, и преобразуют его в аудиосигнал. В программно-определяемой радиосистеме (software defined radio, SDR) сигнал с антенны оцифровывается с помощью АЦП, чтобы программа могла выполнить функции, которые в противном случае возлагались бы на аналоговую электронику. В отличие от обычного приемника, SDR могут получать сигналы на нескольких частотах одновременно. Эта функция позволяет размещать их в Сети, так что разные пользователи могут слушать разные частоты.

Это вовсе не то же самое, что интернет-радио в обычном смысле. Вещание последнего в Интернете осуществляется потоковым транслятором, а web-приемники SDR дают возможность слушать любые сигналы, которые можно услышать — напрямую через длинно-, средне- и коротковолновые диапазоны — там, где располагается сервер.

Вы найдете длинный список web-приемников SDR на www.websdr.org — и убедитесь, что они размещаются по всему миру. Некоторые охватывают только любительские диапазоны, а часто — только коротковолновые, но есть и работающие на низких и очень низких частотах. Для начала неплохо будет использовать локальный сервер (хотя в свое время вы, вероятно, захотите попробовать и другие), потому что это даст вам иллюзию, что вы могли бы принимать и сами, имея вы необходимое оборудование.

Один из немногих широкополосных приемников, в противоположность чисто любительским диапазонам, можно найти на <http://websdr.ewi.utwente.nl:8901>. Web-SDR общего охвата недоступны в США, однако есть один в Вашингтоне, округ Колумбия, на <http://na5b.com:8901>, который охватывает многие интересные частоты. Отобрав страницу, вы поймаете звук с любой заданной частоты, но пока вы не настроитесь на радиостанцию, вероятнее всего, услышите только шум. Прокрутите страницу, пока



Web-SDR дают возможность слушать коротковолновые диапазоны, не разоряясь на приобретении дорогого оборудования.

не увидите спектрограмму с разными элементами управления под ней. Если вы живете в Великобритании, можете начать с кнопки AM и ввести в окне частоты 198. Вы услышите BBC Radio 4 в длинноволновом диапазоне на 198 кГц. Может, это и не слишком интересно, но дает вам шанс услышать Web-SDR в действии. Однако отсюда мы начнем нашу разведывательную экспедицию.

В некоторых SDR (в т.ч. в Университете Твенте города Энсхеде, Нидерланды) внизу спектрограммы аннотировано несколько диапазонов частот, в основном окрашенных зеленым или красным. Для начала рассмотрим красные, коротковолновые вещательные диапазоны, где вещают такие станции, как Всемирная служба BBC и прочие.

Если ваш SDR не различает вещательные и любительские диапазоны, загляните на www.hfradio.org.uk/html/b_cast_radio.html. Разные коротковолновые частоты удобны для приема в разных частях света в разное время, и в конкретное время вы найдете очень мало станций в конкретных диапазонах. Итак, при выделенном 'AM' ищите вещательный диапазон с избытком станций, как показано цветными линиями на спектрограмме. Теперь настройтесь на этот диапазон, нажав на черную область под дисплеем и увеличив масштаб с помощью кнопки 'zoom in'. Затем вы сможете выбрать отдельные станции — или нажав на черную область, или с помощью кнопок с плюсом и минусом.

СКОРАЯ ПОМОЩЬ

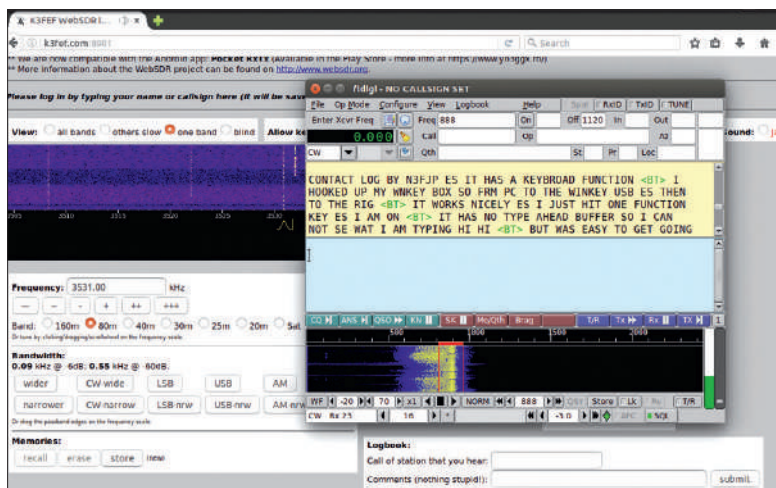
Опытные операторы до сих пор способны лучше переводить азбуку Морзе, чем большинство программных декодеров. Но более новые режимы передачи данных могут декодировать сигналы настолько слабые, что их невозможно расслышать ухом.

» РАДИОДИАПАЗОНЫ

Международный союз электросвязи [International Telecommunication Union, ITU] разделяет радиоспектр на диапазоны, каждый из которых представлен декадой частоты. SHF (сантиметровые волны) охватывает от 3 ГГц до 30 ГГц и включает 5,8-ГГц ISM-диапазон, используемый Wi-Fi плюс спутниковым ТВ. Далее идет UHF (дециметровые волны), охватывающий от 300 МГц до 3 ГГц, и здесь мы находим 2,4-ГГц ISM-диапазон, который тоже используется для Wi-Fi, Bluetooth, мобильных телефонных сетей и наземного ТВ. VHF (метровые волны) охватывает от 30 МГц до 300 МГц и используется для FM- и DAB-радио.

Затем идут HF (декаметровые волны) — 3–30 МГц, MF (гектометровые волны), от 300 кГц до 3 МГц, это средневолновой диапазон. Далее — LF (километровые волны), от 30 кГц до 300 кГц. Диапазоны с еще более низкими частотами используются для специализированных целей, но от этого не менее интересны. Это VLF (мираметровые волны), от 3 кГц до 30 кГц, ULF (гектокилометровые волны), от 300 Гц до 3 кГц, SLF (мегаметровые волны), от 30 Гц до 300 Гц, и ELF (декамега-метровые волны), от 3 Гц до 30 Гц.

Вместо длины волны радиоспектр теперь обычно именуют по частоте. Длина волны обратно пропорциональна частоте, и различается от 10 мм в крайне высокочастотном до 100 000 км в крайне низкочастотном диапазоне.



Созданная для распознавания на слух, азбука Морзе может быть декодирована с помощью *fldigi* и *MultiPSK*.

СКОРАЯ ПОМОЩЬ

Вы найдете станции, не попадающие ни в вещательный, ни в любительский диапазон. Некоторые просто передают ряд чисел. Обычно считают, что они либо шпионские, либо относятся к миру организованной преступности.

В диапазонах 31 м, 22 м и 25 м мы услышали множество станций на многих языках, в том числе на китайском, испанском, арабском, русском, польском, французском и хинди, а также на английском. Язык не обязательно определяет страну вещания. Если вы не слишком хорошо знаете иностранные языки, лучше будет найти англоязычные станции, чтобы выяснить, откуда идет передача. В порядке альтернативы поищите на www.shortwave-schedule.com/index.php, но помните, что частота, отображаемая в Web-SDR, не обязательно будет центральной частотой, указанной на сайте, так что можете попробовать соседние частоты.

Еще одна опция с некоторыми Web-SDR — увеличение масштаба, и она определит те станции, о которых ей известно. Всего за несколько минут мы определили Китайское международное радио, Радио Сауди, Румынское международное радио, Голос Кореи и BBC с достаточной достоверностью.

Любительское радио

Возможность слушать вещательные станции с другой стороны земного шара, принимая их на расположенный рядом приемник без помощи ретрансляторов — настоящее откровение

» ЖАРГОН РАДИОЛЮБИТЕЛЕЙ

Сообщение азбукой Морзе или в иной форме текстового сообщения в любительском диапазоне сложнее понять, чем голосовую передачу. Есть программы для декодирования сигналов в текст, но этот текст будет неизбежно содержать сокращения и жаргон. Здесь мы покажем вам, как перевести то, что иначе покажется бессмыслицей.

Любительский контакт часто начинается с того, что одна станция передает CQ (seek you — ищу тебя). Станция, передающая такой сигнал, приглашает к общению другие станции, хотя она может указывать CQ DX, что означает поиск только удаленных станций. Эта передача принимает форму типа “CQ CQ CQ DE A0AAA”, часто повторяемую несколько раз, где A0AAA — позывные передающей станции. Ответ другой станции будет чем-то вроде “A0AAA DE Z9ZZZ”, где Z9ZZZ — позывные отвечающей станции. Если завязывается контакт, обе станции будут начинать и заканчивать свои сообщения фразой “A0AAA DE Z9ZZZ”. Во время контакта каждая станция обычно указывает свое имя и местоположение, для чего используется аббревиатура QTH. Каждая станция также обеспечивает отчет о сигнале — три цифры, например, 599, где первая цифра — это четкость (до пяти), вторая — сила сигнала (до 9), а третья — тон (до 9, но неизменно 9). Отчет 599 часто сокращается до 5NN. Периодически контакты бывают более многословными и, следовательно, более четкими и понятными.

и впечатляющая иллюстрация возможностей коротковолнового диапазона. Еще более впечатляет слушание любительских радиостанций, поскольку они вещают на мощности всего в несколько сотен ватт по сравнению с сотнями киловатт, используемых большинством вещательных станций. Более того, большинство любительских радиопередач ведутся по-английски, отчего их легче распознать. Радиолюбители стараются завести как можно больше контактов на как можно большем диапазоне, и обозначают себя своими позывными [callsign], именем и местоположением каждый раз, когда выходят на связь с новой станцией.

Поскольку мы теперь ищем любительские станции, займемся любительскими частотами — в некоторых Web-SDR они отображаются зеленым (или справьтесь со списком). Как и с вещательными диапазонами, ищите тот, где окажется плотный трафик, и увеличьте его масштаб. Скорее всего, вы найдете локальные станции в диапазоне 160 м, немного более отдаленные станции в диапазонах 40 м и 80 м, и станции по всему миру в диапазоне 20 м и ниже. Для начала мы попытались обнаружить голосовые сообщения, так как их проще всего понять.

Вместо метода AM [амплитудная модуляция], применяемого вещательными станциями, радиолюбители используют SSB [однополосная модуляция с подавлением несущей частоты], и если вы слушаете в диапазоне ниже 10 МГц, нажмите на кнопку LSB, или, для частот выше 10 МГц, используйте USB. Обнаружив сигнал, похожий на писк, пока что игнорируйте его и поищите станции на более высокой частоте внутри диапазона. В конце концов вы найдете сигнал, который сможете понять, но очень важна настройка, так что вам, вероятно, придется использовать кнопки плюс и минус, чтобы сделать сигнал более четким.

Кроме того, помните, что, найдя общение между двумя станциями, вы, возможно, сможете услышать только одну из них. Тогда, если передает неслышная станция, вам придется подождать передачи той, которую слышно. Если оператор не говорит, где он находится, вы сможете определить страну по позывным. Закон требует, чтобы префикс позывных радиолюбителей определялся страной. Список префиксов вы найдете на <http://rsgb.org/main/operating/licensing-novs-visitors/international-prefixes>.

Передача данных

Наслушавшись любительских голосовых передач, давайте перейдем к чему-нибудь поинтереснее. Итак, обратитесь к низкочастотной области любительского диапазона и настройтесь на один из сигналов, который появляется на спектрограмме как более узкая линия по сравнению с голосовым сигналом. Если вам надо уменьшить помехи, выберите CW вместо LSB или USB, но это потребует еще более тонкой настройки. Вы обнаружите, что большинство этих сигналов — азбука Морзе, которую вы, возможно, узнали, хотя вряд ли смогли прочитать. Фактически, если сообщение морзянкой передается достаточно медленно, то на спектрограмме вы сможете увидеть, что сигнал прерывистый и соответствует точкам и тире, из которых состоят буквы.

Хотя вы не сможете расшифровать морзянку на слух, с этим справится ПО, так же как и с декодированием множества других типов любительских передач данных. *Fldigi* — одна из немногих, доступных в Linux, но хотя она хорошо переводит морзянку, она не поддерживает такого же количества передач данных, как *MultiPSK*. Это приложение Windows, но оно хорошо работает в WINE, и за это мы его любим.

Чтобы декодировать сигнал, надо направить полученное аудио с web-браузера, который отображает Web-SDR, на аудиовход декодирующей программы. Вы можете сделать это физически, соединив проводом выход и вход вашей звуковой карты, или программно, используя *PulseAudio*, которое надо сначала установить, если оно не входит в ваш дистрибутив.



SSTV медленный и картинка часто не идеальна, но всё же позволяют передавать изображения по всему миру без ретрансляторов.

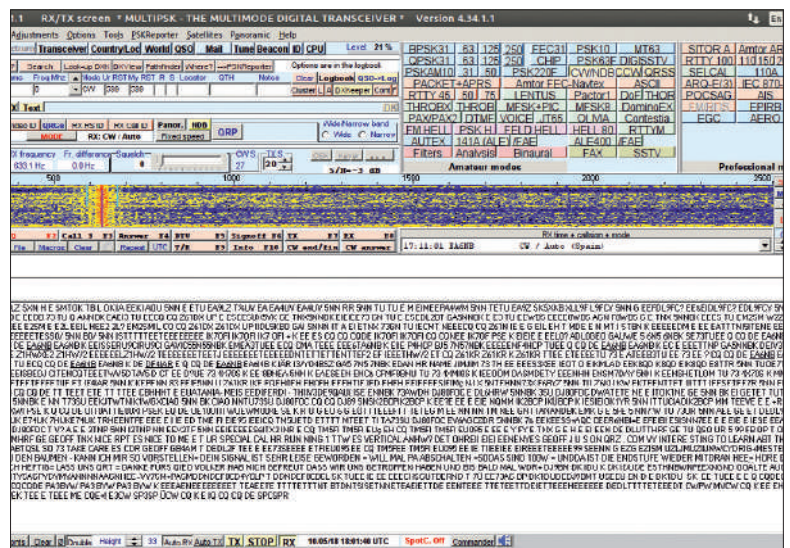
Учтите, что у *MultiPSK* два основных экрана — *Configuration* и *RX/TX*, и если экран *RX/TX* не отображается, *PulseAudio* не сможет его увидеть. Внося все необходимые изменения в страницу *Configuration*, перейдите на страницу *RX/TX*, и вы увидите, что здесь тоже имеется спектрограмма. Аудиосигнал, проходящий с Web-SDR, может содержать несколько сигналов морзянки, поэтому, нажав на спектрограмму *MultiPSK*, вы сможете точнее настроить то, что вам нужно. Теперь нажмите на желтую кнопку *CW/NDB*, и если не будет сильной интерференции, вы увидите соответствующий морзянке текст, который появится области внизу экрана. Сообщения азбукой Морзе могут быть весьма сложными, иногда не распознается пробел между словами, а шум может интерпретироваться как звук Е или Т. В подробном случае, на первый взгляд всё может выглядеть абсолютной чепухой, но загляните во врезку о жаргоне (стр. 72), и всё постепенно начнет проясняться.

В нижней области любительского диапазона, на частоте чуть выше азбуки Морзе, вы найдете другие сигналы данных, которые, в отличие от морзянки, созданы для автоматической, а не ручной передачи и получения. Чтобы декодировать эти сигналы, выбрав правильный режим из десятка кнопок *MultiPSK*, вы должны быть в состоянии распознать их.

Аудиосэмпы разных режимов вы найдете на www.sigidwiki.com/wiki/Category:Amateur_Radio, хотя на практике вы чаще всего встретите RTTY, PSK 31 и SSTV. SSTV особенно интересен, поскольку используется для отправки изображений, а не текста. Да, на получение одного изображения уходит около минуты, и оно может выглядеть не идеально, но не забывайте, что оно прибыло напрямую из какого-нибудь дальнего уголка Земли! SSTV обычно отправляется на нескольких фиксированных частотах в каждом диапазоне — стоит взглянуть на 3,733 МГц и 14,230 МГц, хотя полный список вы найдете на www.amateur-radio-wiki.net/index.php?title=SSTV_frequencies. Если вы хотите

найти станции, отправляющие изображения SSTV, хотя это никак не заменит вам самостоятельного декодирования изображений, загляните на www.worldsstv.com, где есть картинки, полученные радиолюбителями по всему миру и загруженные автоматически.

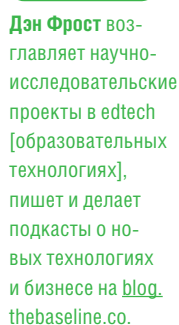
Поскольку это занятие затягивает, рано или поздно вам захочется пойти дальше, чем просто слушать и декодировать сигналы, которые получили другие и сделали доступными в Сети. В какой-то момент это будет означать приобретение дорогого коммуникационного приемника, но сегодня, благодаря SDR, вам не обойдется в целое состояние настройка собственной принимающей станции... но это уже другая история. **LXF**



MultiPSK — полнофункциональный пакет для декодирования любительских передач данных, хотя их интерпретация всё равно остается непостоянной.

» **ПОДПИШИТЕСЬ НА ЖУРНАЛ LINUX FORMAT** на www.linuxformat.ru/subscribe!

Дэн Фрост настраивает системные предупреждения, используя методы машинного обучения.



Подход в примерах нашего урока заключается в создании учебных и тестовых данных с использованием общедоступных URL из файлов BBC — **sitemap.xml**. Это обеспечивает «хорошие» URL, которые выглядят как нормальные пользователи сайта. Нам также нужны «плохие» URL, поэтому мы переделаем «хорошие» для создания «плохих» URL, которые выглядят как попытки

В данном примере мы будем употреблять только URL, а не более подробные данные, скажем, IP-адреса, дата, время и другие сведения; но если вы хотите создать более надежную и более пригодную к эксплуатации систему, соответствующие функции определенно имеет смысл добавить.

Создаем учебные данные

Один из подходов к созданию тренировочных данных — классифицировать их вручную, но на это понадобится время. Но не бойтесь — мы создали удобный скрипт, который будет вводить некоторые типичные нехорошие примеры поверх некоторых обычных URL. Вставьте следующие URL в файл с именем `url_classify.py`:

```
def gen_urls():
```

```
feeds = [
    'https://feeds.bbc.co.uk/news/uk/rss.xml',
    'http://feeds.bbc.co.uk/news/world/rss.xml',
    'http://feeds.bbc.co.uk/news/business/rss.xml',
    'http://feeds.bbc.co.uk/news/politics/rss.xml',
    'http://feeds.bbc.co.uk/news/technology/rss.xml',
    'http://feeds.bbc.co.uk/news/health/rss.xml',
    'http://feeds.bbc.co.uk/news/england/rss.xml',
    'http://feeds.bbc.co.uk/news/northern_ireland/rss.xml',
    'http://feeds.bbc.co.uk/news/scotland/rss.xml'
]

urls = []

for feed in feeds:
    d = feedparser.parse(feed)
    print("Feed loaded")
    for item in d['entries']:
        urls.append(item['links'][0]['url'])
    print("Loaded {} urls".format(len(urls)))
    return urls

def gen_hack_urls():
    bad_bits = [
        '?badbad?attr=1of98wef09e&username=wefwef',
        '..../etc/passwd',
        '../this/./does/not/look/great',
    ]

    urls = gen_urls()

    ret = []

    for u in urls[0:100]:
        ret.append(u + bad_bits[0])
        ret.append(u + bad_bits[1])
        ret.append(u + bad_bits[2])

    return ret
```

Можете поиграть с этим, запустив `python -i url_classify.py` в командной строке.

Готовим данные: токенизация URL

Чтобы применять машинное обучение словами, надо превратить слова в числовое представление. Методы для этого включают «мешок слов [bag of words]», который заменяет каждое отдельное слово на число, и погружения [embedding], которые стремятся найти некий подтекст в словах, чтобы, например, обнаружить общность в «кошки» и «собаки».

Однако многие библиотеки, превращая необработанные словесные данные в данные обучения, убирают знаки пунктуации. В нашем случае это создает проблему, хотя и интересную.

URL можно рассматривать как фразу, поскольку URL очень похож на описание местоположения. Например, <http://example.com/images/panda.jpg> читается как «на example.com находится изображение панды!». По мере усложнения URL символы становятся столь же значимыми, как слова, например, в <http://example.com/login?redirect=dashboard&account&sess=abc123>. Здесь знак вопроса, знак равенства и амперсанд имеют смысл; без них ряд других слов неоднозначен.

С точки зрения многих попыток взлома, которые исходят из скриптов, URL заполнены похожими значимыми символами. Нам надо найти способ превратить то, что могло бы быть выкинуто, в нечто полезное. Чтобы сделать это, мы просто поменяем каждый символ на какой-то примету [token]. Например,

```

..myenv) → ansoonly python -i final.py
>>> url1 = gen_urls(1)
Feed loaded
Feed loaded
Feed loaded
Feed loaded
Feed loaded
Feed loaded
Feed loaded
Feed loaded
Loaded 355 url1
>>> url1[10]
['https://www.bbc.co.uk/news/uk-england-merseyside-44696813', 'https://www.bbc.co.uk/news/science-environment-44612642', 'https://www.bbc.co.uk/news/uk-44780764', 'https://www.bbc.co.uk/news/entertainment-arts-44697212', 'https://www.bbc.co.uk/news/uk-england-london-44780819', 'https://www.bbc.co.uk/news/health-44793518', 'https://www.bbc.co.uk/news/uk-44644374', 'https://www.bbc.co.uk/news/uk-44695458', 'https://www.bbc.co.uk/news/uk-england-south-yorkshire-44698077', 'https://www.bbc.co.uk/news/uk-scotland-44698276']
>>> hack_url1 = gen_hack_urls(1)
Feed loaded
Feed loaded
Feed loaded
Feed loaded
Feed loaded
Feed loaded
Feed loaded
Feed loaded
Loaded 355 url1
>>> hack_url1[10]
['https://www.bbc.co.uk/news/uk-england-merseyside-44696813?badbad?attr=1of98wef090e&username=wefwef', 'https://www.bbc.co.uk/news/uk-england-merseyside-44696813.../this.../does/not/look/great', 'https://www.bbc.co.uk/news/science-environment-44612642?badbad?attr=1of98wef090e&username=wefwef', 'https://www.bbc.co.uk/news/science-environment-44612642.../this.../does/not/look/great', 'https://www.bbc.co.uk/news/uk-44780764?badbad?attr=1of98wef090e&username=wefwef', 'https://www.bbc.co.uk/news/uk-44780764.../this.../does/not/look/great', 'https://www.bbc.co.uk/news/entertainment-arts-44697212?badbad?attr=1of98wef090e&username=wefwef']

```

Наши два метода создания данных
дают нам немного начальной
тренировочной информации.

http://example.com/login?attr=123

станет

TKNHTTP example com TKNSLASH login TKNQM attr TKNEQ 123

Приметы-токены — слова, которые начинаются на “TKN” — теперь являются частью предложения и могут использоваться для обучения модели. Чтобы применить это к нашим данным, можно использовать следующий скрипт:

```
def tokenize(url):
    tokens = {
        'http://': ' ',
        'https://': ' ',
        '/': 'TKNSLASH',
        'www.bbc.co.uk': 'BBCCOUK',
        'example.com': 'EXAMPLECOM',
        '?': 'TKNQM',
        '=': 'TKNEQ',
        '..': 'TKNBDLDT',
        'uk-northern-ireland': 'uknorthernireland'
    }
    for original, replacement in tokens.items():
        url = url.replace(original, " {}".format(replacement))
    return url
```

Добавьте это в `url-classify.py` и запустите Python в интерактивном режиме с помощью `python -i url-classify.py`:

```
tokenize('https://example.com/this/is?a=url')
```

ЧТО ВАМ ВЫДАСТ

'EXAMPLECOM TKNSLASH this TKNSLASH is TKNQM a TKNEQ url'

» ЕЩЕ ОБ ИИ БЕЗОПАСНОСТИ

Конечно, увлекательно изобретать модели, чтобы делать конкретные и полезные вещи. И часто вы оказываетесь не первым, кто до этой идеи додумался — скажем, можно использовать сторонний сервис вроде AWS Guard Duty, или же рассматривать его как источник вдохновения.

AWS Guard Duty — это система машинного обучения для мониторинга трафика с целью выявления аномалий и реагирования на них. Сначала она тренируется, несколько недель занимаясь мониторингом вашего трафика и обращая внимание на подозрительный трафик. Вы сами решаете, как реагировать на него: можете инициировать выключение сервера или просто его заблокировать.

Стоит потратить немного времени, чтобы посмотреть, как работает Guard Duty и остальные, от фазы тренировки до решительных действий. Это может здорово вдохновить проекты машинного обучения и размышления о том, как реально поступать с суждением, которое выносит модель в каждом новом случае.

**СКОРАЯ
ПОМОЩЬ**

Получение данных для машинного обучения может быть огромной частью работы. Покопайтесь в приложениях и журналах сервера, отыскивая идеи, способные пригодиться. Большинство людей считают журналы нудным администрированием, тогда как это источник чудесной магии ML!

СКОРАЯ ПОМОЩЬ

Токенизация – всего лишь один пример подготовки данных. Большинство данных поступает неочищенными, так что учитесь это делать. Читайте изначальные данные и спросите: «Что я могу увидеть из того, что потенциально не попадет в модели?» Это часть «инжиниринга свойств».

Вы можете добавлять и другие приметы, например, квадратные скобки [] или кавычки, поскольку именно так люди взламывают базы данных, используя способ передачи информации в URL более сложными системами. Для наших целей мы на том и покончим с «токенизацией», но в своем приложении вы захотите извлечь как можно больше смысла, потому что алгоритмы машинного обучения полагаются на свойства (значения) исходных данных.

Затем, чтобы потренировать модель, надо смешать хорошие и плохие URL и создать список ярлыков. Это достигается следующим методом:

```
def gen_training_data():
    good_urls = gen_urls()
    bad_urls = gen_hack_urls()
    all_urls = []
    all_labels = []
    for u in good_urls:
        all_urls.append(u)
        all_labels.append(0)
    for u in bad_urls:
        all_urls.append(u)
        all_labels.append(1)
    return list(map(tokenize, all_urls)), all_labels
```

```
all_urls_tokenized, all_labels = gen_training_data()
```

Это прекрасно объединяет все данные и использует метод `tokenize` в конце метода `gen_training_data` method.

Векторизация

Машинное обучение работает с числами, а не словами, поэтому нам нужно каким-то образом трансформировать наши токенизированные URL. Это делается путем их векторизации, которая включает преобразование каждой строки слов в вектор, основанный на словах в этой строке. Если вы пока что не сталкивались с векторизацией, представьте себе двумерную диаграмму с «собакой» по оси X и «костью» по Y. Рассмотрим предложение «Моя собака предпочитает эту кость той кости» и отметим, сколько раз каждое слово появляется в предложении. Вы векторизовали часть предложения, чтобы представить ее в т.н. векторном пространстве. Чтобы векторизовать документы с сотнями или тысячами слов, мы делаем то же самое со многими измерениями, а не только с двумя, как в нашем простом примере с собакой и костью.

Векторизуйтесь

Для этого есть множество библиотек. Мы используем одну такую из *sklearn*, библиотеки машинного обучения Python. Если вы не установили *sklearn*, сделайте это с помощью `pip install sklearn` или `pip3 install sklearn`, в зависимости от своей версии Python.

Затем добавьте этап векторизации:

```
all_urls_tokenized, labels = gen_training_data()
vectorizer = CountVectorizer(analyzer = "word", \
    tokenizer = None, \
    preprocessor = None, \
    stop_words = None, \
    max_features = 5000)
vectorizer.fit_transform(all_urls_tokenized)
train_data_features = vectorizer.transform(all_urls_tokenized)
```

Если вы не знакомы с векторизацией, будет интересно взглянуть на функции, чтобы увидеть, как интерпретировалась наша токенизация, поскольку с этого момента здесь всё превращается в числа. Чтобы это увидеть, понадобится запустить Python в интерактивном режиме:

```
$ python -i url-classify.py
vectorizer.get_feature_names()
```

Вы можете исследовать векторизованные URL, как показано на рисунках на предыдущих страницах.

Тренируем модель

Подготовив данные и трансформировав их, мы готовы тренировать модель. Этот шаг включает самую сложную работу; однако для нас это довольно непрозрачно:

```
model = RandomForestClassifier()
model.fit(train_data_features, all_labels)
```

Здесь создается `RandomForestClassifier`, алгоритм машинного обучения, использующий набор случайных «лесов» для классификации существующих данных. С точки зрения внутреннего устройства, каждый тренировочный пример улучшает структуру случайного леса, уточняя отражение характера данных.

Мы разделяли данные, чтобы часть данных использовалась для обучения, а часть — чтобы проверить, хорошо ли работает только что обученная модель. Поскольку наш образец достаточно мал и не богат примерами, он работает не очень хорошо, но мы можем понять, как использовать его, с помощью:

```
bad_urls = gen_hack_urls()
bad_urls_tkn = list(map(tokenize, bad_urls))
bad_urls_tkn_vctr = vectorizer.transform(bad_urls_tkn)
p = model.predict(bad_urls_tkn_vctr)
```



OWASP.org — отличный источник вдохновения тем зрелищем, каким может выглядеть взлом.

» ЧТО ВЫ МОГЛИ БЫ СДЕЛАТЬ?

Пример нашего урока — немного игрушечный — вроде и работает, но не касается того, что еще с ним можно сделать или куда он может двигаться. Итак, посмотрим, чего еще мы можем достичь...

Важным улучшением было бы использование машинного обучения без учителя для выявления необычных записей журналов, а не того, что мы здесь делаем — обучение для распознавания плохих (необычных) записей. Это называется «обнаружение отклонений от нормы», а библиотеки вроде *sklearn* предоставляют некоторые алгоритмы для создания таких моделей. Преимущество состоит в том, что вам не надо просматривать и маркировать данные. Вместо этого вы готовите их (возможно, токенизируя, как мы поступили здесь), а затем обучаете модель распознавать «норму».

Вы также можете перенести идею в совершенно другой домен, такой как чат-боты. Если вы создали общедоступного чат-бота, который представляет информацию о праздниках или продуктах, нехорошие люди могут задуматься, не взломать ли его. А вы можете некоторое время отслеживать взаимодействие и создать тренировочные данные на основе нормального и аномального.

Вы также можете взглянуть на другие формы сетевого трафика и отслеживать необычные пакеты. Хотя тип данных будет иной, принцип весьма схож.

Other versions

Please cite us if you use the software.

1.11. Ensemble methods

1.11.1. Bagging meta-estimator

1.11.2. Forests of randomized trees

- 1.11.2.1. Random Forests
- 1.11.2.2. Extremely Randomized Trees
- 1.11.2.3. Parameters
- 1.11.2.4. Parallelization
- 1.11.2.5. Feature importance evaluation
- 1.11.2.6. Totally Random Trees Embedding

1.11.3. AdaBoost

- 1.11.3.1. Usage

1.11.4. Gradient Tree Boosting

- 1.11.4.1. Classification
- 1.11.4.2. Regression
- 1.11.4.3. Fitting additional weak learners
- 1.11.4.4. Controlling the tree size
- 1.11.4.5. Mathematical formulation
 - 1.11.4.5.1. Loss Functions
- 1.11.4.6. Regularization

algorithm in order to improve generalizability / robustness over a single estimator.

Two families of ensemble methods are usually distinguished:

- In **averaging methods**, the driving principle is to build several estimators independently and then to average their predictions. On average, the combined estimator is usually better than any of the single base estimator because its variance is reduced.

Examples: Bagging methods, Forests of randomized trees, ...

- By contrast, in **boosting methods**, base estimators are built sequentially and one tries to reduce the bias of the combined estimator. The motivation is to combine several weak models to produce a powerful ensemble.

Examples: AdaBoost, Gradient Tree Boosting, ...

1.11.1. Bagging meta-estimator

In ensemble algorithms, bagging methods form a class of algorithms which build several instances of a black-box estimator on random subsets of the original training set and then aggregate their individual predictions to form a final prediction. These methods are used as a way to reduce the variance of a base estimator (e.g., a decision tree), by introducing randomization into its construction procedure and then making an ensemble out of it. In many cases, bagging methods constitute a very simple way to improve with respect to a single model, without making it necessary to adapt the underlying base algorithm. As they provide a way to reduce overfitting, bagging methods work best with strong and complex models (e.g., fully developed decision trees), in contrast with boosting methods which usually work best with weak models (e.g., shallow decision trees).

Bagging methods come in many flavours but mostly differ from each other by the way they draw random subsets of the training set:

Как и многие библиотеки, *sklearn* содержит кучу ансамблевых методов. С ними легко планировать, как только у вас будут обучающие данные и стадии создания модели.

СКОРАЯ ПОМОЩЬ

Отсутствующий здесь, но важный шаг процесса – сохранение модели. Ищите 'model persistence' в *sklearn* (у других библиотек есть похожая функциональность), который эффективно сериализует модель для сохранения на диске или где-нибудь еще.

Здесь мы пытаемся предсказать, что url всегда плохие. Иногда модель работает, и это отлично. Но, как мы увидим далее, чтобы это работало для вашего контекста, надо настроить и улучшить данные.

На данный момент вы создали основные части системы машинного обучения, потенциально способной выявлять вредоносный трафик. Ее можно использовать для определения попыток взлома, блокировок доступа к службам или определения, когда система была взломана в ретроспективе.

Однако она работает не особенно хорошо, поскольку это искусственный пример, так что давайте посмотрим, как взять да улучшить эту базовую структуру.

Настройка модели

Наибольший пробел здесь — это доступные данные. У нас около 400 хороших URL-адресов и 300 очень похожих плохих URL, а это вовсе не много. Многие люди, запускающие сайты, сидят на куче журналов web-сервера, так что простое нахождение этих журналов, извлечение URL с помощью скрипта и его последующее использование — уже отличное начало.

Вы можете обнаружить попытки взлома, продираясь через журналы вручную или с помощью существующих инструментов с открытым кодом для их обнаружения. Затем надо разделить журналы на два списка: хорошие URL и плохие URL. Решив пойти дальше, вы можете попытаться взломать свой собственный сайт или попробовать службы тестирования на вторжение для создания попыток взлома.

Если вы используете программное обеспечение с открытым кодом, можно объединиться с другими людьми, использующими ту же программу, для слияния всех обучающих данных.

URL — это всего лишь одна функция журнала web-сервера, и из необработанных данных журнала вы можете добыть больше сведений. Например, IP-адреса, время суток (что важно, если взломы связаны с часовыми поясами) и количество запросов в минуту могут открыть вам новые горизонты. Вы можете добавить свойство к каждому ряду, который представляет собой количество запросов, поступивших с одного и того же IP-адреса за последние несколько секунд или минут. В какой-то момент всплеск этого числа может указать на злонамеренную активность, что по запрашиваемому URL не столь очевидно.

Собрав данные в большем количестве и лучшего качества, вы также сможете поэкспериментировать с гиперпараметрами модели обучения. Например, изменение `max_features` может изменить точность. Тестирование по нескольким значениям может выявить компромисс точности/скорости.

`RandomForestClassifier(max_features=100)`

Гиперпараметр `max_depth` определяет глубину дерева, что способно оказать огромное влияние на точность. Хорошей практикой, хоть это и выходит за рамки нашей статьи, будет построить гиперпараметр на оси X и точность на оси Y, чтобы найти оптимальное значение для каждого.

Применяйте знания на практике

Самое важное в машинном обучении — это использовать то, что вы создали, а не копить искусственные примеры. Можно просто каждую ночь запускать скрипт на вашем сервере и сразу обнаруживать, что работает, а что нет. Со временем вы отточите свойства и параметры своей модели, так что в итоге выйдет нечто полезное. **LXF**

Amazon GuardDuty is a managed threat detection service that continuously monitors for malicious or unauthorized behavior to help you protect your AWS accounts and workloads. It monitors for activity such as unusual API calls or potentially unauthorized deployments that indicate a possible account compromise. GuardDuty also detects potentially compromised instances or reconnaissance by attackers.

Enabled with a few clicks in the AWS Management Console, Amazon GuardDuty can immediately begin analyzing billions of events across your AWS accounts for signs of risk. GuardDuty identifies suspected attackers through integrated threat intelligence feeds and uses machine learning to detect anomalies in account and workload activity. When a potential threat is detected, the service delivers a detailed security alert to the

Guard Duty применяет машинное обучение для выявления необычного трафика. Такие инструменты – отличный источник вдохновения для ваших собственных проектов.

» ПОДПИШИТЕСЬ НА ЖУРНАЛ **LINUX FORMAT** на www.linuxformat.ru/subscribe!

АДАПТИВНЫЕ СРЕДЫ

Создаем сайт для всех устройств

Если вы хотите создать отзывчивые web-страницы для крупных игроков, вам повезло: под рукой — **Кент Ельчук**, готовый осветить основы.



НАШ
ЭКСПЕРТ

Кент Ельчук — штатный web-разработчик и энтузиаст Linux, который в свободное время занимается программированием и выращиванием гидропонных культур.

СКОРАЯ ПОМОЩЬ

Используя язык стороны сервера, такой как PHP, мы можем сделать наше меню в отдельном файле и добавить его на любую страницу, используя простую строку кода: `<?php include("header.php");?>`.

Адаптивные среды стали стандартом для современного web-дизайна. Но когда их ввели, многие эксперты не были уверены в их ценности и рассматривали их как всего лишь сноску на пути развития Интернета. Эксперты не всегда бывают правы...

Хронологически примерно в 2010 г. web-разработчики по всему миру поняли, что эти среды CSS, позволяющие создать один сайт для всех устройств просмотра, штука невероятная. Большинство сайтов тогда кодировалось с фиксированной шириной, благодаря чему упрощалось их создание. С точки зрения типового web-дизайна, адаптивный код был совсем другим и требовал сменить образ мышления.

Ныне классы и файлы сред стали для web-дизайнеров и разработчиков второй натурой. Итак, сядем поудобнее и впитаем данный урок, чтобы мы могли ускорить работу с помощью адаптивной среды CSS.

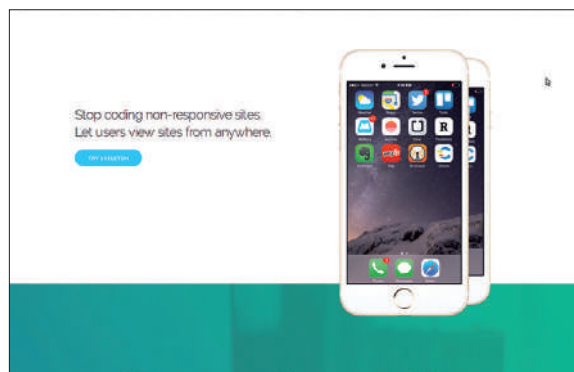
При использовании адаптивной среды у нас есть много вариантов, и все они делают более или менее одно и то же. Поскольку это руководство не может рассмотреть все различные среды, мы рассмотрим три наиболее популярных варианта: Bootstrap, Skeleton и Foundation. Среди них Bootstrap заметнее всех, и мы начнем с него, потому что он поставляется со многими загружаемыми образцами и применяется во многих бесплатных и коммерческих шаблонах.

Тянем за Bootstrap

На протяжении многих лет Bootstrap прошел через несколько основных версий, и мы рассмотрим текущую: Bootstrap 4. Когда мы используем нашу адаптивную среду, у нас есть различные отправные точки на выбор: готовый пример, коммерческий шаблон, или начать с нуля.

Если мы выберем построение с нуля, то у нас есть возможность либо использовать настоящие файлы на нашем web-сервере, либо добыть среду с Content Delivery Network [Сеть доставки контента] (CDN). Код для использования CDN показан ниже, его можно вставить в заголовок файла:

```
<!-- Latest compiled and minified CSS -->
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.1.0/css/bootstrap.min.css">
```



У адаптивной среды Skeleton очень красивая начальная тема. Код минимален, и его легко редактировать и поддерживать.

```
<!-- jQuery library --> <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
<!-- Popper JS -->
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.0/umd/popper.min.js"></script>
<!-- Latest compiled JavaScript -->
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.1.0/js/bootstrap.min.js"></script>
```

В нашем случае мы для начала создадим нечто простое из примера на сайте Bootstrap. Исходный код мы можем загрузить с <https://bit.ly/2eVyaGU>. После загрузки извлечем ZIP-файл и перенесем его в любое место, включая любую папку, наш локальный хост или web-сервер. Например, в локальном компьютере Linux с сервером Apache путь будет `/var/www/html`.

Извлеченная папка будет называться **bootstrap-4.0.0**. Эта папка содержит целый набор подпапок. Итак, приступим к делу и откроем пример, который хотим исследовать, в нашем случае — Jumbotron. Чтобы впервые рассмотреть пример с Jumbotron на нашем локальном компьютере, мы можем зайти на <http://localhost/responsive/bootstrap-4.0.0/docs/4.0/examples/jumbotron>.

Файл для примера Jumbotron, который находится в папке Jumbotron, называется **index.html**. Итак, давайте откроем этот файл. Первым делом проверим код между тегами

`<head>` / `</head>`, указывающий на файлы Bootstrap, которые влияют на стиль страницы.

Используемые файлы Bootstrap находятся в заголовке в папке **bootstrap-4.0.0/dist**. Файл в этой папке называется попросту **bootstrap.min.css**. Страница использует файл под названием **favicon.ico**, расположенный в папке **bootstrap-4.0.0**. Это маленький значок, отображаемый при загрузке страницы в браузере — 'В', помечающее вкладку браузера.

В дополнение к файлам заголовков, мы можем прокрутить до нижней части файла **index.html** и посмотреть на нижнюю часть, которая содержит различные теги скриптов. Первый скрипт JQuery загружается с code.jquery.com, а остальные находятся в разных папках нашей установки Bootstrap.

Теперь, зная файлы, определяющие стиль нашей страницы, мы можем просмотреть страницу сверху донизу, чтобы понять, почему она выглядит и работает именно так. Мы будем ссылаться на HTML-код в **index.html** и изучать классы, теги и другие атрибуты, которые существуют в этих файлах Bootstrap.

Вниз по кроличьей норе

Поскольку ранее мы уже изучали требуемые для Bootstrap файлы, мы не будем снова их ворошить. Но для напоминания в заголовке и нижней части **index.html** присутствуют ссылки на файлы и скрипты, чтобы заставить нашу адаптивную страницу выполнять свою задачу — «быть отзывчивой». Это, в свою очередь, создаст элегантность и единообразие на всех цифровых устройствах.

Теперь поподробнее рассмотрим то, что находится между тегом `<body>` и закрывающим тегом `</body>`. Первый тег, который мы видим после тега `<body>` — открывающий тег `<nav>`. Набор элементов между открывающим и закрывающим тегами `<nav>` / `</nav>` — это меню, которое мы видим в верхней части страницы с темным, черноватым фоном. Легкий способ разобраться в этом — посмотреть на страницу в браузере и сопоставить ее с необработанным кодом.

Начнем с действующего тега **nav**, который имеет пять классов: **navbar**, **navbar-expand-md**, **navbar-dark**, **fixed-top** и **bg-dark**. Эти классы влияют на стиль. Например, класс **bg-dark** устанавливает цвет фона для меню.

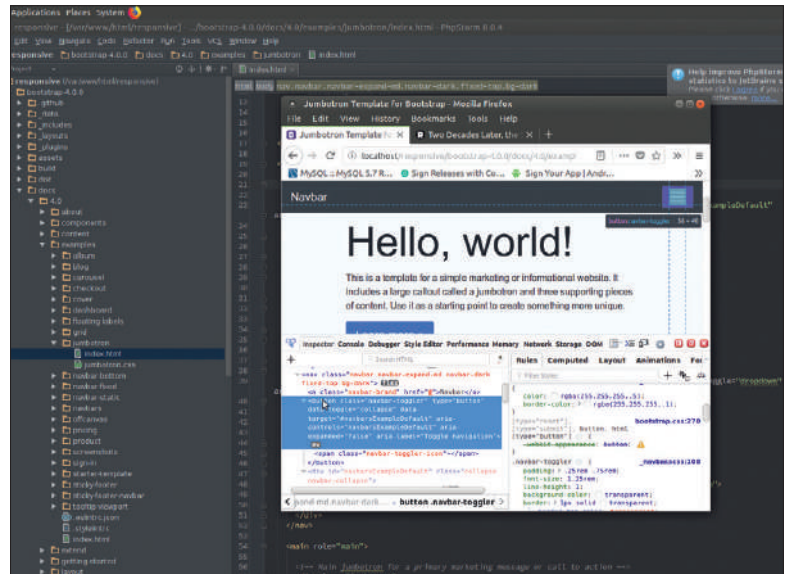
Под классом **nav** у нас есть ссылка, которая находится между тегами `<a>` / ``. У ссылки есть прикрепленный к ней класс, под названием **navbar-brand**, который отвечает за многие факторы, такие как цвет текста, отображение встроенного блока, отступы, поля и размер шрифта — это только некоторые из них.

Для тех, кто не знаком с CSS — существует множество свойств и значений, которые могут быть присвоены любому классу. Свойство — это скажем, размер шрифта, а соответствующим значением может быть 12 пикселей. Свойства и значения разделяются двоеточием и отображаются как **font-size: 12px**.

Идем дальше; следующий кусок кода — это кнопка, которая тоже имеет связанные с ней начальный и конечный теги. И вот что интересно в этой кнопке. На больших устройствах, таких как настольные компьютеры и ноутбуки, кнопка не отображается. Но когда размер экрана уменьшается или сайт загружается на более мелкие устройства, кнопка появляется.

Зато, когда появляется кнопка, исчезают элементы меню, потому что кнопка будет отображать элементы меню при нажатии. Благодаря этому, на маленьком экране навигация становится намного более дружелюбна и предсказуема.

Чтобы сохранить следующий набор элементов более кратким, элементы в первом `` / ``, которыми являются элементы `<i>` / `</i>`, отображают текст и характеристики выравнивания верхнего меню. Фактические ссылки — это Home, Link, Disabled и Dropdown.



Вот что вы видите при изменении размера окна браузера при открытом инспекторе. Теперь мы можем преспокойно настроить изменения для всех экранов.

Несмотря на то, что всё это теги ``, тот, который выделяется — это последний, который является выпадающим меню.

При нажатии на слово Dropdown отображаются подпункты.

Пока все ссылки на меню — это просто заглушки '#'. Если мы хотим сделать для нашего сайта больше страниц, мы можем изменить ссылки, чтобы при щелчке по элементу меню отображались другие страницы.

Таким образом, приведенный ниже код показывает ссылку, которая по существу остается на той же странице, тогда как у той ссылки, что находится ниже, другое имя, и она ведет на другую страницу.

```
<a class="nav-link" href="#">Link</a><a class="nav-link" href="new-page.html">About</a>
```

Перейдя к нижней части элемента **nav**, мы можем увидеть форму, содержащую окно ввода и кнопку Search [Поиск]. Об этих

» СКРИПТ JQUERY ДЛЯ СТОЛБЦОВ

По умолчанию адаптивные среды обычно отображают столбец, основываясь на высоте элементов внутри столбца. Следовательно, один столбец может составлять 100 пикселей, а другой — 300 пикселей.

Поскольку иногда вам нужно, чтобы оба столбца были определенной высоты, скрипт JQuery ниже позволит нам сделать это. Всё, что нам надо сделать, это добавить еще один класс в наши столбцы и назвать его 'inner-top'.

```
$( document ).ready(function() {
    width = $(window).width();
    if (width > 768) {
        var all_heights = $(".inner-top").map(function () {
            return $(this).height();
        }).get();
        maxHeight = Math.max.apply(null, all_heights);
        $(".inner-top").height(maxHeight);
    }
});
```

Вкратце, этот скрипт получает размер окна. Если он больше 768 пикселей, выполняется остальная часть кода. Следующие строки сопоставляют высоту всех столбцов с классом **inner-top** и задают каждому из них на высоту самого большого столбца. Этот скрипт совместим с текущей и предыдущими версиями среды.



Это элементарно

Этот первый набор элементов — довольно простой старый HTML, который охватывает всю ширину страницы. Он уменьшает ширину при уменьшении размера окна браузера, но выглядит одинаково на всех устройствах.

Текст, лежащий внутри класса 'Jumbotron', содержит второй набор тегов `div`, в котором содержатся наши большие теги `<h1>` `</h1>` — это крупные слова на странице, которые гласят: "Hello, world!"

Следующий набор текста находится с элементом `<p>` `</p>`, а после этого другой элемент `<p>` содержит ссылку на атрибут `role` [роль]. Атрибут `role` имеет значение, называемое `button` [кнопка], что объясняет, почему он выглядит как кнопка (достаточно забавно!).

Следует знать, что теги `<p>` будут охватывать страницу браузера. Таким образом, если мы добавим дополнительные элементы `<p>` `</p>` в существующие, они появятся в новой позиции, когда мы обновим нашу страницу.

Разобравшись с элементом `div` и классом `jumbotron`, перейдем к новому элементу `div`, который имеет класс под названием `container` [контейнер]. Прямо под открывающим тегом есть еще один элемент `div` с классом `row` [ряд]. Этот класс `row` уже давно является неотъемлемым атрибутом адаптивной среды. В нем размещается схема для реагирования.

Чтобы сделать всё простым и понятным, посмотрим на наш код и увидим, что у нас есть три экземпляра `<div class="col-md-4"></div>`. По сути, `col-md-4` — это размер столбца. Этот показатель составляет 4/12, т.е. треть от размера страницы. В Bootstrap ряд `[row]` может содержать в столбцах любые количества, в сумме составляя 12. Таким образом, `col-md-12` будет всего лишь одним рядом, а шесть классов `col-md-2` в ряду будут шестью небольшими столбцами. Опять же, с нашими тремя замечательными классами `col-md-4` у нас есть три одинаковых ряда. Внутри каждого из этих столбцов есть заголовки `<h2>` `</h2>`, абзац замещающего текста и кнопка, которая вновь не приведет вас никуда, кроме как на саму страницу.

Теперь мы можем прогнать эту страницу через тесты размера, уменьшая окно нашего браузера. Как мы видим, когда окно доходит до определенного размера, столбцы наслаиваются один поверх другого. Т.е. сверху будет левый столбец, под ним — средний столбец и, наконец, тот, что справа. Кроме этого, у нас осталось не слишком много достойного упоминания HTML, за исключением текста, расположенного между тегами в нижней части страницы.

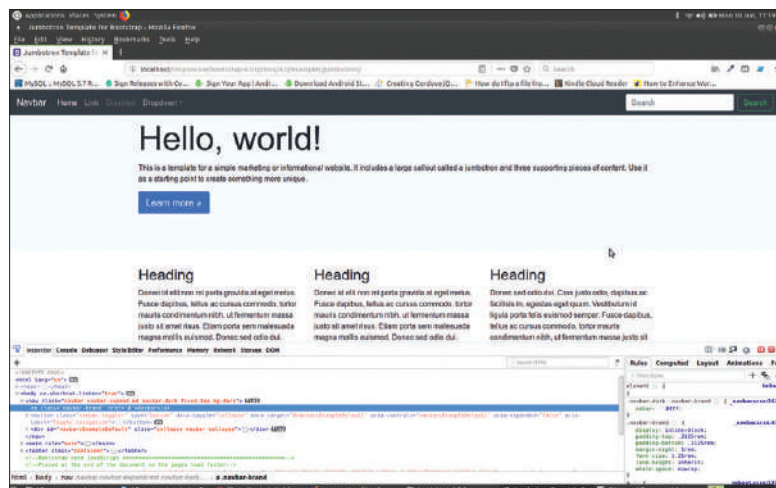
Вот примерно и весь краткий курс молодого бойца по Bootstrap, который доминирует в мире адаптивного дизайна HTML, CSS и JS. Итак, давайте взглянем на некоторые другие популярные варианты гибких сред и посмотрим, что у них общего.

Скелеты в CSS

Проект Skeleton.css основан на файлах из <https://bit.ly/2tj5wm6>. Это ссылка с сайта [getskeleton.com](https://github.com/get-skeleton-css), и эти файлы используются на нашем уроке. Как и Bootstrap, среда Skeleton использует свой набор файлов, чтобы сделать страницу отзывчивой. Эти файлы — по преимуществу `normalize.css` и `skeleton.css`.

Загрузив наш файл `index.html` из кода, содержащегося в данном руководстве, мы сможем искать все классы `row`. Да, Skeleton использует один и тот же класс `row` для хранения столбцов, которые накладываются, когда окно браузера уменьшается до определенного размера. Однако Skeleton называет столбцы иначе, хотя принцип тот же, и сетка позволяет использовать те же 12 столбцов. В качестве примера рассмотрим приведенный ниже код — последует подробное объяснение:

```
<div class="row">
  <div class="one-half column">
```



Вот пример Jumbotron из Bootstrap. Это простой набор файлов, упрощающий создание страницы современного вида, которая работает на всех устройствах.

тегах следует отметить, что входной тег является самозакрывающимся, в отличие от многих других тегов, которые имеют начало и конец тега. Кроме того, существует множество других самозакрывающихся тегов, с которыми мы, без сомнения, столкнемся при кодировании HTML, например, такие, как разрывы строк `
`, горизонтальные линии `<hr />` и `<meta />`.

Чтобы сделать нашу форму реально полезной, следует написать код JavaScript или JQuery, который задавал бы действие при нажатии. На нашем уроке мы не станем этого рассматривать, поскольку это достаточно нетривиальное занятие, и само по себе способно легко послужить темой для целого отдельного учебника!

Следовательно, мы можем просто вырезать его или скрыть. Чтобы скрыть его, мы можем изменить код с помощью приведенной ниже строки и использовать для свойства `display` значения `none`. `!important` удостоверяет, что данное решение — окончательное:

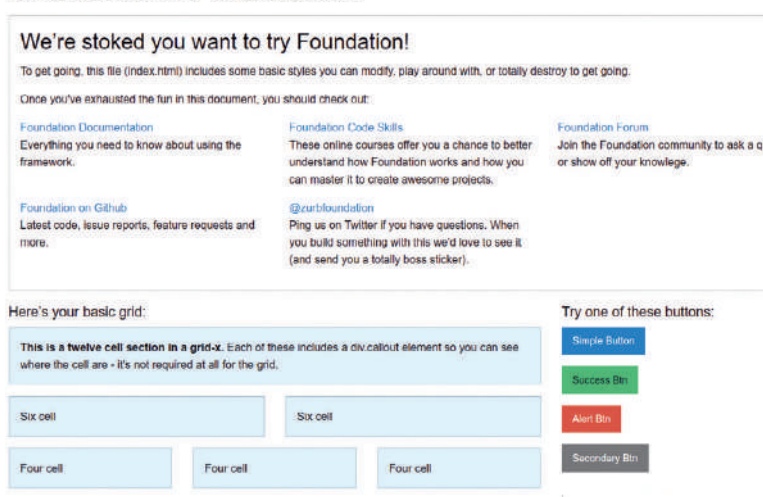
```
<form class="form-inline my-2 my-lg-0" style="display: none !important">
```

Это завершает навигацию, и теперь мы можем продолжить движение вниз по странице страницы и смотреть на элементы между тегами `<main>` `</main>`, начиная с кода между тегами `<div class="jumbotron">` `</div>`.

СКОРАЯ ПОМОЩЬ

Многие продавцы коммерческих тем замутняют код, переименовывая классы среды и создавая новые классы. Мы всегда должны проверять код, прежде чем использовать тему, чтобы сэкономить время на компиляцию.

Welcome to Foundation



Тема Foundation по умолчанию отлично справляется с созданием визуальной темы, которая превращает изучение адаптивного дизайна в простой процесс.

```

<h4 class="hero-heading">Перестанем использовать
неадаптивные сайты! Пусть пользователи просматривают
сайты с любого устройства.</h4>
<a class="button button-primary" href="http://getskeleton.
com">Try Skeleton</a>
</div>
<div class="one-half column phones">
  
  
</div>
</div>

```

Этот код разделяет ряды на два равных столбца, дважды используя класс `one-half column`. Следующий ряд имеет три равных столбца, которые используют значение класса `one-third column`. Это достаточно просто. Но у Skeleton есть несколько дополнительных опций по части присвоения имен столбцам.

Например, если мы используем один столбец, класс будет `one column`, тогда как два и более столбца используют окончания множественного числа: `two columns`. Вдобавок Skeleton использует дроби, которые мы уже обсуждали: например, `one-third column` [треть столбца], `one-half column` [половина столбца] и `two-thirds column` [две третьих столбца].

Foundation

Последнюю версию Foundation всегда можно скачать с <https://bit.ly/2t5VjdH>. Опять же, когда мы смотрим на код, особенно на заголовок и последние несколько строк над закрывающим тегом `</body>` нашего основного `index.html`, мы можем видеть, что используем несколько файлов, которые делают за нас всю разметку.

В отличие от предыдущих двух сред, мы не будем использовать класс `row`. Вместо него используется класс под названием `grid-x`. Внутри сетки, которая снова суммарно составляет до 12, мы используем имена классов наподобие `large-8` и `medium-8`, чтобы указать ширину каждого столбца для указанного устройства:

```

<div class="grid-x grid-padding-x">
  <div class="large-8 medium-6 cell">
    <div class="primary callout">
      <p>Six cell</p>
    </div>
  </div>
  <div class="large-4 medium-6 cell">
    <div class="primary callout">
      <p>Six cell</p>
    </div>
  </div>
</div>

```

Хорошо, подытожим: у нас есть сетка и структура столбца. Видим мы нечто отличное от других примеров? А должны бы. Действительно, мы видим, что в большом окне браузера на настольном ПК и ноутбуке первый столбец будет шириной восемь, а второй столбец будет только четыре.

Однако при уменьшении окна до среднего размера столбцы будут подстраиваться так, чтобы каждый из них стал шесть в ширину, сделавшись одинаковыми по размеру. В конце концов, по мере дальнейшего уменьшения мы увидим обычный процесс перекрытия столбцов друг над другом. В некоторых случаях мы можем захотеть оставить столбцы, даже для самых маленьких устройств вроде смартфона. Для этого мы могли бы использовать такие классы, как `small-6` для двух столбцов в одной сетке.

» УПРОЩЕННЫЕ ПРОВЕРКИ БРАУЗЕРА

Проверка элемента дает нам отличный доступ, чтобы внести небольшие изменения в код и просмотреть изменения в реальном времени. И новички, и опытные web-разработчики используют эти инструменты регулярно. Без использования этого инструмента трудно представить разработку и тонкую настройку проекта.

Щелкнув правой кнопкой мыши по странице в нашем браузере и выбрав `Inspect with Chrome` [Осмотреть с Chrome] или `Inspect Element with Firefox` [Осмотреть элемент с помощью Firefox], мы увидим в нижней части страницы загрузку инспектора. В левом верхнем углу инспектора есть стрелка с левым указателем.

Выбрав стрелку влево, мы можем перетаскивать указатель над страницей, и он будет выделять элементы. Мы можем заставить его выделять все, например, крупный заголовок текста, ряд, столбец или другую комбинацию нескольких элементов.

Как только мы наводим на страницу и делаем выбор, мы можем изменить любой из HTML в поле ниже. Мы можем делать что угодно — менять текст, добавлять и удалять классы, и т.д. Это очень похоже на редактирование реальных файлов, разве что изменения являются временными.

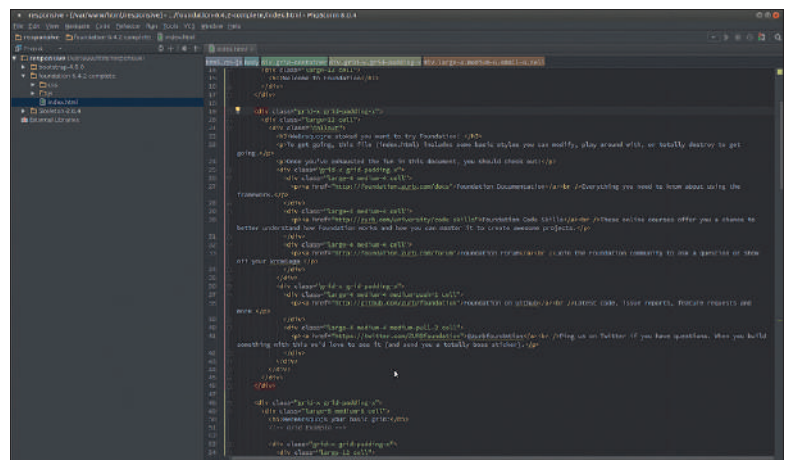
Помимо изменения HTML, мы можем вносить изменения в код CSS, который показан в маленьком поле справа от инспектора. Можно удалять свойства и изменять их значения — например, изменить размер шрифта или сделать фоновое меню немного светлее.

Заключение

Итак, теперь мы познакомились с адаптивным дизайном и можем редактировать базовые темы и темы по умолчанию для адаптивных сред или даже покупать разработанные шаблоны и настраивать их под свои нужды, даже обладая лишь ограниченными навыками кодирования.

Ключ к простоте нашей работы состоит в том, чтобы делать мелкие, незначительные изменения в существующих темах сред, без особых амбиций.

Поскольку все адаптивные среды представляют собой простой набор файлов, которые работают в браузере, не нуждаясь в сервере, мы всегда можем загружать файлы с сайтов создателей и открывать файл `index.html` в любом браузере из любой папки нашей Linux-машины. **LXF**



Хороший редактор кода очень упрощает нахождение открывающих и закрывающих тегов для всех элементов и умеет форматировать текст, улучшая его читаемость.

» ПОДПИШИТЕСЬ НА ЖУРНАЛ **LINUX FORMAT** на www.linuxformat.ru/subscribe!

SWIFT 4

Быстрый тур по новейшим функциям

Михалис Цукалос раскапывает всё вам необходимое, чтобы взяться за последние функции нового релиза Swift на своем компьютере.



**НАШ
ЭКСПЕРТ**

Михалис Цукалос — человек UNIX и автор *Go Systems Programming* и *Mastering Go*. Вы можете связаться с ним через mtsoukalos.eu и mactsouk.com.

СКОРАЯ ПОМОЩЬ

Сайт языка программирования Swift — <https://swift.org/>. Вы можете найти инструкции и информацию о доступных версиях Swift 4 на <https://swift.org/download/>. Узнайте больше о Swift 4, прочитав его документацию на <https://swift.org/documentation/>.

История языка программирования Swift коротка — впервые он был представлен в 2014 г.; но в ближайшем будущем займёт ключевую роль. На данный момент Swift меняется очень быстро. На нашем уроке мы расскажем о функциях последней версии Swift, анонсированной на WWDC 2017, и попытаемся рассмотреть самые важные и полезные новые функции Swift 4 на примере небольших, но автономных программ. Чтобы извлечь из данного урока максимум, вам понадобится прочитать и понять его содержимое и попытаться применить полученные знания в своих собственных программах Swift 4.

Поскольку Swift 4 не поставляется в стандартном пакете ни одного дистрибутива Linux, вам понадобится установить его вручную — мы объясним это вкратце. Пожалуйста, учтите, что, возможно, вам понадобится установить на ваш компьютер Linux пакет *clang*, если у вас его еще нет. Итак, в системе Linux Ubuntu 18.04 вам понадобится выполнить следующие действия:

```
$ wget https://swift.org/builds/swift-4.1.2-release/ubuntu1604/swift-4.1.2-RELEASE/swift-4.1.2-RELEASEubuntu16.04.tar.gz
$ tar zxvf swift-4.1.2-RELEASE-ubuntu16.04.tar.gz
```

В зависимости от места установки Swift 4 вам следует изменить значение переменной среды **PATH**, добавив директорию Swift 4. В нашем случае, поскольку установка Swift 4 оказалась в **/home/mtsouk/code**, новое значение должно быть таким:

```
$ export PATH=/home/mtsouk/code/
swift-4.1.2-RELEASE-ubuntu16.04/usr/bin:"${PATH}"
```

Тогда вы сможете использовать компилятор и инструменты Swift 4 из любого места в вашей системе Linux.

Вы можете узнать точную версию используемого вами Swift, выполнив `swift --version`. В данном случае мы будем использовать Swift 4 версии 4.1.2. Следующий код Swift — сохраненный как **simple.swift** — убедится, что ваша установка Swift 4 завершена успешно и что вы можете компилировать и выполнять код Swift 4 без каких-либо проблем:

```
print("Hello, world!")
```

Компиляция проста и заключается в запуске `swift build`. После этого можно запустить получившийся исполняемый файл с тем же именем, что у исходного файла. Вы можете выполнить **hw.swift** и без предварительной компиляции, запустив `swift hw.swift`. При попытке запустить исполняемый файл вы, возможно, получите сообщение об ошибке, наподобие следующего:

```
$ ./hw
```



Рис. 1. Домашняя страница на <https://swift.org> — лучшее место для поиска информации и разработок в языке программирования Swift.

```
./hw: error while loading shared libraries: libcuc.so.55: cannot
open shared object file: No such file or directory [Нет такого
файла или директории]
```

Это означает, что на вашей системе Debian или Ubuntu Linux вам надо вручную установить *libcuc* версии 55, запустив следующие две команды:

```
$ wget http://security.ubuntu.com/ubuntu/pool/main/i/icu/
libcuc55_55.1-7_amd64.deb
$ sudo dpkg -i libcuc55_55.1-7_amd64.deb
```

Оказывается, установка Swift 4 на компьютер Linux может быть запутанной!

Swift 4 vs. Swift 3

Будет полезно обсудить некоторые различия между Swift 4 и Swift 3, чтобы лучше понять, что произошло между этими двумя версиями. Обратите внимание, что представленные отличия, скорее всего, пригодятся постоянным разработчикам. В Swift 4 есть много изменений и улучшений, которые большинство из нас даже не заметят и применять не будут. Прежде чем вдаваться в подробности, вы должны знать, что Swift 4 не ломает ваш существующий код Swift 3, потому что Swift 4 пытается улучшить Swift 3, а не испортить.

Во-первых, **Strings** [Строки] в Swift 4 теперь стали **Collections**, а значит, вы можете выполнять итерацию по переменной **String** с большей легкостью — такой пример, а также различные другие применения объекта **String** вы увидите в следующем разделе. Во-вторых, Swift 4 включает улучшения в диапазонах,

последовательностях и сопоставлении шаблонов, которые приводят к более читаемому коду. Затем вы увидите улучшения в типах данных Dictionary [Словарь] и Set [Набор]. И наконец, вы увидите более естественный способ определения многострочных строковых литералов. Для полного списка изменений в Swift 4 посетите <https://swift.org>, его также можно увидеть на рис. 1.

Перебор строк

Чтобы начать, давайте узнаем больше о объектах String, которые улучшены в Swift 4. Во-первых, вы узнаете, как итерировать объект String, потому что теперь строки называются 'Collections [Коллекции]':

```
let magazine = "Linux Format Magazine"
for ch in magazine {
    print(ch, terminator: " ")
}
```

Вдобавок, строковые переменные также являются последовательностями, а значит, вы можете применять к строковым переменным функции как последовательностей, так и коллекций! Затем вы узнаете, как определить строковые литералы с несколькими строками, вставив текст в три кавычки:

```
let lxf = "LFX"
let myText = """
Linux Format (a.k.a. \lxf)
is the Best Linux
Magazine
"""
```

Пример выше также показывает, как вы можете поместить значения существующих переменных в строковые литералы.

И наконец, символы Юникода [Unicode] теперь возвращают '1', когда даются в качестве ввода метода подсчета. Ранее символы Юникода не считались одним символом, что может потенциально запутать ваш код.

Весь код этого раздела можно найти в `useString.swift`. Выполнение исходного файла `useString.swift` создаст следующий вывод:

```
LinuxFormatMagazine
Linux Format (a.k.a. LFX)
is the Best Linux
Magazine
52
1
1
3
```

Диапазоны и вне их!

Swift 4 поддерживает умные диапазоны, что позволяет вам делать вашу работу без необходимости использовать 'endIndex' и 'startIndex'. Следующий код показывает это:

```
let days4567 = days[4...]
let days123 = days[..<4]
```

В более старых версиях Swift вам бы пришлось писать `days[4..days.endIndex]` и `days[days.startIndex..4]` соответственно.

Вдобавок Swift 4 поддерживает бесконечные последовательности — при условии, что тип такой последовательности является исчисляемым. Вы также можете звать их «ленивыми [lazy] бесконечными последовательностями» по причинам, которые станут очевидными позже. Бесконечная последовательность определяется следующим образом:

```
let rs = (1...)
```

Однако это еще не всё, поскольку для ее использования надо задекларировать несколько условий. Хорошая новость в том, что эта последовательность будет расти в соответствии с вашими потребностями. Бесконечные последовательности можно использовать в качестве случаев в сопоставлении шаблонов — об этом позже. Весь код этого раздела Swift 4 можно найти в `sequences`.

```
1 // Part 1
2 var days = ["Monday", "Tuesday", "Wednesday",
3             "Thursday", "Friday", "Saturday", "Sunday"]
4
5 // Before Swift 4: days[4..days.endIndex]
6 let days4567 = days[4...]
7 // Before Swift 4: days[days.startIndex..4]
8 let days123 = days[..<4]
9
10 print(days4567)
11 print(days123)
12
13 var numDays = Array(zip(1..., days))
14 print(numDays)
15
16 // Part 2
```

`swift`, который можно увидеть на рис. 2 (вверху). Итак, что же происходит в `sequences.swift`?

Первая часть определяет массив с именем 'days', используемый для создания массивов `days4567` и `days123`. Как мы упоминали ранее, для создания этих массивов вам не нужны `days.endIndex` или `days.startIndex`.

Вторая часть показывает использование ленивых бесконечных последовательностей. Заметьте, что цикл `while` создаст тот же результат, что и первая бесконечная последовательность, то есть поиск первой третьей степени целочисленной переменной, которая больше 10000. Затем ленивая последовательность `rs` используется для ввода в массив первых десяти целочисленных переменных, превышающих 10000, возведенных в куб. Такой же код, но с циклом `while` выглядел бы менее элегантно. Причина, по которой `rs` называется «ленивой», в том, что в переменной `rs` ничто не вычисляется и не вводится до тех пор, пока вы не создадите `Array()` [Массив] и не сохраните его как `r10`.

Третья часть показывает использование умных диапазонов (`case ...2`) и бесконечных последовательностей (`case 8...`) в сопоставлении образов.

Выполнение `sequences.swift` создаст следующий вывод:

```
["Friday", "Saturday", "Sunday"]
["Monday", "Tuesday", "Wednesday", "Thursday"]
[(1, "Monday"), (2, "Tuesday"), (3, "Wednesday"), (4, "Thursday"),
(5, "Friday"), (6, "Saturday"), (7, "Sunday")] 10648
10648
```

Рис. 2. Код Swift 4 для `sequences`. `swift`, показывающий умные ряды, ленивые бесконечные последовательности и использование бесконечных последовательностей в сопоставлении образов.

СКОРАЯ ПОМОЩЬ

Форумы Swift (<http://bit.ly/swiftfor>) — лучшее место, чтобы узнать больше о Swift 4, включая разработку и реализацию Swift (<http://bit.ly/swiftdev>), а также его эволюцию (<http://bit.ly/swftevo>). Форумы Swift даже удобны для новичков.

» SWIFT 4 REPL

Все версии Swift идут с REPL (Read-Eval-Print-Loop), который очень удобен для того, чтобы узнавать и пробовать новое, поскольку вам не надо создавать исходный файл, компилировать и выполнять его — всё оценивается после нажатия клавиши Enter, а значит, ошибки будут выявлены с максимальной быстротой.

Чтобы войти в Swift 4 REPL, надо просто выполнить `swift`, что выдаст вам следующее:

```
$ swift
Welcome to Swift version 4.1.2 (swift-4.1.2-RELEASE). Type :help for assistance.
1>
```

Заметьте, что Swift REPL поддерживает автозаполнение, т.е. если вы что-то введете и нажмете клавишу Tab, REPL выдаст все доступные окончания. Это может оказаться очень полезным, когда вы не помните имя переменной или функции, которую хотите выполнить.

Рис. 5 (см. стр. 85) показывает несколько примеров REPL, включая вычисления, операции на строковых переменных и функцию автозаполнения. Последнее свойство Swift REPL состоит в том, что он также является очень способным отладчиком, и это — помимо прочего — означает, что вы можете создавать контрольные точки кода Swift 4, которые вы поместите в REPL и в случае критических ошибок немедленно восстановитесь без необходимости закрывать Swift REPL и переходить к другому инструменту.

СКОРАЯ ПОМОЩЬ

Хотя Swift 4 – фантастический язык программирования, это, возможно, не лучший выбор, если вы хотите написать инструмент командной строки Linux. И если вас интересует системное программирование на языке, отличном от C или Python, возможно, вы захотите взглянуть на Go (<https://golang.org>) или Rust (rust-lang.org/en-US/).

```
[10648, 12167, 13824, 15625, 17576, 19683, 21952, 24389, 27000, 29791]
Too early for weekend
Enjoy your weekend!
Wrong value!
```

Возможно, вам не каждый день понадобятся диапазоны и бесконечные последовательности, но уж если вы возьметесь их применять, они определенно облегчат вашу жизнь.

Типы словарей

Теперь давайте перейдем к типу данных Dictionary, который будет показан с помощью простой программы. Тип данных Dictionary используется для группировки элементов, являющихся парами ключ–значение.

Хотя тип данных Dictionary существует со времен более старых версий Swift, Swift 4 добавил в него несколько практических функций, сильно облегчив работу разработчика. Вся программа Swift 4, иллюстрирующая тип данных Dictionary, сохранена как `dictSet.swift` и показана на рис. 3.

Оператор `struct Day: Hashable` и код, следующий за ним, создает новую структуру с именем `Day`, которая выводится из типа данных 'Hashable'.

Переменные `Mon`, `Tue`, `Wed`, `Thu`, `Fri`, `Sat`, `Sun` [Пн, Вт, Ср, Чт, Пт, Сб, Вс] относятся к типу данных 'Day', и каждая из них представляет собой день недели. Каждый из этих дней имеет тип, как задано в перечне 'DayType'. Используя все эти семь переменных, мы создадим новую переменную Dictionary, которую мы назовем 'weekDays'.

Затем оператор `let groupBy = Dictionary(grouping: weekDays, by: {item in item.dayType})` позволяет вам создать новый словарь, содержащий те же значения (семь дней недели), сгруппированные по `DayType` (Тип дня), употребляемому как ключ. Он распределяет дни недели в три группы, как задано тремя значениями перечня 'DayType', и позволяет вам легко узнать день каждого набора.

Вывод `dict.swift` выглядит следующим образом:

```
$ swift dict.swift
[dict.Day(name: "Monday", dayType: dict.Day.DayType.working),
dict.Day(name: "Tuesday", dayType: dict.Day.DayType.resting), dict.
```

```
1 struct Day: Hashable {
2     var name: String
3     var dayType: DayType
4
5     enum DayType {
6         case working, holiday, resting
7     }
8
9     var hashValue: Int {
10         return name.hashValue << 2 | dayType.hashValue
11     }
12 }
13
14 let Mon = Day(name: "Monday", dayType: .working)
15 let Tue = Day(name: "Tuesday", dayType: .resting)
16 let Wed = Day(name: "Wednesday", dayType: .working)
17 let Thu = Day(name: "Thursday", dayType: .working)
18 let Fri = Day(name: "Friday", dayType: .working)
19 let Sat = Day(name: "Saturday", dayType: .holiday)
20 let Sun = Day(name: "Sunday", dayType: .holiday)
21
22 // This is a Dictionary
23 let weekDays = [Mon, Tue, Wed, Thu, Fri, Sat, Sun]
24 print(weekDays)
25
26 // Now you can group the values by DayType
27 let groupBy = Dictionary(grouping: weekDays,
28                          by: {item in item.dayType})
29
30 print(groupBy[.resting]!)
31
```

Рис. 3. Показан код Swift из `dict.swift`, демонстрирующий, как Swift 4 с легкостью группирует элементы переменной Dictionary по ключу.

```
Day(name: "Wednesday", dayType: dict.Day.DayType.working), dict.
Day(name: "Thursday", dayType: dict.Day.DayType.working), dict.
Day(name: "Friday", dayType: dict.Day.DayType.working), dict.
Day(name: "Saturday", dayType: dict.Day.DayType.holiday), dict.
Day(name: "Sunday", dayType: dict.Day.DayType.holiday)] [dict.
Day(name: "Tuesday", dayType: dict.Day.DayType.resting)]
```

О типе данных Dictionary можно узнать больше на <https://apple.co/2vIDE2k>.

Ключи и значения

В этом разделе вы узнаете о том, как конвертировать пары ключ–значение в словари, что поможет вам сэкономить массу энергии. Логику конвертирования можно найти в следующем коде:

```
let zipped = Dictionary(uniqueKeysWithValues: zip(days, r))
Переменная r — это последовательность, а переменная days
является массивом; функция zip() использует days и r для созда-
ния нового словаря под названием zipped. Кроме того, следую-
щий код позволяет вам справиться с инициализацией словаря-
Dictionary, который содержит дубликаты ключей, что встречается
не так уж редко:
let duplicates = ["Monday", "Tuesday", "Friday", "Wednesday",
"Thursday", "Friday", "Saturday", "Sunday", "Monday"]
let dup = Dictionary(zip(duplicates, repeatElement(1, count:
duplicates.count)), uniquingKeysWith: +)
```

Функция `zip()` вместе с `uniquingKeysWith: +` добавляет конфликтующие значения вместо создания двух отдельных пар ключ–значение.

Выполнение `keysValues.swift` создаст следующий вывод:

```
$ swift keysValues.swift
["Tuesday": 2, "Saturday": 6, "Sunday": 7, "Friday": 5, "Monday":
1, "Wednesday": 3, "Thursday": 4] ["Tuesday": 1, "Saturday": 1,
"Sunday": 1, "Friday": 2, "Monday": 2, "Wednesday": 1, "Thursday": 1]
```

Не бойтесь экспериментировать с кодом Swift 4 `keysValues.swift` и создавать свои собственные примеры.

Функциональное программирование

Теперь давайте рассмотрим применение Swift 4 в качестве языка функционального программирования. Во-первых, посмотрим, как присвоить функцию переменной:

```
func add(a: Int, b: Int) -> Int {
    return a + b
}
var funVar: ((Int, Int) -> Int) = add
print(funVar(1,2))
```

В данном случае переменная `funVar` содержит функцию `add()`. Кроме того, у вас может быть функция, которая возвращает другую функцию и присваивает эту функцию переменной следующим образом:

```
func retFunc() -> (Int) -> (Int) {
    func double(a: Int) -> Int {
        return a+a
    }
    return double
}
var doubleMe = retFunc()
print(doubleMe(12))
```

`retFunc()` возвращает функцию с именем `double()`, которая на вход принимает 'Int' и возвращает другое значение 'Int'. По сути, функция 'Int -> Int' удваивает свой вход. Возвращаемое значение `retFunc()` присваивается переменной, которая затем используется как функция внутри оператора `print()`!

К сожалению, ограниченный объем нашего учебника не позволяет нам рассказать больше о возможностях функционального программирования Swift 4.


```

1 protocol Computable {
2     static func +(lhs: Self, rhs: Self) -> Self
3     static func -(lhs: Self, rhs: Self) -> Self
4 }
5
6 func add<T: Computable>(_ num1: T, _ num2: T) -> T {
7     return num1 + num2
8 }
9
10 func sub<T: Computable>(_ num1: T, _ num2: T) -> T {
11     return num1 - num2
12 }
13
14 extension Int: Computable {}
15 extension Float: Computable {}
16 extension Double: Computable {}
17
18 print(add(3, 0.2), add(3, 2.2), sub(1.2, 2), sub(-3, 2))
19
20
21 func mySwap<T>(_ a: T, _ b: T) -> (T, T) {
22     let tA = a
23     let tB = b
24     return (tB, tA)
25 }

```

Рис. 4. **gen.swift** демонстрирует Generics на двух примерах, один из которых использует протокол `Computable`, а другому протокол не нужен.

Generics

Если вернуться к руководству **LXF225** о Swift 3, то там говорили о Generics, не представляя вам никакого настоящего кода Swift, который на самом деле использует Generics. На этом уроке мы поговорим о Generics более подробно, поскольку это очень полезная функция Swift. На самом деле вы, скорее всего, уже использовали Generics в ваших программах Swift, не зная этого, ведь массивы, словари и дополнительные функции — всё это типы Generics!

Итак, Generics позволяет вам разрабатывать типы и функции, способные работать с любыми типами, даже не существующими на момент написания вашего кода Generics!

Представьте, что у вас есть функция Swift, которая складывает два целых числа — хотя функция идеальна, она умеет работать только с целыми числами, а значит, вам придется создавать другую функцию для сложения чисел двойной длины, и, возможно, еще одну для сложения чисел с плавающей запятой; и т.д. Представьте, насколько сложнее было бы поддерживать все эти типы, если бы вас попросили создать функцию, которая переставляет два значения, или функцию, которая сортирует три значения одного и того же типа данных.

Файл **gen.swift** представляет пример полного кода, который демонстрирует использование Generics в Swift 4 на основе только что описанного сценария: сложение двух числовых величин. Весь код Swift **gen.swift** можно увидеть на рис. 4, и его следует пояснить.

Вначале вы создаете новый протокол Swift, который будет включать операторы `+` и `-` для типов данных, которые уже поддерживают эти два оператора. Затем вы вводите две настраиваемые функции с именами `add()` и `sub()`, которые используют `+` и `-` соответственно. Обе могут работать со значениями, соответствующими протоколу `Computable` [Вычисляемый].

Хитрость здесь заключается в том, что оператор `+` уже был введен в Swift для значений `Int`, `Float` и `Double`, так что вам не придется вводить что-то еще — нужно просто использовать в функции `add()` оператор `+`. То же относится к оператору `-` и функции `sub()`. Если ни одна из ваших настраиваемых функций не была основана на существующем операторе, то вам не понадобится создавать какие-либо протоколы Swift, такие как протокол `Computable`, найденный в **gen.swift** — пожалуйста, взгляните на реализацию в **gen.swift** функции `mySwap()`, которой не нужен протокол для работы с подобным примером.

»» OPTIONALS

Тип `Optionals` [опционалы] — это очень интересная функция Swift 4, позволяющая работать в отсутствие значения. Возможные значения переменной `optionals` — `None` и `Some(T)`, где `T` — это значение верного типа данных, относящегося к переменной; это означает, что возможные значения для типа `optional` `Int` — `None` или значение `Int`. Когда переменная определяется как опционал (`op`), вам понадобится ее развернуть, чтобы использовать. Развертывание происходит с помощью восклицательного знака.

Следующее взаимодействие с REPL многое прояснит насчет Optionals:

```

> var optInt: Int?
> optInt = 100
> print(optInt!)

```

Вначале вы задаете опционал для `Int`; значение переменной `optInt` на данном этапе — `nil`. Затем вы присваиваете ей значение; но попытавшись напечатать его, в качестве вывода вы получите `Optional(100)`, поскольку вам надо вначале развернуть этот опционал. Итак, чтобы получить актуальное целочисленное значение из целочисленного опционала, вам понадобится использовать восклицательный знак, как показано в третьем операторе.

Optionals могут спасти вас от мерзких ошибок и многих неприятных ситуаций, особенно при работе с неизвестными значениями или вводе информации пользователем, так что используйте их — ведь они дают вам возможность проверить, равно значение переменной `nil` или нет, и действовать соответственно. Это означает, что такие операторы, как `if optInt == nil` и `if optInt! == nil` являются корректным кодом Swift 4.

И наконец, расширьте типы данных `Int`, `Float` и `Double`, сообщив им, что они соответствуют протоколу `Computable`. После этого вы будете готовы использовать функции `add()` и `sub()` со значениями `Int`, `Float` и `Double`. Функцию `mySwap()` можно использовать с любыми типами данных.

Теперь то, что выполнение **gen.swift** напечатает на вашем экране `"3.2 5.2 -0.8 -5", (2, 1)` и `("Num 2", "Num 1")`, определенно имеет смысл.

Если вы хотите поэкспериментировать и узнать больше о программировании, Swift 4 будет хорошим выбором — дерзайте! **LXF**

```

6> let name = "Linux Format"
name: String = "Linux Format"
7> print(name)
Linux Format
8> name.count
Available completions:
compactMap(transform: (Character) throws -> ElementOfResult?(Character) throws -> ElementOfResult?) -> [ElementOfResult]
contains(element: Character) -> Bool
contains(where: (Character) throws -> Bool(Character) throws -> Bool) -> Bool
count: Int
8> name.count
$R3: Int = 12
9> 4 + 3
$R4: Int = 7
10> 3 + 2.1
$R5: Double = 5.0999999999999996
11> var days = ["Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday"]
12. let r = (1...7)
days: [String] = 7 values {
[0] = "Monday"
[1] = "Tuesday"
[2] = "Wednesday"
[3] = "Thursday"
[4] = "Friday"
[5] = "Saturday"
[6] = "Sunday"
}

```

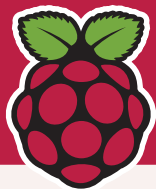
Рис. 5. Показана среда Swift REPL. Вы можете видеть в действии вычисления, операции со строковыми переменными и функцию автозаполнения.

»» ПОДПИШИТЕСЬ НА ЖУРНАЛ **LINUX FORMAT** на www.linuxformat.ru/subscribe/

7 советов быстрого поиска работы от hh.ru

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

И помните, что работа найдется для каждого!



Томас Стратфорд — специалист по информатике в The Misbourne School из Грейт-Миссенден, графство Бакингемшир [Great Missenden, Buckinghamshire].

» ЗАРАЖЕННЫЙ МАЛИНКОЙ

Когда я не исправляю ИТ-проблемы, мне нравится создавать, расковыривать, заниматься ретро-вычислениями и проектами в области электроники. Я активный участник сообщества разработчиков, меня часто можно встретить на различных Raspberry Pi Jam, и меня знают как ведущего спонтанных учебных курсов.

Мне нравится создавать проекты, основанные на Raspberry Pi и micro:bit и показывать людям, что можно сделать с помощью этих устройств.

Впервые я узнал о Raspberry Pi в 2012 г., и за прошедшие годы приобретал все последние модели Raspberry Pi сразу после выпуска. Пожалуй, смело можно утверждать, что я заразился малиновой лихорадкой!

Как только Raspberry Pi у меня появился, я задумался о том, что с ним можно сделать. Начал читать журналы, покупать всякие коммутационные платы и сияющие светодиоды на базе плат расширения, а также посещал разные сборища типа Raspberry Pi Jam, с целью узнать, что с Raspberry Pi делают другие люди.

Переходим к 2015 г., когда BBC начала выдавать micro:bit каждому семикласснику Великобритании. Я снова подумал, что должен иметь такую платформу, поэтому присутствовал на тренировочной сессии, а затем с нетерпением ждал их прибытия. С тех пор я и подсел.

Я обнаружил, что сообщество Raspberry Pi — дружелюбная группа людей, фантастический источник вдохновения, и они охотно делятся идеями. По-моему, детям действительно нравится этот волшебный момент, когда вот они написали код для управления светодиодом, а тот загорелся. Я очень увлечен точными науками — STEM, и полагаю, что мы должны делать всё возможное для вдохновения умельцев и кодеров следующего поколения.

Официальный мини-марсоход NASA Curiosity

Теперь можно познакомиться с Марсом с помощью собственной домашней тележки JPL на базе Pi.

Вы ведь слышали о NASA, верно? Это отважная группа инженеров и аэрокосмических энтузиастов, которые отправили людей на Луну (и обратно); наряду с этим, у них еще много роботов, рассекающих по Марсу. В помощь воплощению в жизнь своих проектов по марсоходам на Земле, Лаборатория



реактивных двигателей [JPL] при NASA создала мини-марсоход под названием ROV-E, для охвата школ и других общественных мероприятий.

В ответ на запрос энтузиастов инженеры JPL разработали модель марсохода, которая может быть собрана инженерами следующего поколения из промышленных, готовых к использованию (COTS) частей дешевле £2000. Естественно, на базе Raspberry Pi.

Всё, что вам нужно для создания собственного марсохода, находится на GitHub. Open Source Rover от JPL — это открытое оборудование, «сделай сам», уменьшенная версия марсохода, которую JPL использует для изучения Марса. Проект предназначен для обучения на опыте тех, кто хочет участвовать в машиностроении, ПО, электронике или робототехнике. Узнайте больше на <https://github.com/nasa-jpl/open-source-rover>.

Создайте собственный мини-марсоход NASA!

Кто сказал Му?

Появилась новая IDE.

Ориентированная на новичков среда Python IDE выпустила версию 1! *Mu* описывается как не единственная IDE Python вашей жизни, но первая. Это редактор, который поможет вам начать свое путешествие по кодированию, но не обязательно тот, с которым вы его завершите. Нам нравится, как это звучит, а копию можно взять на <https://codewith.mu>.

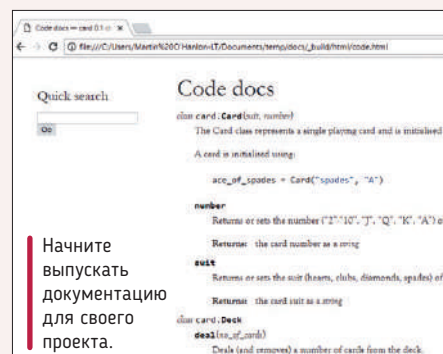


Простой значок для простой среды IDE.

Документация

Вот как это делается!

Документация, отсутствие которой все оплакивают, но никто ее не делает, получила подмогу от Raspberry Pi Foundation... с некоторой документацией, ага! Если вы работаете над проектом и задаетесь вопросом, как его документировать, вот вам отличное место для начала: <https://projects.raspberrypi.org/en/projects/documenting-your-code>.



Начните выпускать документацию для своего проекта.

Etcher 1.4.4

У **Леса Паундера** куча Raspberry Pi, а значит, ему надо записывать множество SD-карт. Поможет ли ему это приложение сэкономить время?

ВКРАТЦЕ

Инструмент GUI для создания SD-карт для Raspberry Pi и загрузаемых дисков USB для любителей менять дистрибутивы. *Etcher* имеет встроенные функции безопасности для минимизации риска потери данных. В этой новой версии мы увидели такие функции, как запись на несколько SD-карт и на внешние устройства, такие как Raspberry Pi Compute Module и Pi Zero.

У всех Raspberry Pi есть одна общая черта, и это — SD-карта, используемая в качестве загрузочного устройства и локального хранилища. Запись SD-карт — особенно нескольких — это проклятие пользователей, запускающих события Raspberry Jam, поскольку на нее уходит масса времени. Конечно, можно использовать `dd` или `dcfldd` для записи нескольких карт, однако одна опечатка на клавиатуре практически означает катастрофу.

Шаг вперед, *Etcher* — приложение GUI, предлагающее при создании карт гибкость и безопасную работу. Предназначенный для употребления пользователями с разными способностями, *Etcher* является приложением Electron, это один исполняемый файл, скачиваемый и запускаемый без необходимости в установке. *Etcher* доступен для Linux, OS X и Windows, и работает совершенно одинаково на всех этих платформах.

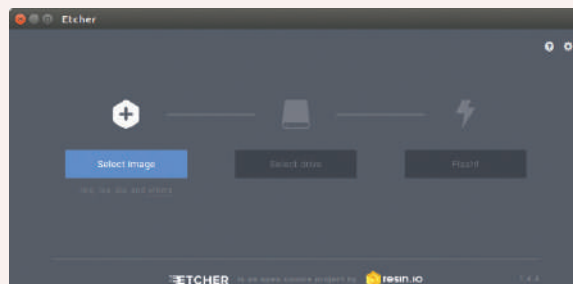
В своей простейшей форме *Etcher* может записать образ — будь то образ SD-карты или ISO для применения с загружаемым диском USB — на ваше целевое устройство. И в этой простой задаче *Etcher* превосходит благодаря простому трехшаговому интерфейсу. Выберите источник образа, выберите целевую SD или устройство USB, затем нажмите Flash! Скорость записи зависит от вашего компьютера, но на среднем ПК 8-ГБ образ отнимет минут восемь. *Etcher* также способен определять, была ли запись успешна, оценивая устройство после записи образа, что снижает вероятность испорченной SD-карты.

Работа с несколькими картами

Для пользователей, которым надо быстро создать несколько карт, *Etcher 1.4.3* предлагает опцию записи на несколько устройств. Разработчики заявляют о стабильности для устройств в количестве до 16 штук, улучшенную в этой последней версии. Мы протестировали ее с двумя карт-ридерами USB 2.0, и обнаружили, что образы скопировались чисто, правда, немного медленнее, чем одна карта. Но мы использовали устройство Core i5 второго поколения; более быстрые CPU и USB 3.0 работают с несколькими картами эффективнее.

Еще одна новая функция — возможность записывать образы прямо на реальные устройства — например, Raspberry Pi Compute Module и Pi Zero. Но это очень сложно, поскольку по умолчанию *Etcher* таких устройств не видит. Для получения доступа к Compute Module через USB нам пришлось следовать инструкциям в блоге

Вот вам *Etcher* в действии, определяющий два диска перед началом записи данных на SD-карту.



У *Etcher* простой интерфейс, совершенно одинаковый для многих операционных систем. Идеально для новых пользователей, желающих записать SD-карты для своих Raspberry Pi.

Raspberry Pi, чтобы клонировать репозиторий Git и скомпилировать драйвер. Сделав это, мы смогли смонтировать Compute Module и с помощью *Etcher* записать на внутренний eMMC. Как упоминалось, *Etcher* умеет делать запись и на платы Pi Zero, но их тоже надо смонтировать тем же методом, что и для Compute Module, и карта microSD в них должна быть совершенно пустой. Мы также смогли сделать запись на несколько плат Pi Zero одновременно.

Если игнорировать причуды, *Etcher* — солидный инструмент для создания SD-карт и загружаемых USB-дисков для смены дистрибутивов. Встроенные функции безопасности снижают риск случайного удаления дисков посредством скрытия более объемистых томов — опытные пользователи могут это отключить.

Новая возможность записи прямо на Compute Module или Pi Zero превосходна, но включить и использовать ее довольно непросто. Мы смогли заставить ее работать только с помощью ранее описанной хитрости. Более того, чтобы записать несколько устройств, в Linux нам пришлось запустить *Etcher* от имени root или с помощью `sudo` (пользователям Windows не придется для этого повышать свои привилегии), но команда *Etcher* признала, что это исправят в будущих релизах. И для тех, кому приходится создавать много SD-карт, *Etcher* — необходимый инструмент, чтобы выполнить эту работу хорошо и с минимальными затратами. Мастера Jam, ваши молитвы были услышаны! **LXF**

ВЕРДИКТ

РАЗРАБОТЧИК: Resin.io

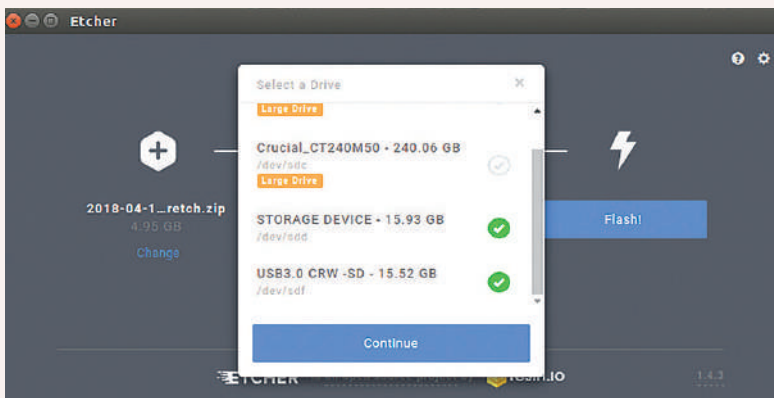
САЙТ: <https://etcher.io>

ЛИЦЕНЗИЯ: Apache V2.0

ФУНКЦИОНАЛЬНОСТЬ	9/10
ПРОИЗВОДИТЕЛЬНОСТЬ	9/10
УДОБСТВО В РАБОТЕ	9/10
ДОКУМЕНТАЦИЯ	8/10

Простой в использовании, но со множеством расширенных функций, *Etcher* — мощная кросс-платформенная альтернатива командной строке.

» **Рейтинг 8/10**



Pi-hole

Лес Паундер любит изучать лучшее в Интернете, но отнюдь не рекламу, которая норовит ему что-то продать. Не послать ли всю рекламу в черную дыру?

ВКРАТЦЕ

Простое приложение, которое позволит избежать рекламных объявлений и повысит производительность вашей сети. Оно предлагает защиту от рекламы и вредоносных программ на основе рекламы независимо от того, какую операционную систему или устройство вы используете, включая планшеты и телефоны.

Интернет полон спама и рекламы, и обычно мы устанавливаем на наших устройствах блокировщики рекламы, чтобы коммерческие компании не портили нам просмотр. Установка таких инструментов требует чувствительных затрат — иногда финансовых, но уж обязательно системных ресурсов. Как же нам заблокировать рекламу? Давайте отправим ее в *Pi-hole*.

Pi-hole является «черной дырой для интернет-рекламы» и действует как DNS-сервер, блокирующий рекламу из доменов, которые попались на рассылке спама. Как вы, возможно, догадались по имени, *Pi-hole* предназначен для использования с Raspberry Pi, причем любой модели Pi, если она подключена к вашему маршрутизатору. В наших тестах мы взяли Raspberry Pi 3 и выполнили простой однострочный скрипт установки, который использует *curl* для загрузки скрипта и запускается в *Bash*, и хотя это не идеальный способ установки программ, поскольку может способствовать вредоносной установке, он с первого раза заработал без проблем. И команда *Pi-hole* предлагает пользователям возможность перед установкой изучить исходный код.

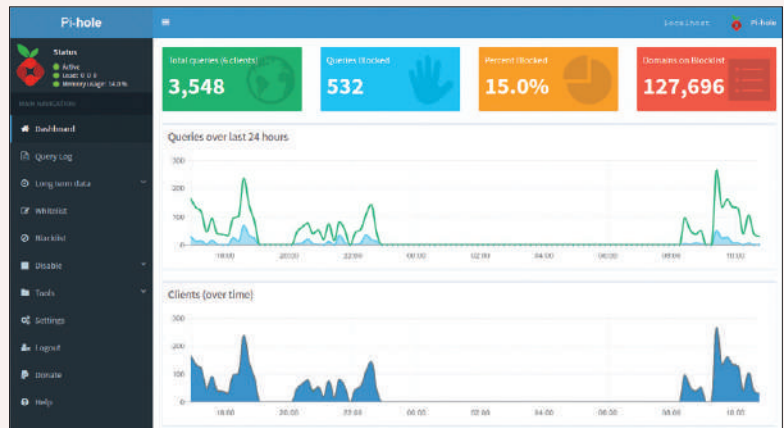
Объявления, вон!

После установки мы вошли на наш ноутбук с Ubuntu и изменили настройки DNS, чтобы он использовал DNS-сервер *Pi-hole*. После быстрого перезапуска сетевой службы наш ноутбук протестировал несколько «набитых рекламой сайтов», и внезапно все они исчезли, причем без всякого программного блокировщика рекламы на стороне клиента удалось обойти обнаружение блокировки.

Поскольку в нашей домашней сети работает аппаратное DNS-устройство, всё, что нам надо сделать для перехода на DNS-сервер *Pi-hole*, это изменить сетевые настройки используемого устройства. Так что пользователи Windows и OS X могут тоже применять данную услугу, как и владельцы устройств Android и Apple!

Использование *Pi-hole* вместо блокировщика рекламы на стороне клиента имеет недостаток. Как только мы сменим маршрутизатор — скажем, подключимся в кафе или дома у друга — реклама вернется! Тогда надо снова обратиться к блокировщику рекламы на стороне клиента или к DNS-серверу *Pi-hole*, используя Pi с переадресацией портов в нашей домашней сети, или настроить *Pi-hole* на виртуальном сервере в облаке. Мы протестировали это с помощью сервера Linode за \$5, и он работал нормально.

Полезным побочным эффектом удаления рекламы является то, что *Pi-hole* заодно увеличивает производительность сети,



Pi-hole прост в установке даже для тех, кто ни разу не сетевой администратор. Но для этих последних *Pi-hole* достаёт массу козырей из своих виртуальных рукавов.

поскольку полоса пропускания не забита изображениями и треками. *Pi-hole* также поставляется с интерфейсом администратора, в который можно войти по IP-адресу вашего устройства через браузер. После этого вы увидите основную страницу администратора с графиками и основными данными. Но войдя в систему с паролем, заданным во время установки, вы увидите гораздо больше! А именно, графики запросов и клиентов, пытающихся всучить нам объявления, и информацию о тех, кто это делает; и можно вручную добавлять сайты/домены в белый список (разрешенные сайты) и черный список (запрещенные сайты), просто введя их. Страница администратора также позволяет выполнять аудиты — удобно для сетевых администраторов, которые хотят проверить, откуда могут возникать угрозы или проблемы.

Интернет-реклама — это нудно. *Pi-hole* дает нам управляемый инструмент для уменьшения ее влияния на наш просмотр. Как базовое решение для Raspberry Pi дома это отлично, но вне дома не забывайте защищать себя. Понабравшись Linux-фу настроить сервер с *Pi-Hole* в облаке провайдера — раз плюнуть, и вы забудете про рекламу на всех своих мобильных устройствах. **LXF**

ВЕРДИКТ

РАЗРАБОТЧИК: Pi-Hole

САЙТ: <https://pi-hole.net>

ЛИЦЕНЗИЯ: Европейский Союз PL 1.2

ФУНКЦИОНАЛЬНОСТЬ	9/10
ПРОИЗВОДИТЕЛЬНОСТЬ	10/10
УДОБСТВО В РАБОТЕ	10/10
ДОКУМЕНТАЦИЯ	9/10

Достаточно простой для обычных пользователей, но достаточно мощный для сетевых админов. Надежный инструмент для сообщества.

» **Рейтинг 9/10**

Pi-hole нацелен на пользователей Pi, но может быть установлен во множестве дистрибутивов Linux. Ubuntu, Fedora, CentOS — все годятся.

Pi-hole работает на многих дистрибутивах Linux



BASH

Легко и просто управляем GPIO

Лес Паундер демонстрирует три примера управления GPIO Raspberry Pi плюс некоторую магию *Bash* — притом без волшебной палочки.



**НАШ
ЭКСПЕРТ**

Лес Паундер помогает развитию навыков разрабатчиков. Его блог идей и проектов — www.bigl.es.

ВАМ НУЖНЫ

- > Любый Pi
- > Raspbian
- > Макетная плата
- > Светодиод
- > Кнопка
- > Резистор на 220 Ом
- > Контроллер L298N для мотора
- > Коробка на 4 батарейки AA
- > 7 контактных перемычек
- > мама-папа
- > 1 перемычка мама-мама
- > Клавиатура, мышь, монитор, Интернет
- > Код получите с github.com/leesp/LXF240/archive/master.zip

Есть бесчисленное множество проектов для Raspberry Pi, и большинство из них написано на Python. Но знаете ли вы, что скромный *Bash*, тот самый интерфейс, который мы используем для управления нашей системой, также пригоден для управления GPIO? А всё благодаря проекту WiringPi Гордона Хендерсона [Gordon Henderson]. На этом уроке мы покажем, как управлять GPIO утилитой *gpio*. Всё необходимое программное обеспечение у вас предустановлено, поэтому включите Pi и откройте терминал!

Проект 1: Мигание светодиода

Аналог программы “Hello world” при работе с аппаратурой — это мигание светодиода. Начнем со сборки схемы для этого проекта, которая состоит из светодиода [LED] на макетной плате, соединении длинной ножки светодиода с GPIO17 на Pi, а короткой ножки — с резистором 220 Ом, а затем с любым выводом GND [Земля] на Pi. См. схему в загрузке для этого урока.

В терминале создадим новый файл с именем **01-blink.sh** и отредактируем его в *nano*:

```
$ nano 01-blink.sh
```

В *nano* мы можем начать писать код, начав с “shebang [ша-лаш]” (**#!/**), который велит Raspbian обрабатывать файл как исполняемый и сообщает, где найти «интерпретатор», который будет запускать код:

```
#!/bin/bash
```

Для настройки контакта GPIO, чтобы в данном случае светодиод, подключенный к GPIO 17, зажегся, надо задать его как выход:

```
gpio mode -g 17 out
```

Основным циклом нашего проекта будет бесконечный цикл **while true**, как в Python, чтобы код внутри цикла работал всегда. Чтобы выполнять код в цикле, начнем его с **do** и завершим **done**:

```
while true; do
```

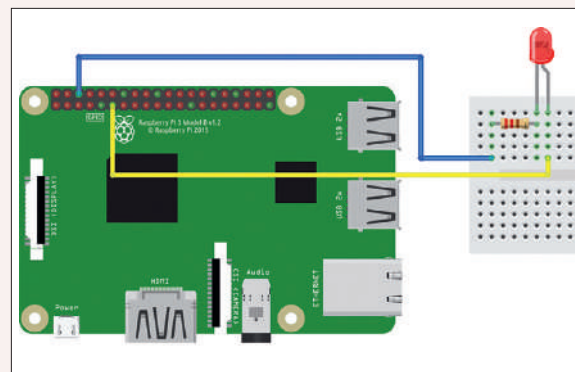
Код для мигания светодиода включает вызов команды *gpio* и указание ей использовать GPIO17 для включения. Это делается с помощью **write 17 1**. Затем мы велит коду подождать одну секунду, с помощью команды **sleep**:

```
gpio -g write 17 1
```

```
sleep 1
```

Теперь надо выключить LED, той же командой *write*, но на сей раз для выключения GPIO17 мы используем **0**. Затем — **sleep** еще на секунду до окончания цикла. Задержка после выключения светодиода дает нам идеальный цикл мигания светодиода:

```
gpio -g write 17 0
```



Проекты 1 и 2 используют одну и ту же схему с обычным светодиодом, подключенным к GPIO при помощи макетной платы, резистора и нескольких перемычек.

```
sleep 1
```

```
done
```

Сохраните файл, нажав **Ctrl+X**. Когда попросят сохранить буфер, нажмите **Y**, а затем **Enter**.

Прежде чем тестировать код, надо сделать файл исполняемым. В терминале введите

```
$ chmod +x 01-blink.sh
```

Теперь мы можем испытать код. Введите в терминале следующую команду (когда понадобится выход из нее, нажмите **Ctrl+C**):

```
$ ./01-blink.sh
```

Ну вот, у нас есть мигающий светодиод чисто на *Bash*.

Проект 2: Настройка цикла **for**

Используем ту же схему, что и в Проекте 1. В терминале создайте новый файл и отредактируйте его в *nano*:

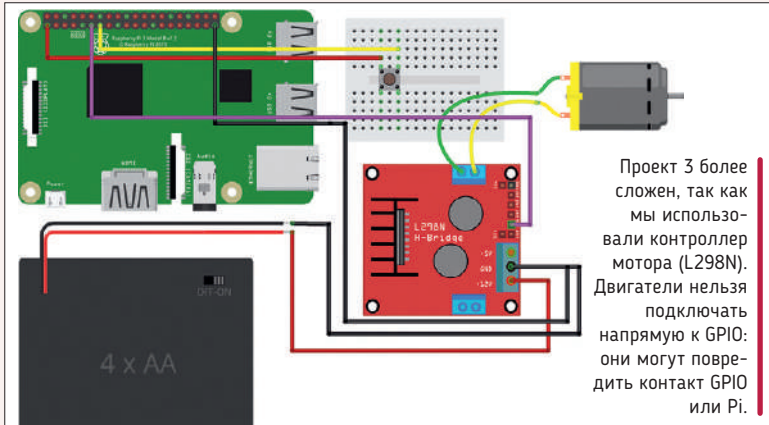
```
$ nano 02-for-loop.sh
```

Начнем код точно так же, как и в Проекте 1 со светодиодом — **#!/bin/bash**

```
gpio mode -g 17 out
```

но вместо бесконечного цикла используем цикл с тремя повторами. Циклы в основном используются для повторения задания несколько раз или перебора элементов в массиве (списке); в нашем случае мы используем цикл **for** для трехкратного мигания светодиода. Как и в случае цикла **while**, мы обрамляем нужный нам код внутри цикла с помощью **do** и **done**.

```
for i in {1..3}; do
```

Проект 3 более сложен, так как мы использовали контроллер мотора (L298N). Двигатели нельзя подключать напрямую к GPIO: они могут повредить контакт GPIO или Pi.

А для мигания светодиода мы используем код из проекта 1 для включения и выключения светодиода с односекундным интервалом.

```
gpio -g write 17 1
sleep 1
gpio -g write 17 0
sleep 1
done
```

Сохраните файл, нажав Ctrl+L, а когда попросят сохранить буфер, нажмите Y, а затем Enter.

До проверки кода надо сделать файл исполняемым, как и в проекте 1, только не забудьте изменить имя файла.

Теперь мы можем протестировать код. В терминале введите следующую команду (когда будет нужен выход, нажмите Ctrl+C):

```
$ ./02-for-loop.sh
```

Итак, мы выполнили Проект 2 — светодиод, мигающий три раза.

Проект 3: Управление мотором

Если вы смогли включить светодиод, то сможете управлять и мотором. Всё, что вам нужно, это плата управления двигателем! Для Проекта 3 надо обратиться к электрической схеме в загрузке для нашего урока. На макетной плате соединим кнопку с GPIO 27 и 3V; тогда при нажатии кнопки 3V от GPIO подключается к GPIO27, подавая на него высокий уровень и запуская работу кода.

GND схемы L298N должен быть подключен к GND Pi с помощью проволочной перемычки мама–папа, а GND батареи также подключается к L298N. Затем подключите Vcc (красный провод) батарейного отсека к клемме 12V L298N. Теперь подключите GPIO17 на Pi к IN1 на L298N, используя проволочную перемычку мама–мама. Наконец, подключите двигатель постоянного тока к OUT 1 и OUT 2 на L298N; порядок подключения провода двигателя значения не имеет. В терминале создайте новый файл и отредактируйте его в *nano*, набрав

```
$ nano 03-motor.sh
```

В *nano* начнем код точно так же, как в проектах 1 и 2.

```
#!/bin/bash
gpio mode -g 17 out
```

Однако добавим настройки для кнопки на GPIO27 и обеспечим переключение этого контакта в режим входа:

```
gpio mode -g 27 input
```

Нам понадобится цикл `while true`, чтобы постоянно запускать код, необходимый для обнаружения нажатия кнопки. Первым шагом в цикле является создание переменной с именем `PINSTATE` — в ней хранятся выходные данные команды, считывающей состояние контакта кнопки (GPIO27):

```
while true; do
PINSTATE=$(gpio -g read 27)
```

Считывание контакта вернет либо 0 (не нажато), либо 1 (нажато), и нам следует использовать условный тест для проверки данных, хранящихся в `PINSTATE`. Если кнопка нажата, мы видим 1, и тогда активируем следующий раздел нашего кода:

```
if [ $PINSTATE = "1" ]; then
```

Мы используем цикл `for` из Проекта 2 для включения и выключения GPIO17 три раза с односекундной паузой между каждым действием. Когда GPIO17 подключается к IN1 на L298N, наш двигатель запускается!

```
for i in {1..3}; do
gpio write 0 1
sleep 1
gpio write 0 0
sleep 1
done
```

Последние две строки кода используются для замыкания проверки условия и затем для закрытия цикла.

```
fi
done
```

Сохраните файл, нажав Ctrl+X. Когда попросят сохранить буфер, нажмите Y, а затем Enter.

До проверки кода сделайте файл исполняемым, как в проектах 1 и 2. Теперь мы можем запустить код:

```
$ ./03-motor.sh
```

Нажмите кнопку и посмотрите, как двигатель оживает! Итак, у нас готов Проект 3 — мотор, управляемый *Bash*.

Три проекта завершены, и мы использовали не что иное, как немного магии *Bash*! Так что стряхните пыль с книг про *Bash* и попробуйте создать собственные проекты на этом превосходном языке. **LXF**

СКОРАЯ ПОМОЩЬ

Не уверены, чем у вас является контакт GPIO — входом, выходом, включением или выключением? Команда `gpio readall` сообщит вам, включен ли контакт, и вход это или выход. Надо проверить один контакт? Попробуйте `gpio -g read <номер контакта GPIO>`. Мы делали так в Проекте 3.

» WIRING Pi

Wiring Pi (<http://wiringpi.com>) создана Гордоном Хендерсоном для опытных разработчиков; это полнофункциональная библиотека для доступа GPIO через контакты, с интерфейсом, подобным C и C++. Она работает аналогично системе “wiring” Arduino, так что кодеры Arduino почувствуют себя как дома, управляя GPIO Raspberry Pi. Используя *Wiring Pi*, мы всё еще можем взаимодействовать с шиной I2C и SPI и использовать имеющиеся компоненты и устройства, основанные на этих интерфейсах.

Утилита *gpio*, которую мы применяем на нашем уроке, на самом деле не из *Wiring Pi*: это скорее подручный инструмент *Bash* для управления контактами. Говоря о контактах, *WiringPi* и утилита *gpio* используют нумерацию выводов, отличную от стандартной схемы Broadcom (BCM), одобренной Raspberry Pi Foundation. В уроке мы использовали ключ `-g`, чтобы заставить *gpio* использовать схему выводов BCM, это намного проще для тех, кто приходит с Python. Если вы не можете вспомнить назначение контакта, не беспокойтесь — у утилиты *gpio* есть специальная команда. Она опросит все контакты, проверит их состояние и выдаст таблицу соответствия контактов BCM и *Wiring Pi*.

```
$ gpio readall
```

И если вам надо включить светодиод в скрипте *Bash*, то *gpio* сделает именно это, не заставляя вас писать код Python.

» ПОДПИШИТЕСЬ НА ЖУРНАЛ **LINUX FORMAT** на www.linuxformat.ru/subscribe!

МЕДИА-СЕРВЕР Часть 2

Расширим многокомнатную настройку

Ник Пирс открывает, как расширить вашу многокомнатную аудиосистему, добавив сенсорные контроллеры и возможности Bluetooth.



**НАШ
ЭКСПЕРТ**

Ник Пирс

Начиная с первой части, Ник экспериментировал со своей многокомнатной системой и наслаждался музыкой... когда ему удавалось отвоевать планшет у своей восьмилетней дочери! (Да, он чудовище.)

На прошлом уроке мы сумели создать себе многокомнатную аудиосистему всего за часть цены системы Bose или Sonos, используя Raspberry Pi в качестве сервера с *Logitech Media Server* с открытым кодом (известным также как *Squeezebox*). Сегодня мы восполним пробелы, добавив ряд сенсорных контроллеров, обеспечивающих большие удобства при воспроизведении и управлении вашей библиотекой.

Рассматривайте контроллеры как улучшенное дистанционное управление, с дополнительным преимуществом: они позволяют контролировать любой или все плееры в вашей домашней системе. Когда Logitech продавал собственную многокомнатную систему *Squeezebox*, его основной сенсорный контроллер имел вид высокотехнологичного дистанционного управления, который по-прежнему можно найти в секонд-хэнде... если знать, где искать.

Контроллер Pi

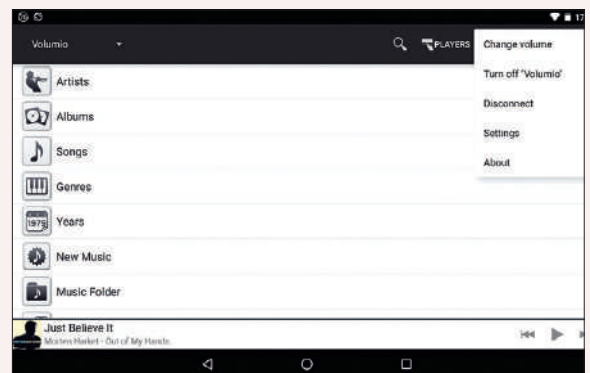
Но зачем платить £99, когда есть альтернатива куда дешевле? Например, почему бы вместо этого не задействовать официальный сенсорный экран Pi в роли полнофункционального контроллера? То, что мы покажем далее, прекрасно работает с официальным сенсорным экраном Raspberry Pi, но адаптируемо для более мелких экранов (см. врезку внизу справа стр. 93).

Если у вас работает *PiCorePlayer*, то для добавления поддержки сенсорного экрана всё необходимое уже встроено в ОС. Начните со входа на свой Pi через браузер — его IP-адрес вы можете получить из сообщений, появляющихся на сенсорном экране при загрузке Pi.

Убедитесь, что у вас выбрано *Normal* или *Advanced*; потом перейдите на вкладку *Tweaks* [Настройка] и прокрутите вниз до раздела *Jivelite Setup*. Сначала нажмите кнопку *Install* [Установить] внизу и подождите, пока закончится загрузка и установка.



Скин *Grid* предлагает интерфейс со значками, который превратит ваш сенсорный экран Pi в контроллер LMS.



Пользователи Android могут бесплатно управлять своей многокомнатной аудиосистемой через приложение *Squeezeboxer*. Оно немного ограничено, но вполне функционально.

Когда вас попросят, нажмите *OK*, чтобы перезагрузиться, и переключите внимание на сенсорный экран, чтобы завершить процесс настройки.

Сначала выберите язык, затем скин. *Grid* (800×400) использует крупные значки, а *Joggler* показывает пункты в списке. Сделайте свой выбор (вы можете сменить его позднее через *Settings > Screen > Select Skin* [Параметры > Экран > Выбор скина]), и вы попадете в главное меню, где подтвердите свой выбор.

Точная настройка

Далее нажмите *Choose Player* [Выбор плеера] (помните, контроллер работает с любым плеером). Когда выбор сделан, отображаются все плей-листы, приписанные к плееру. Нажмите на кнопку с нотой — вы увидите, что воспроизводится в данный момент, и получите доступ к управлению воспроизведением.

При поиске музыки для воспроизведения на контроллере сенсорного экрана вашего Pi нажмите и удерживайте исполнителя, альбом или трек, и вы увидите контекстное меню со множеством опций, включая немедленное воспроизведение, воспроизведение после текущего трека и сохранение в избранном.

Кнопки *My Music* [Моя Музыка] и *Radio* [Радио] отображают такие же опции, обнаруженные на экране воспроизведения сервера, хотя имеется встроенный инструмент поиска. Вы также увидите ссылки на приложения — нажмите *App Gallery*, чтобы увидеть имеющиеся приложения, разделенные на категории. Они содержат разные потоковые сервисы, но есть также приложение для интеграции Flickr в ваш контроллер для просмотра фото.



Перейдите в **Settings** [Параметры], и вы найдете опции для тонкой настройки вашего контроллера. Среди полезных настроек — возможность интегрировать слайд-шоу локально размещенных фото, опция включения спящего режима и меню *PiCorePlayer*, где можно выключить или возобновить сенсорную работу Pi.

Если официальный сенсорный экран вам не по карману, врезка внизу расскажет о более доступных опциях для добавления в вашу систему дружелюбного к пользователю сенсорного контроллера.

Динамики Bluetooth

Мы уже видели, как легко интегрировать динамики AirPlay в вашу многокомнатную аудио систему; а как насчет динамиков или наушников Bluetooth? Качество аудио не блестящее — например, нет поддержки aptX — но это хорошая опция для интеграции в вашу систему доступного динамика Bluetooth.

В отличие от AirPlay, динамики Bluetooth надо подключать к отдельному плееру; самая простая опция — через ваш мобильный телефон (тогда вам еще понадобится приложение медиаплеера — помните, что можно установить плагин для добавления поддержки DLNA/UPNP в LMS или приобрести отдельный плеер Squeezebox, например, *SqueezePlayer* — £4,19 для Android), но можно подключить их и к вашему плееру Raspberry Pi.

Если вы идете путем Pi, есть одна важная оговорка: старайтесь не использовать динамики Bluetooth с любым Pi, подключенным к сети через 2,4-ГГц диапазон Wi-Fi — по причине интерфейса. Это означает, что вы подключаетесь через кабель Ethernet, или используете 5-ГГц диапазон (тогда нужен внешний 802.11ac Wi-Fi адаптер, если у вас не самая последняя модель Pi 3 Model B+).

Опять же, *PiCorePlayer* делает интеграцию динамиков Bluetooth простой задачей, и весьма разумно блокировать любую попытку включить одновременно и внутренний Wi-Fi, и Bluetooth, если только у вас не RPi 3B+.

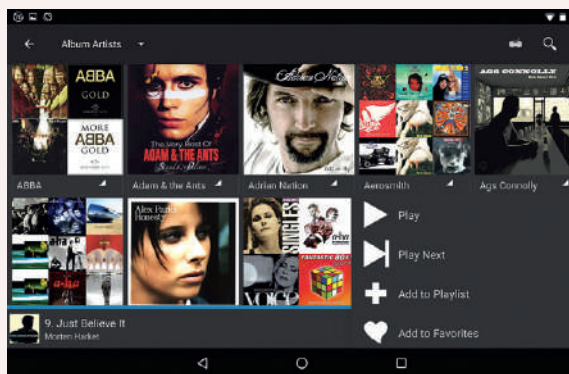
Сначала перейдите на вкладку **Wi-Fi Settings** [Настройки Wi-Fi] — убедитесь, что **RPi built-in Wi-Fi** [Встроенный Wi-Fi RPi] отключен, а **RPi built-in Bluetooth** [Встроенный Bluetooth RPi] установлен в **On**, после чего нажмите на **Save/Connect** [Сохранить/Подключить] и **Cancel** [Отмена], когда вам предложат перезагрузиться. Теперь перейдите на вкладку **Squeezelite Settings** [Настройки Squeezelite], установите устройство вывода аудио на **Bluetooth-audio** из выпадающего меню и нажмите на **Save** [Сохранить] — на сей раз по приглашению согласитесь перезагрузиться.

Вернувшись в браузер, выберите вкладку **Beta** на главной странице, затем вернитесь в **Squeezelite Settings** и нажмите **Card Control** [Контроль карты]. Нажмите **Install Bluetooth on rCP** и подождите, пока загрузятся и установятся расширения.

Перезагрузитесь, когда вам это предложат, и вернитесь в **Squeezelite Settings** > **Card Control** [Настройки Squeezelite > Управление картой]. Вы должны увидеть, что теперь отмечены первые два пункта, и только последний — **BT Device is Not Connected** [Устройство BT не подключено] — еще отключен. Прокрутите вниз, где вы увидите опции для подключения и выбора устройств Bluetooth. Переключитесь на динамик Bluetooth, переведите его в режим сопряжения и нажмите **Scan**.

Будем надеяться, ваше устройство обнаружится с первой попытки, но пробуйте дальше, пока оно не покажется и не появится кнопка **Pair** [Соединить]. Выберите устройство в выпадающем меню и нажмите на кнопку. Чтобы соединение удалось, может потребоваться более двух попыток.

После соединения выберите динамик из выпадающего меню рядом с **Select** [Выбрать], если еще видно **BT Device is Not Connected** [Устройство BT не подключено], затем нажмите кнопку; когда экран обновится, это должно быть отмечено, показывая, что вы готовы к попытке воспроизвести музыку через ваши новые беспроводные динамики Pi. Зайдите на сервер или



укажите выбранному вами контроллеру на требуемый Pi и затем запустите воспроизведение.

Если окажется, что оно работает, но звука нет, отключите динамик, перезагрузите Pi и включите динамик сразу, как только он начнет загружаться — это должно решить проблему. Не все динамики будут работать — нам, например, удалось подсоединить Aether Cone, но звук так и не пошел; однако наше радио Azatom Horizon DAB с Bluetooth после перезагрузок работало отлично.

Продолжайте исследования

Теперь вы добавили контроллер и, возможно, интегрировали в свою систему динамики или наушники Bluetooth, и ваша многокомнатная аудиосистема начинает обретать форму. Продолжайте исследовать — в частности, загляните на вкладку **Plugins** [Плагины] в **Settings** вашего LMS-сервера, чтобы узнать о возможностях расширения системы.

Например, по мере роста вашей музыкальной библиотеки вы, возможно, захотите создать меньшие по размеру под-библиотеки по определенным критериям, например, по исполнителям. Установите плагины **Custom Browse** и **Multi Library** и следуйте руководству на http://wiki.slimdevices.com/index.php/Multi_Library_plugin. В частности, весьма интересны могут быть ссылки на создание меню просмотра для классической музыки или динамического меню для просмотра по тега вместо жанров. **LXF**

Orange Squeeze, может, и не бесплатно, но у вас будет больше опций и куда более симпатичный интерфейс пользователя, чем в бесплатном приложении *Squeezer*.

СКОРАЯ ПОМОЩЬ

Если в вашем сенсорном контроллере Jivelite всё вверх ногами, вернитесь в свой браузер и раздел **Tweaks**. Здесь вы увидите опцию поворота экрана на 180 градусов, что — после перезагрузки — решит проблему.

» БОЛЬШЕ ОПЦИЙ КОНТРОЛЛЕРА

Если у вас нет Pi с официальным сенсорным экраном, есть множество более дешевых альтернатив добавления контроллера к вашей многокомнатной аудиосистеме. Очевидным выбором будет использовать в качестве контроллера свой телефон или планшет. Пользователи Android могут управлять своей LMS-системой с помощью бесплатного приложения *Squeezer* — оно дает доступ ко всем основным функциям: просмотру библиотеки, созданию плей-листов и переключению плееров. Имеются функции управления воспроизведением, вы можете управлять настроенными предупреждениями, но есть и слабые места. Одно из самых заметных — отсутствие вида **Album Artists**: если вам это подходит, потратьте вместо этого £3,99 на *Orange Squeeze*, где имеется этот вид, а также дополнительные функции.

Если вы хотите контролировать свою музыку с устройства Apple, то пользователи iPad могут попробовать *SqueezePad* (£4,99) или выделить £8,99 на *iPeng*, который также работает на iPhone и iPod. Если вам нужен только сенсорный экран Pi, но вы не можете позволить себе официальную семидюймовую модель, найдите маленький экран подешевле.

Если вы пойдете по этому пути с экраном 480×320, установите расширение **pcpjivelite_default-wav35skin.tcz** (в репозитории **piCore Sourceforge**, доступном из основного web-интерфейса *PiCorePlayer*), чтобы оно обеспечило вам рабочий интерфейс пользователя.

УДАЛЕННОЕ УПРАВЛЕНИЕ

Соединения с Raspberry Pi

Лес Паундер показывает, насколько легко надежно управлять любой моделью Raspberry Pi с устройства Windows или Android.



**НАШ
ЭКСПЕРТ**

Лес Паундер — фрилансер, который работает с такими организациями, как Raspberry Pi Foundation, продвигая навыки умельцев. Читайте его блог на bigl.es.

Будучи пользователями Linux, мы отлично подкованы технически. Мы знаем о преимуществах SSH перед telnet. Мы используем VNC, а не RDP. Но если вы — новичок в мире Raspberry Pi, есть шанс, что вы новичок и в Linux. Поэтому на нашем уроке мы рассмотрим, как подключить Pi с помощью устройств Windows и Android. Для всех частей нашего руководства мы будем использовать самый последний образ Raspbian, скопированный на пустую microSD карту с помощью *Etcher*, скачанного с <https://etcher.io>.

1 Подключение Pi через SSH

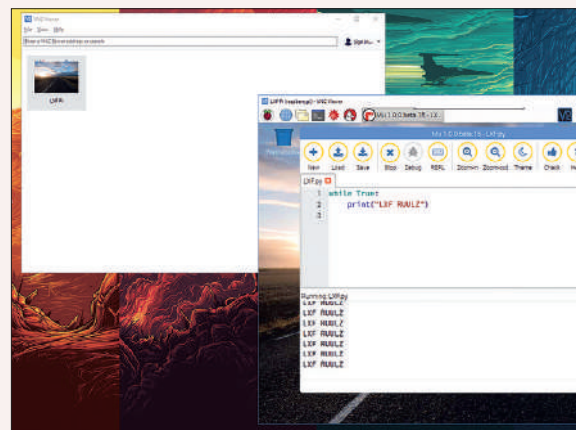
SSH — Secure Shell — это безопасный метод удаленного входа на компьютер через ненадежную сеть (обычно Интернет), и это самый общий способ удаленного управления Raspberry Pi. Чтобы использовать SSH на нашем Pi, надо вставить в наш компьютер карту microSD с установкой Raspbian, и мы должны увидеть, как в менеджере файлов появится раздел BOOT. Откройте раздел BOOT и создайте новый файл с именем **ssh**, оставив его без расширения, иначе он работать не будет. Извлеките microSD карту и установите ее на свой Pi.

Включите Pi и подождите, пока он успешно загрузится. Экран, интернет-соединение, клавиатура и мышь пригодятся, пока мы будем настраивать SSH, но после настройки можно будет отключить всё, кроме питания и сетевого соединения. Просто создав этот файл **ssh** на Pi, мы успешно настроили наш Raspberry Pi на использование SSH. Это будет наш SSH-сервер.

Bonjour и добро пожаловать, друзья мои!

Итак, теперь нам надо установить две программы на наш компьютер с Windows. Первая называется *Bonjour*, приложение от Apple, которое позволяет использовать имя хоста нашего Raspberry Pi вместо IP-адреса. Это означает, что IP-адрес нашего Raspberry Pi может измениться, но имя хоста останется прежним, пока мы его не изменим. Скачайте *Bonjour* с <https://support.apple.com/downloads/bonjour-for-windows> и установите. После установки нам не надо будет снова его вызывать — оно будет работать в фоновом режиме на нашем компьютере.

Итак, вторая программа для установки на нашем компьютере Windows — клиент SSH под названием *puTTY*. Мы можем скачать *puTTY Portable* с https://portableapps.com/apps/internet/putty_portable и затем установить его на нашем компьютере



Здесь мы видим VNC Viewer, используемый в Windows 10 для удаленного управления нашим тестовым Raspberry Pi 3. Мы пишем код прямо на Pi.

с Windows. Теперь откройте *puTTY*, и вы увидите довольно мудреное окно. Но сосредоточьтесь в центре окна, и вы увидите там “Host Name”. Теперь введите

raspberrypi.local

Затем ищите **Saved Sessions** [Сохраненные Сессии] и в пустом поле назовите сессию

My Pi

Нажмите **Save**. Теперь, когда бы вы ни загрузили *puTTY*, жмите **My Pi** и затем **Load**. А сейчас нажмите **Open**, чтобы запустить подключение к вашему Pi. У вас спросят ваше имя пользователя — это **pi**, а пароль по умолчанию — **raspberrypi**. Обратите внимание, что когда вы вводите пароль, он не будет виден на экране по соображениям безопасности. Нажмите **Enter**, и через несколько секунд откроется терминал, через который можно управлять вашим Raspberry Pi.

Оставаясь в терминале, займемся техобслуживанием. Для начала позаботимся об обновлении нашей системы; в Терминале введите следующее и нажмите **Enter** для запуска каждой строки:

```
$ sudo apt update
```

```
$ sudo apt upgrade -y
```

Обновив программы, мы можем запустить еще одну команду в Терминале. Эта команда запустит VNC-сервер, способ

ВАМ НУЖНЫ

- Любая модель Raspberry Pi
- Самая свежая версия Raspbian
- Ноутбук или ПК с Windows
- Планшет или телефон на Android
- Интернет-соединение для всех устройств

удаленного управления рабочего стола компьютера через подключение. VNC-соединения не шифруются, поэтому не используйте их через Интернет. Используйте их только в надежных сетях, например, дома или в офисе. Запустите в терминале следующую команду:

```
$ vncserver
```

На экране промелькнет много всякой информации, но самая важная часть — последняя строка, которая будет выглядеть примерно так.

```
New desktop is raspberrypi:1 (192.168.0.10:1)
```

Это IP-адрес нашего Pi, за которым следует двоеточие (:). Запишите эту информацию. Она вам понадобится!

Управляем нашим Pi через VNC

Теперь мы можем отойти от *puTTY*, но на всякий случай не закрывайте его. На компьютере с Windows откройте браузер, зайдите на сайт *RealVNC* и найдите установку *VNC Viewer* для Windows (www.realvnc.com/en/connect/download/viewer/). Скачайте и установите *VNC Viewer*, и когда будете готовы, запустите приложение *VNC Viewer*.

Появится большое серое окно. Основная область — наша “address book [адресная книга]”, где обычно хранятся часто используемые соединения VNC. Давайте создадим новое соединение. В адресной строке вверху окна введите имя хоста вашего Pi, за которым следует двоеточие и затем номер один — номер логического экрана. Например,

```
raspberrypi.local:1
```

Нажмите Enter, и вас попросят подтвердить, что наш VNC-сервера, наш Pi, в порядке. Нажмите *Continue* [Продолжить], и когда вас попросят, введите имя пользователя и пароль для вашего Pi. Это должны быть имя пользователя **pi** и пароль **raspberrypi**, если вы их не изменили, как предложено в последнем релизе Raspbian.

Через несколько секунд вы увидите рабочий стол Raspbian, и поводя мышью по рабочему столу, вы увидите, что можете управлять рабочим столом Raspbian, как будто вы сидите перед Raspberry Pi!

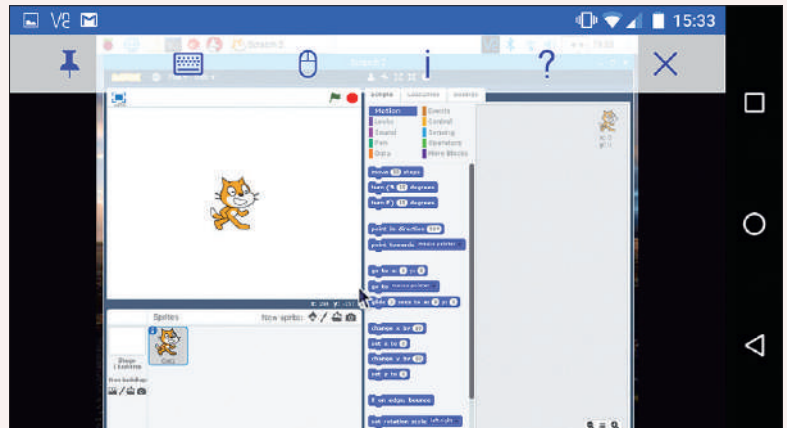
Итак, наш VNC-сервер работает на нашем Pi; давайте позаботимся, чтобы он запускался автоматически, при загрузке нашего Pi. Для этого надо перейти в меню Raspbian, которое находится в верхней левой части экрана. Выберите *Preferences* [Предпочтения] и затем нажмите *Raspberry Pi Configuration* [Настройка Raspberry Pi]. Когда приложение откроется, нажмите *Interfaces* [Интерфейсы] и затем *Enable the VNC interface* [Включить интерфейс VNC]. Нажмите OK, и приложение закроется.

Чтобы проверить его работу, перезагрузите Raspberry Pi через меню *Shutdown* [Выключение]. Как только Pi полностью перезагрузится, переподключитесь к VNC-серверу с помощью приложения *VNC Viewer*. Но вам, возможно, придется изменить номер дисплея после двоеточия на 0 — поскольку Pi работает без экрана — например, `raspberrypi.local:0`.

Маленькие зеленые человечки

У Android имеется ряд приложений, применимых для удаленного управления вашим Raspberry Pi. Для SSH мы рекомендуем *Juice SSH* (<https://play.google.com/store/apps/details?id=com.sonelli.juicessh>) и *ConnectBot* (<https://play.google.com/store/apps/details?id=org.connectbot>), оба они — бесплатные приложения, которые можно использовать в качестве клиентов SSH.

У *Juice SSH* лучше интерфейс, и простое нажатие на Terminal выведет расширенные клавиши быстрого запуска и кнопки курсора для навигации по журналу в *Bash*. Правда, проблема



Использовать VNC Viewer для Android бывает непросто, поскольку ваш тачскрин станет тачпадом для мыши, а ввод с клавиатуры усложнится. Но удобно, если вам реально нужен контроль.

в том, что мы не можем использовать имя хоста Pi: придется использовать IP-адрес Pi. Его можно найти с помощью другого приложения под названием *Fing* (<https://play.google.com/store/apps/details?id=com.overlook.android.fing>), которое умеет сканировать сеть в поисках устройств, сообщать их IP-адреса и сканировать на предмет наличия у них портов (например, для подключения VNC и SSH), открытых и готовых к использованию.

Для VNC официальный *VNC Viewer* для Android (<https://play.google.com/store/apps/details?id=com.realvnc.viewer.android>) так же быстр и прост в использовании, как и наша версия Windows. И снова для подключения через Android нам придется использовать IP-адрес.

Вот мы и узнали, как подключать модель Raspberry Pi и управлять ею с помощью устройства Windows и Android. Удобно для робототехники, удаленного управления и получения данных, которые находятся за несколько миль от вашей спальни. **LXF**

СКОРАЯ ПОМОЩЬ

Включили SSH на своем Pi, но не можете найти его в сети? Войдите на свой роутер и проверьте раздел *Connected Devices* [Подключенные Устройства]. Он покажет все устройства, подключенные к вашей сети, и информацию о них.

» ГДЕ МОЙ Pi, ЧУВАК?

Использование имени хоста для подключения к Pi отлично работает в Linux и Mac (где оно установлено по умолчанию) и в Windows, где вы установили *Bonjour*. Но как надежно подключиться к Pi пользователям Android или тем, кто хочет использовать только IP-адреса?

Во-первых, надо сесть за Pi, доступ к которому нам нужен, с клавиатурой, мышью и т.д., готовыми к работе. В Терминале выясним IP-адрес Pi. Введите следующее и нажмите Enter:

```
$ hostname -I
```

IP-адрес обязательно запишите.

Далее, в том же Терминале используем этот IP-адрес для создания статического IP-адреса. В Терминале введите следующее и нажмите Enter:

```
$ sudo nano /boot/cmdline.txt
```

Прокрутите до конца строки и добавьте свой IP-адрес в строку. Но не нажимайте Enter! Это должна быть одна очень длинная строка.

```
ip = <ВАШ IP-АДРЕС>
```

Чтобы выйти и сохранить, нажмите Ctrl+X, затем Y, и нажмите Enter. Перезагрузите Pi и проверьте свой IP-адрес командой

```
$ hostname -I
```

Всё должно быть исправлено на указанный вами IP. Ну вот, у вас есть статический IP-адрес для вашего Pi.

» ПОДПИШИТЕСЬ НА ЖУРНАЛ **LINUX FORMAT** на www.linuxformat.ru/subscribe/

Ответы

Вас жгут вопросы по открытому коду или ядру? Каким бы ни был ваш уровень, присылайте их на lxformat@futurenet.com



Джонни Бидвелл

Пытается уладить ваши разборки с Туксом.

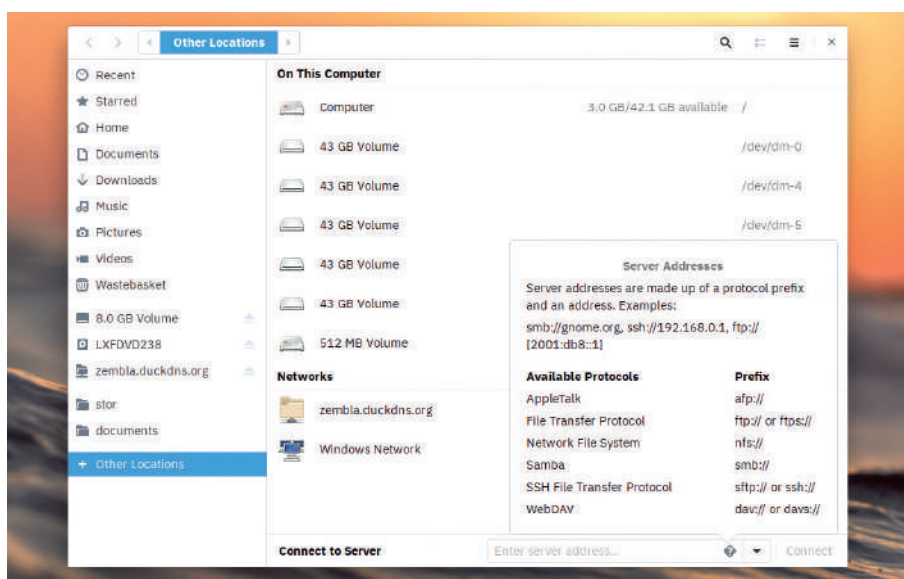
В Лучшие редакции Mint

Я пытаюсь установить Mint Xfce на Acer One 10 (модель S1003) с предустановленной Windows 10.

Rufus не работает. Я записал Mint с помощью Rufus на USB-диск, а Acer просто загружается в Windows. Поиск в Сети выявил, что у каких-то 64-битных Acer One 10 имеется 32-битный EFI вместо 64-битного. Это выше моего уровня понимания. Я позвонил в Acer в поисках помощи. Они сообщили, что за £60 решат мою проблему. Но, как многолетний подписчик, я надеюсь, вы мне поможете. Я хочу полностью избавиться от Windows 10, и если вы считаете, что для Acer есть более подходящая ОС, чем Mint Xfce, пожалуйста, сообщите мне!

Дэвид Брукс [David Brooks], по электронной почте

О Это отличное оборудование, и Mint прекрасно подойдет, если Вы сможете его установить. Есть несколько компьютеров с 32-битным EFI и 64-битным CPU, но я не уверен, что Ваша модель — одна из них. Может, это и неправильно, но работа с BIOS могла бы избавить Вас от головной боли. Некоторые модели Acer требуют включить загрузочное меню и добавить файл `grubx64.efi` с Вашего USB-диска



Вы можете просматривать ресурсы WebDAV из Gnome Files с помощью префикса `dav://` или `davs://`, однако с большими директориями будут проблемы.

в список надежных образов. Затем перезагрузитесь, нажмите F12 и выберите этот файл из загрузочного меню. Кроме того, попробуйте отключить безопасную загрузку (возможно, у Вас

не получится) и загрузиться в классический режим BIOS (CSM).

Предполагается, что 32-битный UEFI поддерживается в Ubuntu вскоре после выхода 16.04 (см. <https://bugs.launchpad.net/ubuntu/+source/grubinstaller/+bug/1592229>). И эти изменения дошли и до Mint 18.3. Стоит попробовать снова с Ubuntu 18.04, Fedora 28 или свежим Mint 19 (есть на нашем диске **LXF239/240!**) на случай, если что-то изменилось.

Вы всегда можете попробовать старый способ — скачать 32-битный образ *Grub* с <https://github.com/jfwells/linux-asust100ta/tree/master/boot> (или следовать инструкциям здесь, чтобы создать собственный на компьютере Linux) и затем поместить этот файл в директорию `/EFI/boot` вашего USB, созданного с помощью Rufus. И снова, Вам может понадобиться добавить этот файл в список загрузки в BIOS.

Как это ни печально, поддержку Linux на потребительском уровне предлагают весьма немногие производители, и любая техподдержка первой линии, к которой Вы обратитесь, вряд ли сообразит, о чем Вы спрашиваете. Вероятно, сумма в £60, которую они запросили — это стандартная плата за переустановку исходной ОС. И они уж точно не станут устанавливать Вам неподдерживаемую ОС.

» ВЫРЫВАНИЕ ВОЛОС

В Я использую Plasma с проприетарным драйвером Nvidia. Для игр это отлично, но при просмотре видео — в частности, потокового видео в браузере — появляется горизонтальный разрыв примерно в нижней четверти экрана. Он очень заметный и очень раздражает, особенно во время активных сцен. Как мне это исправить, чтобы мои друзья перестали советовать мне использовать Windows?

Джон Шепард [John Sheppard], телеграммой

О Это давняя ошибка, которая и меня безумно раздражает. Способ взаимодействия композитора *Kwin* с драйвером Nvidia делает это проблемой скорее Plasma, чем Gnome, но предлагаемая нами отладка может сработать и там.

Сначала откройте панель Nvidia Settings, затем перейдите в X Server Display Configuration > Advanced [Настройки отображения X Server > Расширенные] и выберите Force Composition Pipeline [Принудительный конвейер композитора]. Теперь перезапустите браузер и посмотрите, решена ли проблема. Если нет, попробуйте снова с Force Full Composition Pipeline. Если сработает, можете сделать это изменение постоянным, создав файл (или добавив к нему) `/etc/X11/xorg.conf.d/20-nvidia.conf` следующие строки в разделе Screen:

```
Option "metamodes" "nvidiaauto-select +0+0
{ForceCompositionPipeline=On}"
Option "AllowIndirectGLXProtocol" "off"
Option "TripleBuffer" "on"
```

А что касается Ваших друзей, тут хватит двух слов: обновления Windows.

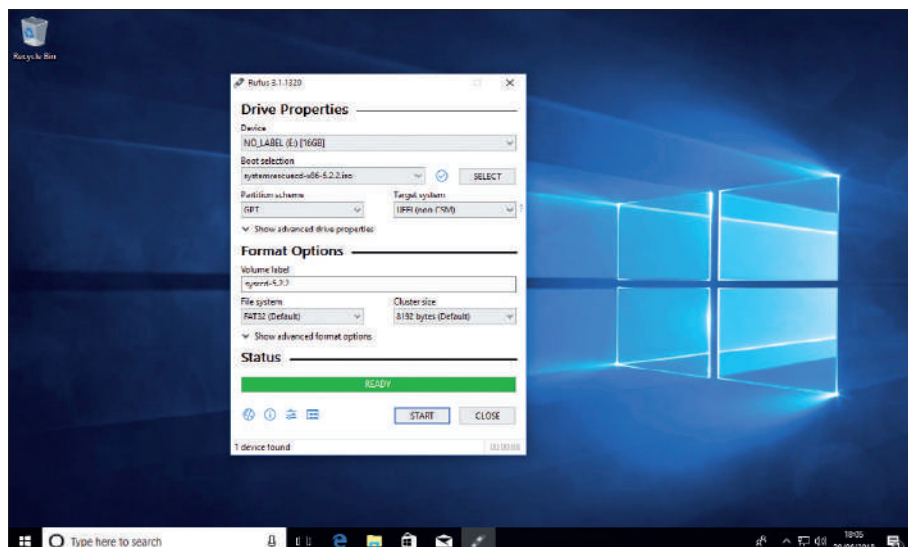
В Облака и серебряные локальные зеркала

Мне понравился ваш всесторонний материал о Nextcloud в прошлом выпуске — я сумел настроить рабочий экземпляр с HTTPS и рабочим DNS в течение дня. Я знаю, что запуск синхронизации клиента рабочего стола будет поддерживать постоянное обновление с сервером, эффективно создавая резервные копии. Однако я осторожен, и меня беспокоит возможность повреждения на сервере файла (или всех моих файлов) и что это повреждение может распространиться на все компьютеры, где работает клиент *sync*. В отсутствие другого режима резервного копирования это будет означать потерю данных. Не существует ли способа периодически синхронизировать мои файлы Nextcloud с моим домашним сервером, в идеале — без использования настольного клиента?

Происходит это ежедневно, у меня, по крайней мере, были бы шансы заметить, что на удаленном сервере что-то произошло, и я мог бы спасти файлы на своем домашнем сервере, пока резервное копирование их не ликвидировало. Я догадываюсь, что подобная стратегия предполагает использование *cron* и *rsync*, но с этими инструментами я как-то не сталкивался.

Родни Маккей [Rodney McKay],
по электронной почте

Это можно сделать несколькими способами. Один — использовать поддержку Nextcloud для WebDAV, но это процесс обходной. Сначала установите пакет *davfs2* на свой домашний сервер, что позволит Вам смонтировать



Rufus только для Windows, но он очень удобен, если вам нужно изменить ISO с помощью 32-битного образа Grub.

ресурсы WebDAV из командной строки. Посмотрите, работает ли это, создав точку монтирования, например, */mnt/webdav*, и затем запустив следующую команду:

```
$ sudo mount -t davfs https://nextcloud.yourdomain.com/nextcloud/remote.php/dav/files/username /mnt/webdav
```

Вам надо соответствующим образом изменить URL и имя пользователя [username]. У Вас должны спросить имя пользователя и пароль. Затем следует дать возможность Вашему пользователю монтировать, не вводя каждый раз пароль, поэтому размонтируйте */mnt/webdav* и создайте файл

~/davfs2/secrets на своем домашнем сервере с содержимым в виде:

```
https://nextcloud.yourdomain.com/nextcloud/remote.php/dav/files/username username password
and chmod 600 для безопасности. Потом добавьте в /etc/fstab строку
https://nextcloud.yourdomain.com/nextcloud/.../mnt/webdav davfs noauto,user 0 0
```

и посмотрите, может ли Ваш пользователь это смонтировать. Возможно, Вам предложат добавить своего пользователя в определенную группу (*davfs2* в Debian, *network* в Arch), в этом случае так и сделайте (*gpasswd -a user davfs2*), выйдите из системы и затем зайдите снова, чтобы увидеть изменения.

Davfs просто обеспечивает доступ по требованию к веб-ресурсу, он не создает его зеркала; поэтому, когда Вы его размонтируете, директорию */mnt/webdav* будет пустой. Чтобы создать зеркало, создайте другую директорию */mnt/ncmirror*, и используйте *rsync*, чтобы она зеркально отобразилась:

```
$ rsync -av /mnt/webdav /mnt/ncmirror
```

Необходимо настроить разрешения в обеих директориях, но, будем надеяться, идею Вы поняли. Чтобы всё это автоматизировать, внесите приведенную команду в скрипт (хорошо бы предварить ее командами *mount* и *umount*), а затем добавьте процедуру *cron* (или таймер Systemd) для запуска этого скрипта с требуемой Вам частотой.

Например, чтобы ежедневно запускать скрипт в полночь (предполагая, что *cron* настроен — это мы оставляем Вам в качестве упражнения), запустите *crontab -e* и добавьте строку

```
0 0 * * * ~/ncscript
```

На <http://bit.ly/nextcloud-webdav> есть немного более подробные официальные инструкции. Если WebDAV окажется слишком медленным, это можно также сделать с помощью *sshfs*, но тогда получится обход внутреннего аудита Nextcloud, что может вызвать — а может и не вызвать — проблемы.

» ПРИВЕТ УМНИКАМ

Современные рабочие столы часто хотят получить доступ к разным внешним источникам. Это нормально, но меня беспокоит то, какие сервисы (если они вообще есть) слушают локально и как мне их отключить.

Конечно, я знаю, когда у меня запущен SSH (и я могу сделать его безопасным с помощью открытого ключа); но есть ли команда, которая показывает, какие сервисы слушают и на каких портах? Я знаю, что некоторые сервисы прослушивают только локальный интерфейс, и потому они безобидны, но понимаю также, что к тому, что слушает интерфейс LAN, можно получить доступ через взлом роутера или через скомпрометированную машину в моей сети.

И как насчет сервисов, работающих через IPv6? Они блокируются роутером, но мне бы не хотелось на это полагаться.

Меня также беспокоит DNS. Я отказался от использования DNS сервера моего интернет-провайдера в пользу Cloudflare 1.1.1.1, который вычищает журналы через

24 часа. Но можно ли как-то зашифровать мои запросы DNS?

Имя и адрес скрыты

Отобразить все слушающие сервисы поможет любая из утилит *nmap*, *netstat* или *lsof*. Последняя используется в основном для LiSt Open Files, но также может показать, какие процессы слушают/отсылают на каких портах, через опцию *-i*. Запустите *lsof -i | grep LISTEN*, чтобы отфильтровать только слушающие сервисы. Будут показаны сервисы IPv4 и IPv6. Если Вы хотите вернуть контроль, надо будет создать компьютер с брандмауэром, работающим с чем-то вроде *pfSense*, *IPFire* или даже *DD-WRT*. Однако готовьтесь побороться с потерями связи.

DNSSEC предназначался для решения проблем DNS, но он сложный, и не был внедрен повсеместно. Можно направлять запросы DNS через HTTPS, используя, скажем, клиент *Cloudflare* (см. <https://developers.cloudflare.com/1.1.1.1/dns-over-https/>), чтобы найти более подробную информацию).

В BlueTooth-блюз

Привлеченный вашим захватывающим предложением динамика Bluetooth, с радостью сообщаю, что теперь я полноценный подписчик LXF. Это потрясающе, динамик работает с моим телефоном идеально, так что я могу действовать людям на нервы на улице своими эклектичными музыкальными вкусами.

Мне также нравится возможность доступа к архиву подписки, это просто невероятный кладёз информации. Меня интересуют проекты Raspberry Pi, поэтому я обзавелся Pi 3; но досаждают проблемы с работой звука. Звук прерывистый, особенно с YouTube. Такое ощущение, что машина надрыгается, стараясь справиться из последних сил, хотя у меня создалось впечатление, что Pi 3 довольно мощный.

Я был весьма горд и доволен собой, потому что изначально устройство вообще не работало. Пытаясь подключиться к устройству, я постоянно видел следующую загадочную ошибку: `connection failed — GDBus.Error.org.bluez.Error.Failed: Input/`

`output error`. Просидев некоторое время на форумах Raspberry Pi, я нашел решение, которое предусматривало переустановку файла настройки. Или это была моя интерпретация событий. Волшебной командой было:

```
sudo apt-get -o Dpkg::Options::="--force-confmiss"
install --reinstall raspberrypi-sys-modes
```

И всё-таки я хотел бы знать, не послужило ли это каким-либо образом причиной спотыкающегося звука, что на самом деле ненамного лучше, чем полное отсутствие такового, как мне теперь стало казаться.

ДБ Купер [DB Cooper], по электронной почте

О Спасибо за подписку — пожалуйста, посоветуйте своим друзьям тоже подписаться, чтобы начальство смотрело на нас благосклонно.

Прочитанные Вами ошибка и команда предполагают, что нечто сумело переписать или удалить критически важную часть настройки Bluetooth. Однако это не причина прерывания

звука; это общая проблема, не связанная именно с Raspberry Pi. Она связана с проблемами планирования и *PulseAudio*; аудиоданные недостаточно быстро передаются на устройство, что вызывает остановки и прерывания. Это усугубляется подключением Bluetooth.

Как я упоминал в статье о мультимедиа в LXF232, *PulseAudio* иногда обнаруживает ошибки в низкоуровневых драйверах ALSA. В 2010 г. *PulseAudio* перешел с планирования на основе прерываний на планирование на основе таймера. В теории драйверы должны выработать корректный временной график, но вот в области практики всё оказывается не так просто.

Самое простое решение — отключить планирование на основе таймера (у нас это сработало), что делается посредством добавления/изменения следующей строки в `/etc/pulse/default.pa`:

```
load-module module-udev-detect tsched=0
```

Затем выключите и перезапустите *PulseAudio* с помощью команд `pulseaudio -k` и `pulseaudio --start`.

Если это не работает, Вы можете применить более радикальные меры: удалить уровень *PulseAudio* и просто использовать чистый ALSA (*PulseAudio* не является заменой ALSA, он просто добавляет поверх него современные функции), однако это лишит Вас управления громкостью в каждом приложении по отдельности и прочих блистательных удобств.

В Прикольные огоньки

У меня есть одна идея для проекта Pi, но я не представляю, с чего начать. Мне бы хотелось воспроизводить музыку с моего Raspberry Pi, каким-то образом передавать ее в Python, Python будет выполнять рудиментарную обработку звука, и затем зажгутся светодиоды-LED на прямоугольной матрице, чтобы получился базовый спектральный визуализатор. Как только это будет запрограммировано, было бы здорово реализовать также распознавание ритма и соответствующую дискотечную светомузыку. Однако не всё сразу.

В Сети больше руководств типа «зажги LED», чем у меня испорченных светильников DMX, и есть также несколько руководств по обработке звука, но мне, похоже, не хватает какого-то мостика, чтобы объединить эти знания. У меня есть некоторые теоретические сведения об обработке цифровых сигналов, но я представления не имею, как всё это будет работать в Python.

Гарри Найквист [Harry Nyquist], по электронной почте

О Отличная идея, Гарри — Вы не против, если мы ее позаимствуем? Есть пара досадных мелочей, но мы никогда не позволяем досадным мелочам стоять на пути качественного программирования в Python.

Первая мелочь заключается в том, что аудио по своей природе зависит от времени. Поэтому всегда есть некоторая задержка между аудио и включением LED. Это не должно сильно мешать, особенно для спектрального визуализатора.



Представьте себе ярость и праведное негодование, если бы мы раздавали периферийные устройства, которые не работали с Linux.

» СОСТОЯНИЕ SAMBA

В Еще одно решение вопроса Марселя Фауста [Marcel Faust] из Ответов в LXF239/240 — это установка *Samba* и уникальное имя Raspberry Pi в ресурсе общего пользования. Это сделает возможным поиск из большинства операционных систем. Например, он может давать распределенный доступ к директории только для чтения с файлом состояния, содержащим IP-адрес и всё, что вам будет угодно (я использую *cron*, чтобы обновлять этот файл каждую минуту). У меня несколько Pi, выполняющих разные задачи, и это отлично работает.

Майк Спайерс [Mike Spiers], по электронной почте

О Это отличная идея, Майк. *Samba*, безусловно, подходит не каждому, но у нее есть свои скрытые достоинства. Вот однострочная команда, которая создаст такой файл состояния:

```
echo $(ip addr | grep 'state UP' -A2 | tail -n1 | awk '{print $2}' | cut -f1 -d '/') > /var/samba/status
```

Команда не использует фиксированных имен интерфейсов, поэтому она должна работать и для проводных, и для беспроводных соединений. Здесь отыскивается соответствие в первом из обнаруженных активных интерфейсов и извлекается IP-адрес, который сообщается инструментом *ip* (вездесущим в системах Linux).

Однако если бы Вы решились заняться обнаружением темпа/ритма, то понадобились бы определенные хитрости.

Как Вам, без сомнения, известно, в основе цифровой обработки сигналов [Digital Signal Processing, DSP] лежит преобразование Фурье. Ее можно вполне эффективно выполнять в Python с помощью *Numpy*, который обеспечивает «правильные» массивы для выполнения FFT [Fast Fourier Transform — Быстрое преобразование Фурье, БПФ] — списки Python очень гибкие, но не очень быстрые.

На Pi 2 и выше это, вероятно, будет достаточно быстро, и можно сделать еще быстрее с помощью Cython, если Вы готовы потратить часы на сравнительные тесты. Для первого Pi и Pi Zero можно использовать библиотеку Эндрю Холма [Andrew Holme] GPU_FFT, чтобы передать эту работу GPU (который проводит расчет «бантика» [butterfly, базовая вычислительная операция алгоритма БПФ, — прим. ред.] параллельно, т.е. значительно быстрее, чем одноядерный CPU). Но затем Вам надо каким-то образом передать трансформацию назад в Python, и, вероятно, лучше всего это сделать в Cython.

Все, кто интересуется этим, но ощущает недостаток знаний в математике/программировании, могут познакомиться с отличным материалом Питера Оньюна [Peter Onion] на www.peteronion.org.uk/FFT/FastFourier.html.

Касательно перевода аудио на Pi, вам нужно внешнее звуковое устройство для аналогового ввода, встроенное устройство Pi не поддерживает ввода (с микрофона или с помощью настройки ALSA устройства закольцовывания из канала



Adafruit Neomatrix – отличный способ начать работу с мигающими разноцветными проектами Pi и Arduino.

вывода). Если вы не хотите этого делать, MPD может выводить результат в буфер FIFO на диске (в то же время как аудиоустройство), что можно сделать с помощью Python. Именно это *ncurses* и другие клиенты MPD используют для своих плагинов визуализации.

Рассмотрите также вариант с CAVA (Console-based Audio Visualiser for ALSA — Визуализатор

аудио для ALSA на базе консоли) от Карла Ставестранда [Karl Stavestrand] на <https://github.com/karlstav/cava>.

Что касается матриц LED, Adafruit изготавливает их всевозможных видов и размеров. Однако вам понадобится для них отдельный источник питания: массиву 32×32 необходимо примерно 4А на 5В. **LXF**

» СТРАЖ MINT

В Я только что установил Mint 19, и, как и вы, нахожусь под сильным впечатлением от этого замечательного дистрибутива. Однако сейчас я пытаюсь установить его на ноутбук друга, и ничего не получается. Я уверен, что это проблемы с UEFI, потому что установка идет нормально, пока не доходит до попытки установить *Grub*. Мой компьютер использует классический BIOS, но моему другу нужна UEFI для двойной загрузки с Windows.

Я не хочу путаться с переключением между BIOS и UEFI для смены операционной системы, и не хочу рисковать установленной Windows, поместив работающий в BIOS *Grub* в master boot record (MBR).

Генри Корнхилл [Henry Cornhill],
по электронной почте

О Да, после того, как Mint выпустил обновление для пакета Ubuntu (*ubuntu-system-adjustments*), была, к сожалению, повреждена часть скрипта установки Mint Grub EFI. Простое решение — во время

установки отключить целевой компьютер от Интернета. Это позволит Вам обойти момент загрузки проблемного обновления. Как только система будет установлена, можно спокойно подключиться к Интернету и обновить всё обычным способом, поскольку к этому моменту *Grub* уже будет нормально пристроен к месту.

Единственный недостаток здесь в том, что после установки первое обновление отнимет немного больше обычного времени, поскольку придется обновлять все те пакеты, которые не были обновлены во время установки. На момент написания статьи команда Mint готовит отдельный ISO-образ, который должен быть готов к тому моменту, когда вы будете это читать.

Можно также установить *Grub* вручную из среды live, но сейчас на дворе 2018 г., и заниматься такими вещами вручную нам уже не подходит. В теории Windows с UEFI не должна мешать установка Linux в режиме BIOS, но мы бы не стали рисковать — тем более на чужом компьютере.

ПОЛУЧИТЕ ПОМОЩЬ!

Мы рады ответить на все вопросы по Linux, которые вы присылаете на lxr.answers@futurenet.com, независимо от их уровня.

У нас у всех когда-то были проблемы, так что не стесняйтесь. Однако мы всего лишь люди (хотя многие подозревают, что Джонни — робот), и очень важно, чтобы вы указывали максимум информации. Если что-то работает на одном дистрибутиве, а на другом — нет, сообщите нам. Если вы получили сообщение об ошибке, пожалуйста, сообщите нам его точный текст и из-за чего оно появилось.

Если у вас есть, или вам кажется, что есть проблема с оборудованием, расскажите нам о нем. Подумайте об установке *Hardinfo* или *Ishw*. Эти программы показывают оборудование на вашем компьютере. Если вы не хотите или не можете установить эти программы, запустите следующие команды в терминале root и пришлите нам также файл **system.txt**.

```
uname -a > system.txt
```

```
lspci >> system.txt
```

```
lspci -vv >> system.txt
```


HotPicks

MyTetra » SolarSystem » Elisa » Dilay » Shotwell
 » Stereophotoview » Browsh » Alacritty » SFXR-Qt
 » OpenRA » Defend



Александр Толстой

человек не злой; но в игре *OpenRed Alert* он из патриотизма всегда выбирает для себя становиться на сторону Советов и воевать против Запада.

МЕНЕДЖЕР ПЕРСОНАЛЬНОЙ ИНФОРМАЦИИ

MyTetra

Версия: 1.43.27 Сайт: https://github.com/xintrea/mytetra_dev

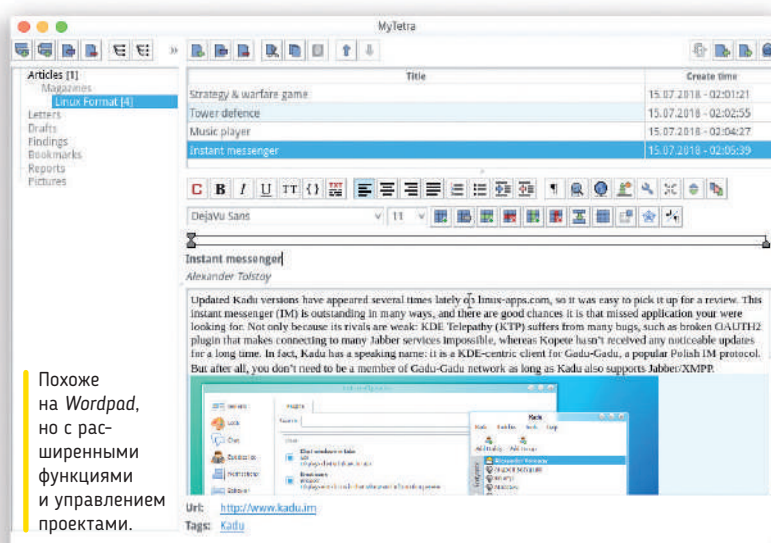
Мы бы рады воспевать бесконечную хвалу *Akonadi*, если бы он частенько не рушился; так что спрос на надежный, стабильный и хорошо продуманный менеджер персональной информации (PIM) в Linux продолжает существовать. Прежде чем брать для обзора *MyTetra*, мы погуглили существующие приложения PIM и вскоре обнаружили, что *MyTetra*, может, и не самый лучший, но предлагает интересные функции и в общем выглядит прочным.

Приятности возникают с момента его запуска. На нашей памяти *MyTetra* — это первое приложение, которое спрашивает, хотите ли вы сохранить его данные в `~/.config` (стандартное поведение) или в текущем каталоге (переносимая версия). Во втором варианте вы можете записывать свои заметки где угодно, например, на USB-накопителе.

Главное окно напоминает знаменитое приложение для заметок *CherryTree*, по которому у нас был премилый учебник в **LXF237**. Здесь располагаются три панели: на левой отображается дерево заметок; верхняя служит для табличного вида, подобного входящим в почте; плюс основная область для работы с вашими документами. *MyTetra* способен отлично форматировать текст, не зависая, потому что у *MyTetra* есть всё, что полагается иметь текстовому процессору, и по этой части он явно превосходит *Calligra Words* (ранее *KWord*), единственный текстовый процессор на Qt.

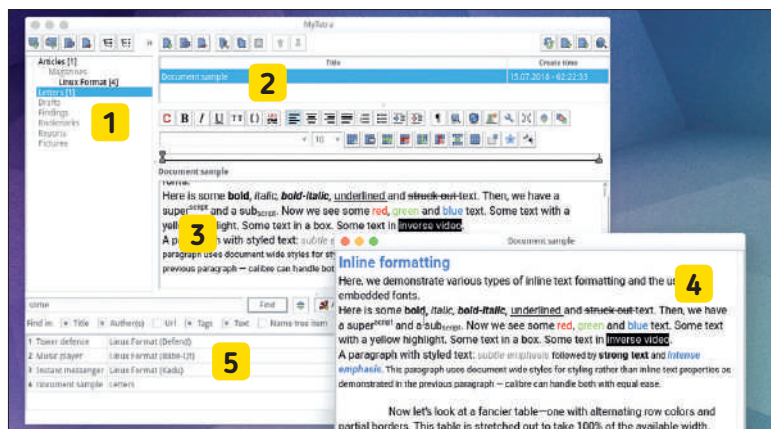
MyTetra позволяет выбирать шрифты и их размеры и стили, изменять отступы по линейке, работать с таблицами и ячейками, превращать текст в нумерованный список и еще много чего делать. Всё хранится в формате HTML, и вы даже можете редактировать код прямо в приложении, как если бы это был web-композитор!

MyTetra — приложение кросс-платформенное, да вдобавок способное послужить онлайн-энциклопедией, если вы загрузите файлы `mytetra.xml` и другие файлы данных на удаленный хост. Сайт проекта содержит инструкции о необходимых шагах с GitHub и Bitbucket, которые используются в качестве примеров хостов. Страница загрузки *MyTetra* предлагает статическую сборку для тех, кто не испытывает острой потребности компилировать код самостоятельно.



Похоже на Wordpad, но с расширенными функциями и управлением проектами.

Что где в MyTetra...



1 **Дерево ваших заметок**
 Независимо от объема вашей библиотеки *MyTetra*, вы всегда видите здесь всю ее структуру, со всеми ветками и дочерними элементами.

2 **Список заметок внутри элемента**
 Каждый элемент дерева может содержать более одной заметки. Весь список можно просмотреть в табличном виде.

3 **Документы с форматированием**
 Умения *MyTetra* простираются

далеко за рамки перебора шрифтов. Таблицы и изображения здесь тоже поддерживаются!

4 **Вы видите, что получаете**
 Каждая заметка — это HTML-документ, а здесь вы видите ее визуализированный предпросмотр.

5 **Поиск в базе знаний**
MyTetra усердно помещает все ваши тексты, заголовки и теги в поисковый индекс.

ОБРАЗОВАТЕЛЬНОЕ ПРИЛОЖЕНИЕ

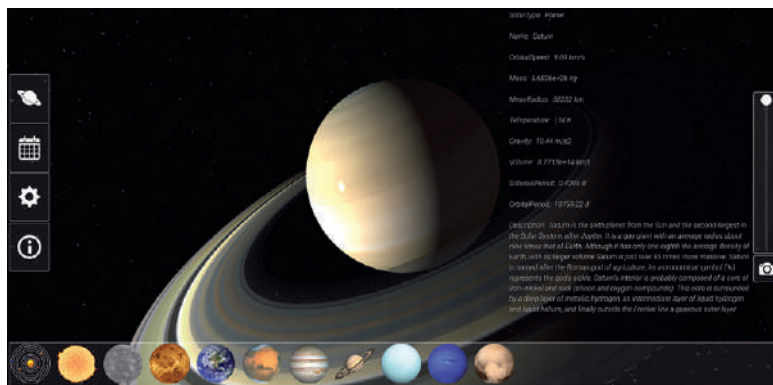
SolarSystem

Версия: Git

Сайт: <https://github.com/Instand/SolarSystem>

Греческий философ Сократ однажды сказал: «Человек должен подняться над Землей — в атмосферу и за ее пределы — ибо только так он полностью поймет мир, в котором живет». Только представьте: будь у Сократа этот 2400 лет назад библиотека Qt 3D (и работающий ПК), он мог бы легко обозреть всю Солнечную систему.

Итак, вот *SolarSystem*, приятное приложенище с непритязательным именем. Оно показывает нашу Солнечную систему в 3D и добавляет описания и факты из энциклопедии, благодаря чему еще полезнее. Мы протестировали *SolarSystem*, и это было очень весело, не только потому, что многие люди путают порядок планет, а *SolarSystem* тут помогает, но также и благодаря потрясающему внешнему виду каждого объекта. Очевидно, что не требуется уж очень много усилий для создания относительно простых фигур — в основном это сферы, изредка — с кругами вокруг, как Уран и Сатурн с их кольцами, но графика тем не менее замечательная. Вся Солнечная система анимирована, и вы можете видеть, как планеты вращаются вокруг своих осей и вокруг Солнца, как их орбиты соотносятся друг с другом, и т.д. Можно варьировать масштаб колесиком мыши, вращать планеты, перетаскивая с правой кнопкой мыши, а через добавочные панели



и меню добывать множество характеристик планет нашей Солнечной системы. Это очень познавательно и годится для обучения детей и взрослых.

Приложение имеет несколько простых элементов управления; наиболее полезный из них — слайдер в правой части экрана. Просто перетаскивайте его, чтобы изменить скорость анимации; затем, пользуясь кнопкой над слайдером, задайте множитель скорости для большей точности. Поскольку *SolarSystem* еще и поддерживает световые эффекты, вы можете приблизиться к Земле, немного замедлить работу и посмотреть, как день сменяет ночь в разных частях мира.

Чтобы запустить это красивое приложение, у вас должен быть мощный графический процессор и куча пакетов разработки Qt 3D. Соберите приложение с помощью небезызвестной последовательности `$ qmake-qt5 && make`.

Перетаскивайте ползунок, чтобы ускорить вращение Сатурна и увидеть красивое свечение его колец.

МУЗЫКАЛЬНЫЙ ПЛЕЙЕР

Elisa

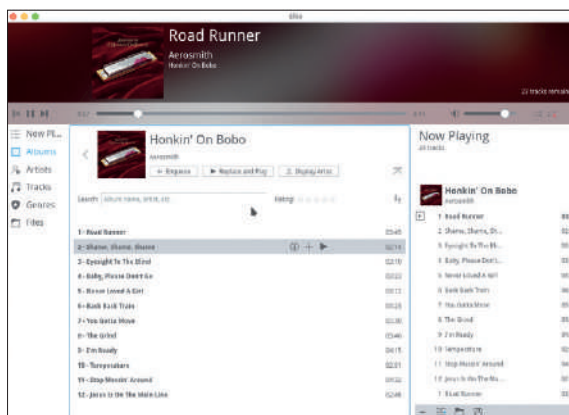
Версия: 0.2

Сайт: <https://github.com/KDE/elisa>

Еще в LXF225 мы рассмотрели первый выпуск музыкального плеера *Elisa*. Тогда он был примитивным и незавершенным, и хотя нам понравился его новый подход к визуальному дизайну, во многих других отношениях *Elisa* был явно не лучше других плееров. Новая версия 0.2 исправляет многие проблемы — и делает работу с *Elisa* намного приятнее. Помимо получения очень приятного значка «компакт-кассеты», появились некоторые улучшения в интерфейсе, а также несколько очень полезных функций.

Прежде всего, *Elisa* теперь позволяет воспроизводить любой файл из вашей файловой системы. Вроде и тривиально, но ранее поддерживались только треки из вашего каталога `~/Music`. Более того, теперь вы можете избежать общесистемного индексирующего *Baloo* и обойтись базой музыки только для *Elisa*. Оба способа имеют свои сильные стороны. Например, если *Baloo* пережевывал вашу местную музыку, *Elisa* в ноль секунд индексирует свою коллекцию без всяких лишних усилий.

Пользовательский опыт работы с *Elisa* может быть очень приятным, если вы хорошо организовали свою музыкальную библиотеку, а ваши треки имеют значащие и правильные теги. Это важно, потому что *Elisa* в качестве вида по умолчанию предлагает сетку



Наведите указатель мыши на альбом и отправьте его в плей-лист одним щелчком.

альбомов и автоматически находит обложку при запуске. Вид альбомов со всеми обложками смотрится очень хорошо, но тогда при первом запуске *Elisa* вам надо присутствовать онлайн. В прочих отношениях *Elisa* — сама простота. Здесь нет ни настроек, ни параметров; вместо этого предлагается всё необходимое, что средний пользователь может захотеть от музыкального плеера. Сюда включаются простые способы воспроизведения вашей любимой музыки по исполнителю, альбому, жанру и т.д., а также управление воспроизведением.

Мы довольно приятно провели время с последней версией плеера *Elisa*, и думаем, что это приложение больше всего понравится тем, кто любит простоту и доступность основных функций.

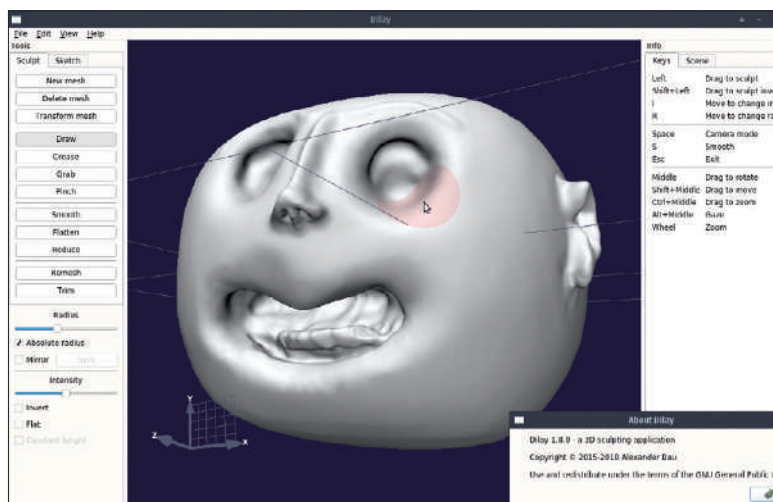
ПО 3D-МОДЕЛИРОВАНИЮ

Dilay

Версия: 1.9.0 Сайт: <https://abau.org/dilay>

Возможно, лучший вид производительных программ — такой, когда вы можете открыть программу (предположительно в первый раз), начать работать над чем-то и не замечать, как летит время. Именно это произошло с нами после открытия *Dilay*, многофункционального инструмента 3D-скульптуры. Приложение простое и четкое, но позволяет создавать сложные и забавные формы из шара сырого материала, который в *Dilay* называется «сетка [mesh]». Хотя вы не можете ощутить тактильно этот виртуальный материал, под воздействием инструментов Draw [Вытянуть], Crease [Согнуть], Grab [Взять] или Pinch [Ущипнуть] он напоминает натуральную глину.

Работать в *Dilay* очень просто, благодаря тщательно подобранному набору необходимых инструментов в левой части окна и советам с панели Info [Информация] с правой стороны. За какие-то секунды вы сообразите, как вращать, масштабировать и перетаскивать шар из цифровой глины. Вы можете смело лепить человеческую голову, или осьминога, или что в вашу собственную голову придет, и это требует только художественных, а не технических навыков. Сначала вы определяете основную форму, а затем подправляете ее части более тонкой отделкой. Не бойтесь, у *Dilay* есть инструменты для сглаживания



шероховатостей, вытягивания частей сетки и работы с пропорциями. Всего за несколько минут, проконсультировавшись с лаконичной базой знаний (для этого ознакомьтесь с руководством к приложению), вы станете цифровым скульптором! При этом порог входа у *Dilay* заметно ниже, чем у *Blender*, а значит, гораздо больше пользователей-любителей могут попытаться сформировать нечто прекрасное благодаря мощи открытого кода!

Dilay сохраняет всю вашу работу во внутреннем DLY-формате, но также поддерживает проприетарный формат Wavefront OBJ, так что потом вы можете использовать свою модель в более мощном приложении, таком как *Blender*, и оживить своего персонажа. Если вы не хотите компилировать *Dilay* из исходника, возьмите его как AppImage с <https://abau.org/dilay/download.html>.

Ваяние головы — непростая задача, как в реальной жизни, так и в *Dilay*!

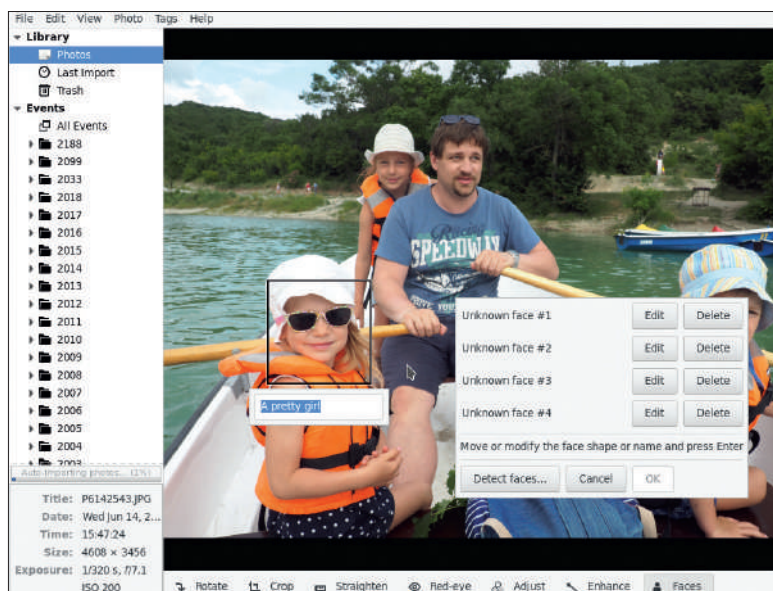
МЕНЕДЖЕР ФОТО

Shotwell

Версия: 0.29.3 Сайт: <https://gitlab.gnome.org/GNOME/shotwell>

Мы в последний раз рассматривали *Shotwell* в LXF237, но с тех пор его разработчики выкатили убойную функцию, игнорировать которую никак нельзя. Сборка *Shotwell* 0.29.3, который явно скоро станет 0.30, оснащена функцией распознавания лиц на основе OpenCV! *Digikam*, главный соперник *Shotwell*, обладает ею с 2011 г., хотя, по правде говоря, сами разработчики *Shotwell* кодировали ее еще в 2012 г. во время Summer of Code в Google. Теперь распознавание лица наконец-то здесь, и отлично работает прямо из коробки.

Открыв изображение в *Shotwell* и взглянув на нижнюю панель, вы обнаружите появление новой кнопки Faces [Лица]. Она выводит простой диалог, который автоматически определяет лица в изображении. Механизм обнаружения отлично работает для изображений с высоким разрешением, с улыбающимися взрослыми, когда на фото — лицо полностью. Если одно из этих условий не выполняется, в распознавании людей бывают странности, но польза в общем всё равно есть. Мы долго играли с обнаружением и распознаванием лиц в *Shotwell* (возможно, дольше, чем в некоторые простые игры!), и пришли к выводу, что в целом новая функция великолепна. Она достаточно разумна, чтобы правильно распознавать реальные лица, отсеивая рисунки и аниме,



поэтому, в итоге, единственное естественное ограничение (и это, очевидно, не ошибка *Shotwell* или OpenCV) заключается в том, что человека в детстве надо отмечать другим тегом, просто потому, что по мере роста внешний вид у всех меняется.

Грядущий *Shotwell* 0.30 также обзаведется дополнительной темной темой и улучшит поддержку не зависящих от дистрибутива пакетов Flatpak. Прямо сейчас вы можете получить последнюю ночную сборку *Shotwell* с <http://bit.ly/2Ng1Quk>.

Автоматическое распознавание лиц в большинстве случаев является точным.

СТЕРЕОПРОСМОТРИК

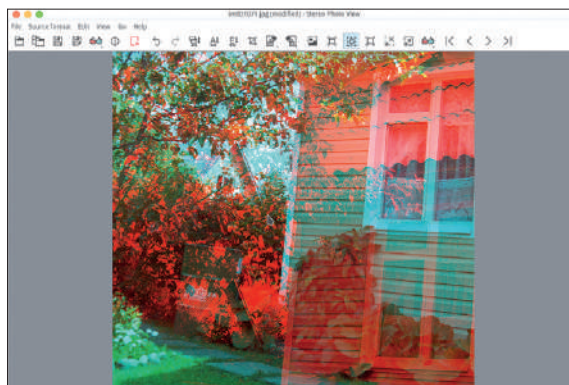
Stereophotoview

Версия: 1.11.0 Сайт: <https://bitbucket.org/av-mamzikov/stereophotoview>

Вот еще одно горячее приложение с открытым исходным кодом, с которым мы можем сэкономить фразу-другую на объяснении его имени. Что такое *Stereophotoview*, понятно сразу, так что давайте без дальнейших экивоков перейдем к системным требованиям.

Наряду с довольно стандартным набором заголовков библиотек *Qt* и *OpenCV* вам также понадобится подходящее оборудование для создания 3D-магии из обычных изображений. Да, это нишевое приложение, но его работа не зависит от того, есть ли у вас высококачественный 3D-дисплей, экран потребительского класса с пассивной поляризационной пленкой или обычный монитор. Что более важно — вам, скорее всего, понадобятся красные/синие анаглифные очки, чтобы увидеть стереоскопическую картинку.

Вы немало повеселитесь, исследуя *Stereophotoview*, особенно меню *Source format*, где можно играть с различными бинокулярными стереоэффектами. Приложение поддерживает OpenGL, и эффекты накладываются мгновенно, без каких-либо накладных расходов. Но приложение умеет гораздо больше — оно годится и как браузер изображений, графический редактор и даже видеоплеер.



Stereophotoview дает массу возможностей для настройки изображения, в том числе анаглифических, анаморфных, чередующихся видов и видов бок о бок с возможностью перемещения и поворота отдельных половинок. Специальные инструменты выравнивания из меню *Edit* [Редактировать] помогут настроить изображение под ваш взгляд. Изображения слева и справа можно редактировать по отдельности, создавая 3D-иллюзию; и, наконец, сохраните стереоизображение в формате MPO, который поддерживается многими устройствами 3D-просмотра.

Приложение очень умное, четкое и набитое множеством различных функций, которые вы обязательно захотите попробовать.

Для правильного просмотра изображения требуется специальное оборудование!

WEB-БРАУЗЕР

Browsh

Версия: 1.3.2 Сайт: <https://github.com/browsh-org/browsh>

Мы знаем, что самый известный текстовый веб-браузер — *Lynx*, и с учетом его почтенного возраста и сильного упора на клавиатурную навигацию его целевая аудитория ограничена. *Browsh* тоже попадает в категорию «текстовый интернет-браузер», но отличается от *Lynx* практически во всем. *Browsh* — это мрачная смесь графического рендеринга ASCII (см. также *Chafa* в LXF239/240) с тонким клиентом для просмотра веб-страниц. Общий дизайн одновременно причудливый и удивительный, и подразумевает запуск клиента и сервера как двух отдельных объектов, даже если вы запускаете *Browsh* на локальном хосте. Сам *Browsh* является фоновым процессом, который делегирует фактическую визуализацию веб-страницы в *Firefox*, работающий в режиме «без головы». *Browsh* применяет к отображаемой странице собственное специальное расширение веб-страницы, а затем передает псевдографический вид в командную строку через веб-сокеты. Всё это выглядит как пиксельный макет, потому что всё, кроме текста, изображено с помощью сплошных полублоков из UTF-8. Тем не менее, веб-страницы в *Browsh* вполне интерактивны, а значит, вы можете просматривать любой вид движущихся изображений внутри вашей сессии CLI, включая видео на YouTube!



BBC.com по-прежнему выглядит потрясающе в этом стильном режиме с низким разрешением.

На первый взгляд, *Browsh* скорее для забавы, чем для пользы. Он создает заметную дополнительную нагрузку на ЦПУ и для бесперебойной работы требует достойной производительности видео (и терминала с графическим ускорением). Но это дает вам еще одну причину подольше посидеть в вашем сеансе TTY, и если вы передаете работу безголовой части *Firefox* на удаленный сервер или решили запустить его как контейнер Docker, вы заодно сэкономите пропускную способность Интернета. Другими словами, запустите серверную консоль на одном хосте и получите доступ к ней от другого, чтобы резко увеличить скорость просмотра. В конце концов, это необычное и умное приложение первым позволяет просматривать веб-страницы в терминале с помощью мыши, как если бы это был настоящий графический интерфейс. Существует демо-сервер (`$ ssh brow.sh`) для опробования *Browsh* немедленно, но мы рекомендуем завести полноценную локальную установку.

ИГРА-СТРАТЕГИЯ

OpenRA

Версия: 20180307

Сайт: <https://github.com/OpenRA/OpenRA>

В HotPicks пришло время для настоящей борьбы за выживание цивилизации! Сегодня мы намерены сыграть в возрожденные стратегические игры с конца 1990-х гг., разработанные Westwood Studios, в частности, *Red Alert*, *Tiberian Dawn* и *Dune 2000*, без WINE или какой-либо эмуляции — прямо на Linux. *OpenRA* — ответ на вопрос о том, как это возможно. Это тщательно воссозданный движок с открытым исходным кодом, общий для всех трех игр. *OpenRA* приносит реальность *Red Alert*, как *Julius* для *Caesar III* (см. LXF233), но с еще более достоверными ощущениями. Причина здесь в том, что *OpenRA* поставляется со встроенным мастером, требующим всего несколько щелчков мыши для загрузки исходных активов выбранной вами игры. Для любопытных: это бесплатно и законно для конечного пользователя.

Игры очень похожи; основное отличие — уникальное произведение для каждого заглавия. Поэтому сюжет для *OpenRA* не слишком много значит, хотя вам может быть интересно узнать, что бесконечная война происходит на параллельной шкале времени, где нацистской Германии не было, но Советский Союз при Иосифе Сталине набрался мощи и захватил земли соседей.



Короче, игра подразумевает классическую конфронтацию Востока и Запада, и независимо от выбранной вами стороны ваши действия будут в основном такими же. Они включают создание мощной и производительной военной экономики, добычу золотой руды, обучение солдат и нападение на вашего врага, как только сможете (время здесь много значит). *OpenRA* — полноценная и высокоразвитая игра с умными ботами, многопользовательским режимом и богатым набором миссий, а также забавными трюками и персонажами.

Вы оцените высокое качество *OpenRA* во многих отношениях, включая хорошо продуманный интерфейс, где большинство действий дублируются нажатием клавиш, мощный сетевой режим, идеально подходящий для ожесточенных онлайн-столкновений с друзьями, а также общую скорость и плавность игры. *OpenRA* — отличная игра, достойная любви!

Время в *OpenRA* — это всё. Потеряв его, вы, скорее всего, безвременно погибнете.

ОБОРОНА БАШНИ

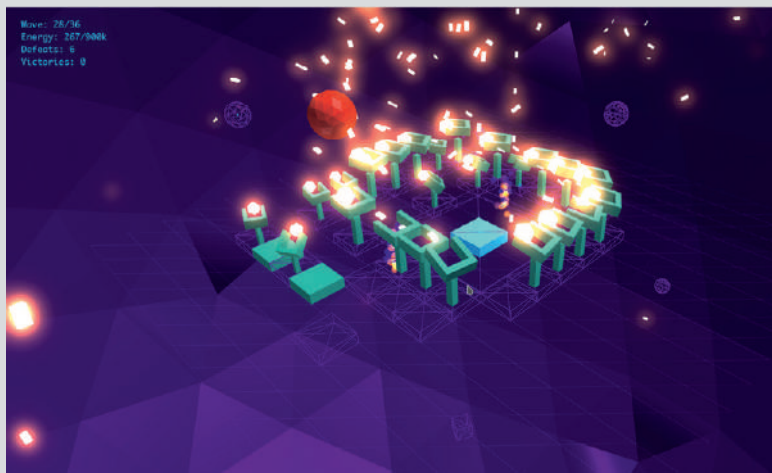
Defend

Версия: Git Сайт: <https://github.com/xtreemze/defend>

В последние годы жанр обороны башни превратился из классических двумерных игр в 3D-стрелялки. Мы обнаружили футуристическую и очень захватывающую новую игру такого рода под названием *Defend*. Это браузерная игра, созданная с помощью BabylonJS, многофункциональной платформы JavaScript для создания 3D-игр с HTML5, WebGL, WebVR и Web Audio. А значит, требуется только web-браузер с поддержкой WebGL, чтобы сыграть либо локально, либо по адресу <https://xtreemze.github.io/defend>.

В *Defend* вы попадаете в центр бесконечной 3D-плоскости с огороженной стеной территорией, которую вам надо защищать от врагов. Ваш обзор контролируется тремя камерами, которые вращаются на интерактивной орбите вокруг этой плоскости, где беспорядочно размещены башни. Ваши враги — синие сферические тела, которые падают с неба в незанятые районы и пытаются пробиться к центру.

Играть в *Defend* легко, если вы знакомы с *Neverball*, где вы управляете не движущимся объектом, а плоскостью, по которой он движется. *Defend* дает вам контроль над миром игры: вы можете щелкать левой кнопкой мыши и перемещать мышью, чтобы летать по пространству, и прокручивать колесико для масштабирования. Щелкните по своей территории,



чтобы нагромоздить участок стены, и щелкните еще раз, чтобы построить башню.

Вы можете просто окружить свою крепость стеной, но без башни вам не одолеть первой вражеской волны. Таких волн в общей сложности 36, с постоянно растущей трудностью. Чем больше вы стреляете и добиваетесь успеха, тем с большей энергией вы должны строить стены и башни, модернизировать башни и сопротивляться дальше. Куски стен склонны со временем исчезать, а вы обязаны поддерживать свои границы неразрывными. В результате сражения выглядят как фейерверки.

Defend сильно нагрузит WebGL вашего компьютера, поэтому, играя, вы также сможете примерно оценить, на что способна ваша видеосистема.

Может упасть частота кадров, но не крепость, которую вы защищаете от злых чужих сфер!

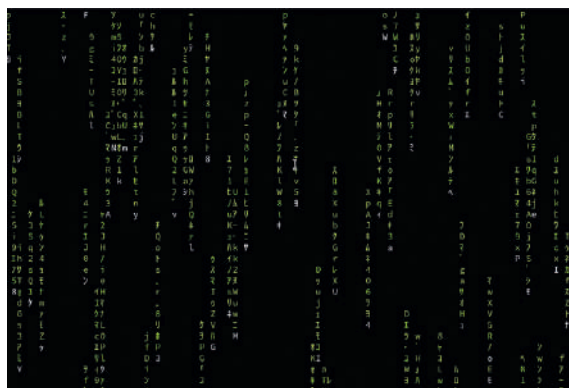
ЭМУЛЯТОР ТЕРМИНАЛА

Alacritty

Версия: Git Сайт: <https://github.com/jwilm/alacritty>

Наша первая попытка запустить *Alacritty* на Linux внутри *Virtualbox* не удалась, поскольку этот суперсовременный эмулятор терминала требует поддержки шейдеров OpenGL и для запуска нуждается в реальном аппаратном обеспечении — это, пожалуй, самое требовательное приложение для терминала, которое мы когда-либо видели. Команда разработчиков *Alacritty* утверждает, что это самое быстрое приложение в мире, благодаря графическому ускорению. Как тут было не попробовать!

Особенным *Alacritty* делает то, что это приложение сосредоточено вокруг одной убойной функции, тогда как во всех прочих отношениях это очень простой и базовый эмулятор терминала. Мы решили провести тесты и определить, какие именно сценарии применения с *Alacritty* будут полезны. Это было сложновато, так как во многих случаях *Alacritty* демонстрировал очень малое, если вообще какое-либо преимущество над *Konsole* или *Gnome Terminal*, но стоило нам попросить его пережевать нечто огромное (\$ `time cat huge.file`), как разница стала драматичной. *Alacritty* работал очень хорошо — почти в 1,5 раза быстрее, чем его конкуренты. Теоретически это означает, что вы можете просматривать



Опять же, неподвижная картинка не отражает, насколько плавной была эта анимация Матрицы в *Alacritty*.

очень большие объемы текстовых данных, таких как журналы или вывод приложений CLI, значительно быстрее, чем вы привыкли. Кроме того, мы не забыли великолепный web-браузер *Browsh* (см. стр. 103) и запустили его с *Alacritty*. Традиционный статический контент вел себя без какой-либо разницы, но фильм был более плавным. Если вы считаете, что от *Browsh* вам не только забава, но и польза, запустите его внутри терминала *Alacritty* для лучшей реакции и более плавного воспроизведения видео.

Другая область, где *Alacritty* блистает — мультиплексирование терминала. В *Alacritty* нет вкладок, но можно запустить *Tmux* или аналогичную программу для деления одного окна терминала на сетку из нескольких сеансов внутри одного окна. Обычно это создает заметную нагрузку на движок рендеринга эмулятора, но *Alacritty* в таких случаях выглядит лучше.

Если вы чувствуете, что ваше текущее приложение терминала тормозит, взгляните на *Alacritty*.

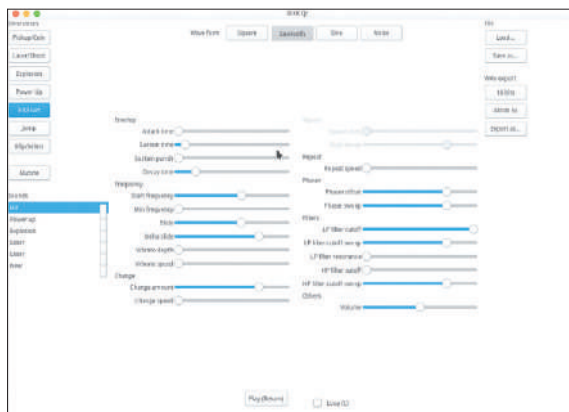
ГЕНЕРАТОР ЗВУКА

SFXR-Qt

Версия: 1.1.0 Сайт: <https://github.com/agateau/sfxr-qt>

Сколько функций может предложить одноэкранное приложение без контекстных меню, вкладок и настроек? Довольно много, если это *SFXR-Qt*, универсальный генератор звуковых эффектов для разработчиков игр. Это продолжение приложения-оригинала *SFXR* с 2007 г., которое всё еще можно собрать и использовать (<https://github.com/grimfang4/sfxr>). Идея заключалась в том, что разработчики игр были очень напряженными по графику, и программистам не хватало времени на просмотр тысяч бесплатных образцов звуков на соответствующих сайтах. Вместо этого они могли использовать это крошечное приложение, чтобы самим создавать звуки и экономить драгоценное время.

SFXR-Qt имеет в общей сложности семь звуковых генераторов, таких как 'coin pickup [звон монеты]', 'laser [лазер]', 'explosion [взрыв]' и другие, все с говорящими именами, пригодных для игр-аркад, стрелялок, платформеров и подобных жанров. Основной экран, однако, является полем для целых 23 ползунков, которые позволяют контролировать почти все аспекты звучания. Перетаскивание ползунков очень познавательно с точки зрения получения информации о том, как частота, резонанс, изменение или удержание влияют на результат, так что на это занятие легко подсесть.

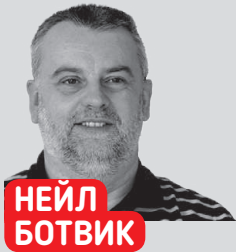


Этот продукт создан для творчества. Дайте вашему проекту больше звуковых вибраций!

В верхней части *SFXR-Qt* имеются переключатели сигналов. Четыре доступных варианта отражают вид формы сигнала вашего звука в редакторах, таких как *Audacity* и *KWave*. Разумеется, разница тоже слышна. Может показаться, что прежде чем удастся получить с *SFXR-Qt* приличные результаты, надо приложить время и навыки, но как только вы возьметесь за приложение, вы увидите, что рабочий процесс оптимизирован благодаря встроенной рандомизации. Всё, что вам надо делать — это многократно нажимать кнопку желаемого генератора и получать разнообразные звуки с различным шагом и частотой. Ползунки пляшут в такт с вашим созданием звуков, и вы можете настроить каждый сэмпл. Удовлетворившись результатом и получив самый сладкозвучный (и настроенный) сэмпл, можете экспортировать его как WAV-файл и использовать везде, где хотите, а не только в играх. **LXF**

На диске

Дистрибутивы, приложения, игры, книги и всякое-разное...



В мире дистрибутивов Linux новых очень мало. Взгляните, например, на DVD этого месяца. Kali Linux — бывший BackTrack Linux, BunsenLabs восстал из непаля CrunchBang, Feren OS построен на Linux Mint, а тот — на Ubuntu; последний же, в свою очередь, берет за основу Debian.

Зато в каждом из этих дистрибутивов масса новых элементов, да и вне их такого хватает. Эти дистрибутивы обычно не разрабатываются комитетом, определяющим, чего хотят и за что будут платить пользователи, и тестирующим свои идеи на фокус-группах. Они родились потому, что какие-то разработчики подумали: «Разве не здорово было бы это сделать?». Иногда пользователи соглашаются — это действительно круто, и идея овладевает массами.

Такое было бы невозможным, если бы не модель лицензирования, позволяющая вам брать продукт других, вносить изменения и делиться этим. Можно утверждать, что от свободных лицензий зависит эволюция. Если бы кроманьонцы выпускались под проприетарной лицензией, человеческой расы не было бы! Многие компании, исходя из своей узко коммерческой точки зрения, называют лицензирование с открытым исходным кодом странно. Но такие свободы — основа самой жизни. Глядя на Книгу Месяца, я только тихо надеюсь, что Человечество 2.0 не окажется киборгами, хотя сам заменил в этом году уже второй сустав!

БЫСТРЫЙ ДИСТРИБУТИВ

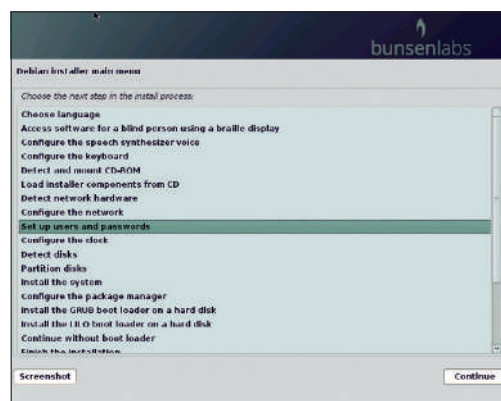
BunsenLabs Helium

32-БИТНЫЙ

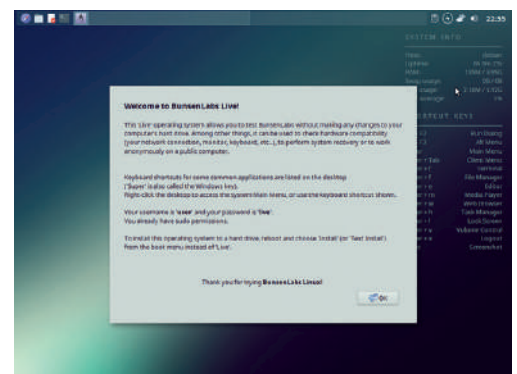
Жил да был ультралегкий дистрибутив CrunchBang — превосходный выбор для желающих иметь минимальный рабочий стол, который не путается под ногами и просто позволяет заниматься делом... К сожалению, несколько лет назад CrunchBang повторил историю многих хороших вещей, почив в бозе; но в мире открытого исходного кода ничто не умирает. Его кончина привела к созданию двух дистрибутивов: ArchBang переосмыслил CrunchBang, избрав за свою базу Arch Linux, а BunsenLabs — более традиционное ответвление.

Построенный на базе Debian, BunsenLabs использует оконный менеджер *OpenBox*, а не среду, склонную к алчности до ресурсов. Результат — быстрый и отзывчивый рабочий стол, оставляющий больше тактов процессора для производительных программ, которые вы собираетесь запустить.

Установка BunsenLabs на компьютер, где ранее был рабочий стол Gnome или KDE, похож на аппаратное обновление — настолько он отзывчивее. Так что это хороший выбор для более старого оборудования (по данной причине мы и включили на диск 32-битную версию), но и современные компьютеры намного ускоряет.



Если вы хотите установить BunsenLabs или Kali Linux, запишите ISO-файл на DVD или USB-брелок и загрузитесь с него.



Пока не совсем радуга, но рабочий стол BunsenLabs (немного) менее серый, чем раньше.

Насчет его обновленного внешнего вида мнения разделились. По мне, старый пастельный монохромный вид был достаточно унылым, и намек на цвет на рабочем столе превращает его в более современный, не жертвуя производительностью. Но, похоже, некоторые пользователи недовольны, если их экран не выглядит как продукт времен 1980-х гг.

Это Live-дистрибутив, загружаемый прямо с DVD — чтобы вы могли его попробовать. Как уже упоминалось, BunsenLabs основан на Debian, и хотя система Live Debian прекрасно сработалась с нашим методом загрузки дистрибутивов с **LXFDVD**, про ее установщик такого утверждать нельзя; поэтому мы удалили установку из загрузочного меню, чтобы уберечь вас от лишних страданий. Но не спешите расстраиваться — установить BunsenLabs с нашего DVD всё равно можно; вам придется только записать файл ISO-образа из каталога **BunsenLabs** на DVD или USB-брелок (все необходимые данные для этого приводятся в FAQ про DVD), а затем загрузиться с него.

На экране вы увидите собственное загрузочное меню BunsenLabs с опциями текстовой и графической установки.

Аутентификационная информация: имя пользователя — **user**, пароль — **live**.

ВНИМАНИЕ! ЭТО ВАЖНО!

ПОРЧЕННЫЕ ДИСКИ: В маловероятном случае какого-либо дефекта вашего LXFDVD обращайтесь, пожалуйста, по адресу disks@linuxformat.ru или телефону +7 (812) 309-0686.

ДИСТРИБУТИВ – ДРУГ НОВИЧКОВ

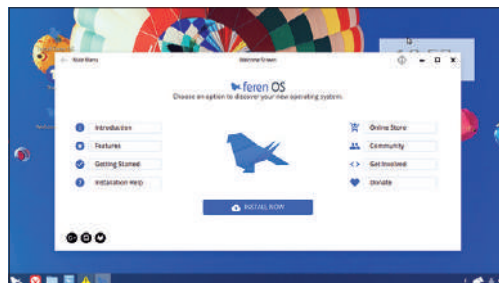
Feren OS 2018.07

64-БИТНЫЙ

Существует немало дистрибутивов, которые стремятся быть легкими в использовании для новичков в мире Linux. Некоторые стараются выглядеть как другая ОС, пытаясь облегчить переход. Но Linux не Windows или MacOS, и никакое количество тем не сделает его ими, да это и не особенно полезно.

Feren OS основан на Linux Mint и использует рабочий стол Cinnamon, поэтому он уже прост в использовании; но его даже улучшили, чтобы сделать еще более доступным, хотя и он не притворяется тем, чем отнюдь не является — см. наш обзор на стр. 12!

Аутентификационная информация: имя пользователя **mint**, пароль **net**.



Mint просто стал еще дружелюбнее к новичкам, когда компания Feren OS взяла дистрибутив на себя, чтобы помочь пользователям уйти от проприетарных ОС.

ВОРОТА К ИНСТРУМЕНТАМ БЕЗОПАСНОСТИ

Kali Linux Light 2018.2

64-БИТНЫЙ

3 здесь мы мало что можем добавить о Kali Linux, поскольку об этом дистрибутиве подробно рассказывалось в статье по теме номера.

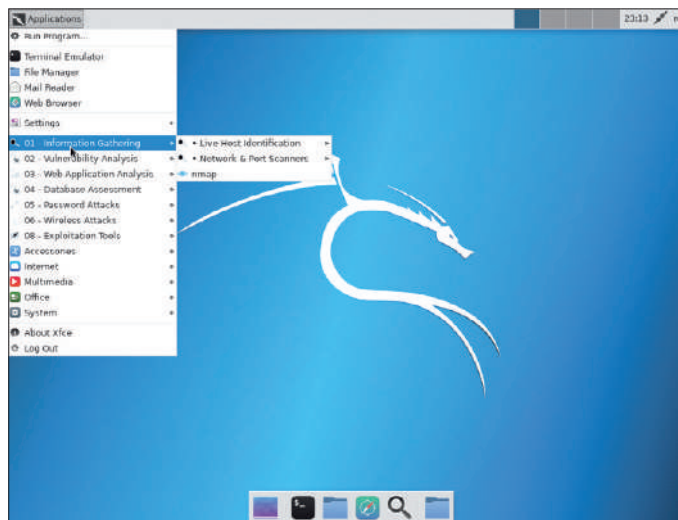
Kali Linux — ответвление более старого и уважаемого BackTrack Linux, ставшее одним из ведущих дистрибутивов для обеспечения безопасности. Полная версия Kali Linux содержит множество инструментов безопасности, аудита и криминалистической экспертизы. В итоге, это дистрибутив увесистый — слишком увесистый, чтобы

попасть на наш DVD, если мы хотим включить и другие дистрибутивы тоже. Так что предлагаем вам облегченную версию, Kali Light.

Kali Light содержит самые основные инструменты, и немаловажно то, что остальные инструменты находятся в репозиториях Kali. А следовательно, всё, чего вам не хватает, устанавливается несколькими щелчками мыши (или быстрым вводом команды **apt-get**, если менеджеры пакетов GUI ниже вашего достоинства).

По части установки, Kali страдает от тех же ограничений, что и BunsenLabs, когда дело доходит до запуска установщика с DVD *Linux Format*, и решение тут предлагается такое же.

Аутентификационная информация: имя пользователя **root**, пароль **toor**.



Ваша мечта — отгонять хакеров или стать одним из них (белым, конечно)? Дистрибутив Kali Linux — частый выбор в подобных случаях.

» НОВИЧОК В LINUX? НАЧНИТЕ ОТКУДА!

Мы ответим на ваши вопросы, установите DVD и откройте [index.html](#).

- Что такое Linux? Как его установить?
- Есть ли эквивалент MS Office?
- Для чего нужна командная строка?

**СКАЧАЙТЕ
СВОИ ОБРАЗЫ
DVD!**

Найдите код и образы DVD
на [www.linuxformat.com/
archives](http://www.linuxformat.com/archives)

» И ЕЩЕ!

**BITS AND BOLTS:
THE PAST, PRESENT,
AND FUTURE OF OPEN
HARDWARE**

**БИТЫ И БОЛТЫ:
ПРОШЛОЕ,
НАСТОЯЩЕЕ
И БУДУЩЕЕ
ОТКРЫТОГО
ОБОРУДОВАНИЯ**

Книга
месяца

Как философия открытого исходного кода работает с ПО, понять легко. А в этой книге обсуждается применение данной философии к аппаратуре, и изучаются реальные примеры. Когда технологии, подобные 3D-печати, становятся все доступнее, надо ожидать взрыва открытого оборудования (нет, мы не про 3D-печатные бомбы) — вот ваш шанс взять прогресс под контроль.

КНИЖНАЯ ПОЛКА

- **Расширенное руководство по скриптам Bash** Улучшите написание скриптов.
- **Руководство по Bash для начинающих** Осваивайте скрипты *Bash*.
- **Скрипты Bourne Shell** Начиная осваивать скрипты оболочки.
- **Собор и Базар** Классический текст Эрика С. Реймонда [Eric S Raymond] о преимуществах открытой разработки.
- **Справочник админа Debian** Руководство для сисадминов.
- **Нырните в Python** Все, что вам надо знать.
- **Введение в Linux** Удобное руководство с массой подсказок для новичков в Linux.
- **Словарь Linux** Linux от А до Я.
- **Linux Kernel in a Nutshell [Ядро Linux в двух словах]** Книга великого магистра ядра Грегa Кроа-Хартмана [Greg Kroah-Hartman].
- **Руководство сисадмина Linux** Контролируйте свою систему.
- **Обзор инструментов** Полный обзор инструментов GNU.
- **Справочник GNU Emacs** Шестсот страниц информации!
- **Создание программ с открытым кодом** Всё, что вам надо знать.
- **Программирование с основ** Сделайте первые шаги.

ПРОШЛЫЕ НОМЕРА » ПРОПУСТИЛИ?

LXF239/240

Август 2018

- Проходите, облака Запустим свои
- Резервные копии Чтобы с вашими данными не случилось страшное
- Плюнет, поцелует? Машинный анализ эмоций
- Аудио по всему дому Дешево и сердито



LXFDVD

Devuan, Linux Mint с Cinnamon и Mate, openSUSE Leap, Q4OS, Sparky Linux LXQt, Trisquel Mini, 12 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_239-240/
PDF-версия: shop.linuxformat.ru/elxf_239-240/

LXF238

Июль 2018

- Сделаем сами Рабочий стол мечты
- Редактируем изображения Как создать фотомонтаж
- Децентрализуем Интернет IPFS — межпланетная файловая система
- Гадаем на ромашках Raspberry Pi различает цветы!



LXFDVD

Fedora, NixOS, Pinguy, Salix, Ubuntu, Void OS, 12 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_238/
PDF-версия: shop.linuxformat.ru/elxf_238/

LXF237

Июнь 2018

- Бионический Бобер Ubuntu долгосрочный
- CMS Кому доверить управление контентом
- Коты, кубиты и кванты Передний край технологий
- Запускаем мультисайт Удобная иерархия



LXFDVD

Linux Lite, Pinguy, Rosa, Sculpt, Slax, Ubuntu, Zorin, 12 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_237/
PDF-версия: shop.linuxformat.ru/elxf_237/

LXF236

Май 2018

- Роботы на Pi Терминатор своими руками
- Веселый архивариус Управляем хранилищем
- Сайтострой Стилизуем под кого-то
- Raspberry Pi греет Отопление на контроле



LXFDVD

FArch, ArchLabs, Arch Linux ARM, Manjaro, подборка для NAS, 12 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_236/
PDF-версия: shop.linuxformat.ru/elxf_236/

LXF235

Апрель 2018

- Отгоним шпионов Все виды самозащиты
- Как стать телеведущим Видеоконференции
- Linux — фотография Шедевры и их хранение
- Открываем лавочку Интернет-магазин на WordPress



LXFDVD

Anarchy, LXLE, Parrot Security OS, Solus, Tails, 12 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_235/
PDF-версия: shop.linuxformat.ru/elxf_235/

LXF234

Март 2018

- Экспресс-курс Linux 60 минут — и вы в дамках
- Не только навигация Выберем менеджер файлов
- Терминал под музыку Работать будет весело
- В порту по случаю Нарастиваем энтропию



LXFDVD

AV Linux, Bodhi, elementary OS, Kali Light, ROSA, 11 книг о Linux (на английском языке), горячие новинки и прочее...

Печатная версия: shop.linuxformat.ru/lxf_234/
PDF-версия: shop.linuxformat.ru/elxf_234/

Закажите на сайте www.linuxformat.ru

Подпишитесь на печатную версию журнала на www.linuxformat.ru/subscribe или www.linuxcenter.ru, и получите электронную версию в подарок!

Телефон: (812) 309-0686

Linux Format ВКонтакте: vk.com/linuxform

Содержание

Страница 0

ДИСТРИБУТИВЫ

BunsenLabs (32-битный)
Дистрибутив на базе Debian Stable, с легковесным и легко настраиваемым рабочим столом Openbox. Продолжение от сообщества дистрибутива OpenBang Linux.

Feren OS 2018 (64-битный)

Настольный дистрибутив Linux на базе Mint с рабочим столом Cinnamon, включает уровень совместимости Wine для запуска приложений Windows, а также ПО для обработки текстов, в основном совместимое с Microsoft Office, и веб-браузер Uivald.

Kali Linux 2018 (64-битный)

Дистрибутив на базе Debian с набором средств для тестирования на вторжение и компьютерной криминалистики. Предлагает выбор из четырех популярных настольных сред и плавное обновление до более новых версий. Поддерживает архитектуру x86 и ARM.

НОТРИСКС

Alacritty (git) Эмулятор терминала
<https://github.com/jandedevries/alacritty>

Bravest 1.3.2 Веб-браузер

<https://github.com/bravest-001/bravest>

Defend (git) Игра — оборона башни
<https://github.com/klavente/Defend>

Dlib 19.0 ПО 3D-моделирования
<https://dlib.ai/dlib/>

Elisa 0.2 Музыкальный плеер
<https://github.com/KDE/elisa>

Mutella 1.4.3.27 Менеджер персональной информации
<https://github.com/klavente/mutella>

OpenRA 20180307 Игра-стратегия
<https://github.com/OpenRA/OpenRA>

SFXR-01 1.1.0 Генератор звука
<https://github.com/adafruit/sfxr-01>

Shdwel 0.2.3 Менеджер фото
<https://github.com/0x0000/shdwel>

SolarSystem (git) Образовательное приложение
<https://github.com/instanoid/SolarSystem>

StegorhinoView 1.1.0 Стереоскопический
<https://github.com/dm-papajko/stegorhinoView>

УЧЕБНИКИ

Swift

Описание на обороте >

>> ИНФОРМАЦИЯ О ДИСКЕ

ЧТО-ТО ПОТЕРЯЛИ?

Часто случается, что новые программы зависят от других программных продуктов, которые могут не входить в текущую версию вашего дистрибутива Linux.

Мы стараемся предоставить вам как можно больше важных вспомогательных файлов. В большинстве случаев последние версии библиотек и другие пакеты мы включаем в каталог **Essentials** [Главное] на прилагаемом диске. И 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 всё же прилагаемый к журналу диск поврежден, пожалуйста, свяжитесь с нашей службой поддержки по электронной почте: disks@linuxformat.ru



ОРУДИЯ ХАКЕРА

ПОЛНОЕ ОБНОВЛЕНИЕ

Кали Linux 2018

ЛЕГКАЯ ВЕРСИЯ

ОС для хакеров-профессионалов

- Все необходимые инструменты
- Анализ уязвимостей
- Компьютерная криминалистика
- Сетевые атаки

64-БИТНЫЙ



32-БИТНЫЙ



64-БИТНЫЙ



Классический дистрибутив Linux — легковес, отлично настраиваемый

BunsenLabs

Легко работать с функциями Windows — они встроены

Feren OS 2018

Содержание

ПОМОЩЬ

ЧаВо (FAQ) по DVD

Ответы

Rule Знаменитый справочник

ГЛАВНОЕ

Grub

Mentest86+

ДОКУМЕНТАЦИЯ: 17 книг о Linux (на английском языке)

Advanced Bash Scripting Guide. Подробное руководство

по программированию на Bash

Bash Guide for Beginners. Руководство по Bash

для начинающих

Bits and bolts. Открытый исходник — это не только

про ПО

Bourne Shell Scripting. Начальное руководство по про-

граммированию на Bash

The Cathedral and the Bazaar. Классический текст

Эрика Реймонда [Eric S Raymond] «Сбор и базар»,

о преимуществах открытой разработки

The Debian Administrator's Handbook. Руководство

администратора, написанное разработчиками Debian

Drive Into Python. Учебник по программированию

на Python

Intro to Linux. Начальное руководство по Linux

Linux Dictionary. Словарь Linux, объясняющий специфич-

ескую терминологию

Linux Kernel in a Nutshell. Описание ядра Linux,

созданное одним из его выдающихся разработчиков —

Грегом Кроа-Хартманом [Greg Kroah-Hartman]

Producing OSS. Как выпускается СПО

Programming from the Ground Up. Программирование

с самых основ

Python 3.7.0. Официальная документация

System Administrators Guide. Руководство по базовому

администрированию Linux

GNU Tools Summary. Руководство по работе в командной

строке и обзор основных утилит GNU

Ultimate Linux Newbie Guide. Руководство по Linux

для новичков, но не только

Understanding Open Source And Free Software Licensing

Как лицензируется открытое и свободное ПО

Пожалуйста, перед использованием данного диска ознакомьтесь с инструкцией, опубликованной в журнале на стр. 109.

Страница 2

ДИСТРИБУТИВЫ

GhostBSD (64-битный)

Дружелюбная к пользователю настольная ОС на базе

TrueOS, основанной на ветке разработки FreeBSD.

Цель проекта — создать удобное и знакомое рабочее

пространство, пригодное для дома и офиса или для

спасения данных. GhostBSD поддерживает множество

популярных легковых рабочих столов, включая Mate

и Xfce. Также предоставляется система управления

пакетами FreeBSD, LibreOffice, LibreCAD, среда

разработки Eclipse/Anjuta для C++/Java, JavaScript,

Java и Python.

MX Linux (64-битный)

Настольный дистрибутив с рабочим столом

Xfce, на базе «стабильной» ветки Debian. В этой

совместной разработке сообществ antiX и MEPIS Linux

элегантность и эффективность сочетается с простой

конфигурацией. Высокой стабильностью и высокой

производительностью, при среднем объеме.

Все дистрибутивы представлены ISO-образом, который можно записать на одноплатный носитель, и загрузить в live-режиме прямо с LXFDVD.

У всех присутствует возможность установки на жесткий диск.

Настольный дистрибутив предназначен для использования на всех стадиях производства, однако, как и в случае с любым другим ПО, мы рекомендуем вам использовать аппаратный сканер. Мы также рекомендуем всегда иметь под рукой актуальную резервную копию данных вашего жесткого диска. К сожалению, редакция Linux Format не в состоянии принимать на себя ответственность за любые повреждения, разрушения или иные убытки, которые могут повлечь за собой использование этого DVD, представленных на нем программ или данных. Прежде чем устанавливать какое-либо ПО на компьютер, пожалуйста, скачайте программу и проверьте ее с помощью антивируса.

Тираж изданий ООО «Марком», 188652, Россия, Ленинградская область, Всеволожский р-н, дер. Юрки, Школьная ул., 7-а. Лицензия ИДТР ВАО N 77-03.

» СОЗДАНИЕ УСТАНОВОЧНЫХ ДИСКОВ ПРИ ПОМОЩИ CDRECORD

Самый быстрый способ записать ISO-образ на чистую матрицу — это обратиться к программе *cdrecord*. Для всех перечисленных ниже действий потребуются права суперпользователя-root. Сначала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdrecord -scanbus
```

После этого на экране терминала должен отобразиться список устройств, подключенных к вашей системе. SCSI-адрес каждого устройства представляет собой три числа в левой колонке — например, 0,3,0. Теперь вы можете с легкостью записать образ на диск: *cdrecord dev=0,3,0 -v /путь к образу/image.iso*

Чтобы упростить дальнейшее использование *cdrecord*, сохраните некоторые настройки в файле */etc/default/cdrecord*. Добавьте по одной строке для каждого устройства записи (скорее всего, в вашей системе присутствует только одно такое устройство): *Plextor=0,3,0 12 16M*

Первое слово в этой строке — метка; затем после адреса SCSI-устройства вы должны указать скорость и объем буфера. Теперь можете заменить SCSI-адрес в командной строке на выбранную вами метку. Всё будет еще проще, если вы добавите следующее:

```
CDR_DEVICE=Plextor
```

Для записи ISO-образа вам осталось набрать команду

```
cdrecord -v /path/to/image.iso
```

Если вы не принадлежите к любителям командной строки, в таком случае вам придет на помощь утилита *gcombust*. Запустите ее от имени root и выберите вкладку *Burn* и *ISO 9660 Image* в верхней части окна. Введите путь к образу, который вы хотите записать на диск, и смело нажимайте на *Combust*! Пока ваш образ прожигается на диск, можете выпить чашечку кофе.

ДРУГАЯ ОС?

Использовать Linux для записи компакт-диска не обязательно. Программы вроде *cdrecord* просто переносят двоичные данные на чистую матрицу. Все необходимые файлы уже включены в ISO-образ, который распознается любой операционной системой, будь то Linux, Windows, Mac OS X или AmigaOS.

НЕТ УСТРОЙСТВА ДЛЯ ЗАПИСИ ДИСКОВ?

Если у вас нет устройства, с помощью которого можно было бы записать образ на диск, можно найти кого-нибудь друга или организацию, у кого есть компьютер с дисководом, и прожечь диск у них. Опять-таки, вам подойдет любая операционная система, способная распознать пишущий привод (см. выше).

Некоторые дистрибутивы умеют монтировать образы дисков и выполнять сетевую установку или даже установку с раздела жесткого диска. Конкретные методы, конечно, зависят от дистрибутива. За дополнительной информацией обращайтесь на сайт разработчика дистрибутива.

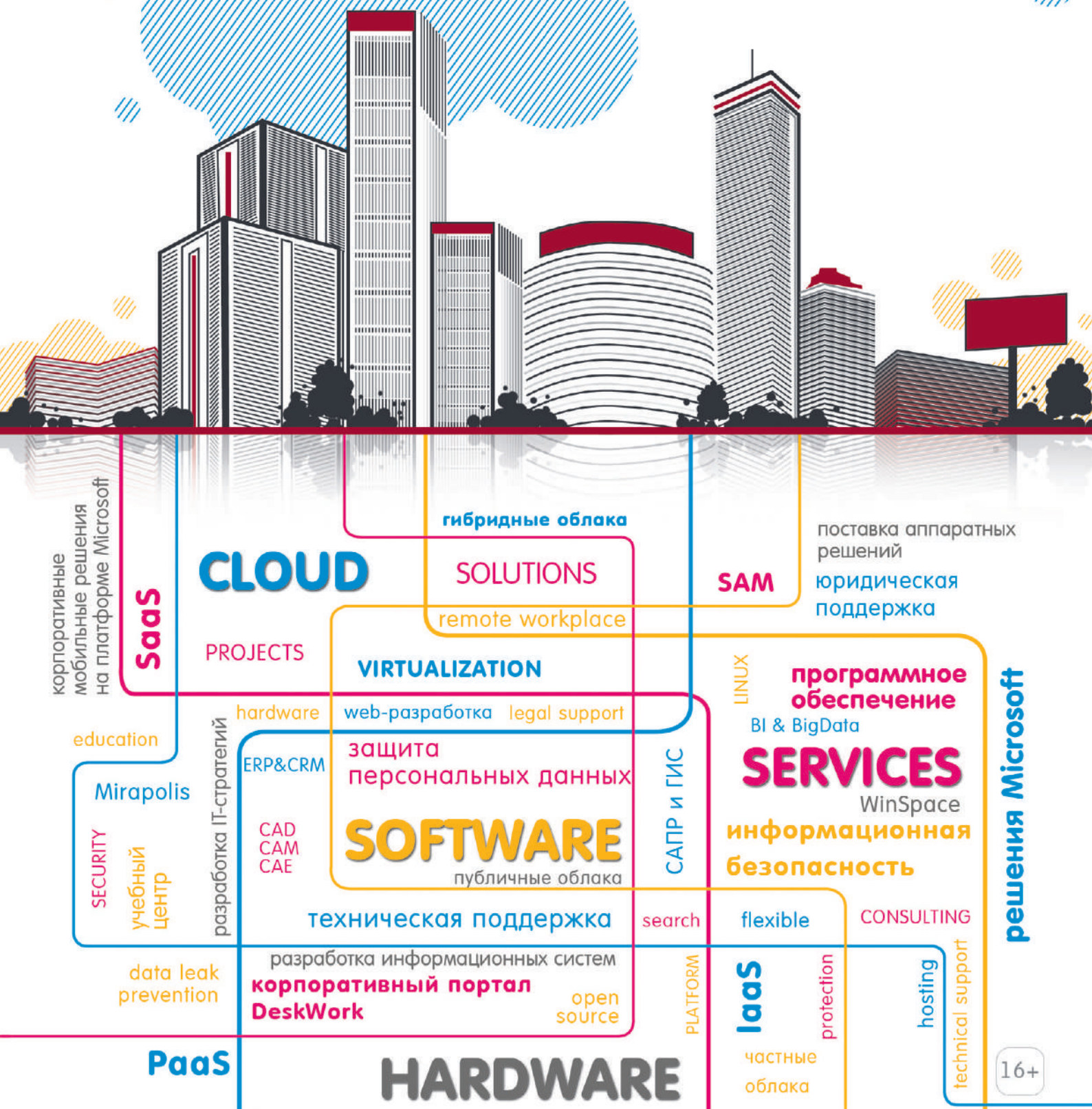
softline®

Cloud Software Hardware Services

20+

Years in IT

IT-архитектура вашего бизнеса





ЛУЧШИЕ ИЗ ЛУЧШИХ!

Какой, по-вашему, лучший дистрибутив 2018 года?
Перечисляем 20 кандидатов и помогаем собрать свой.

Игры с открытым кодом

Когда игра умирает, сообщество берет ее код и собирает открытый движок, чтобы ее воскресить.

Секрет вечной молодости

Установите себе ядро, которое актуально всегда, и мы подберем вам лучший самообновляющийся дистрибутив.

Linux на ноутбуке

Разберемся, как установить Linux на одном из последних ноутбуков Lenovo и настроить его до идеального совершенства.

Сладкая ловушка

Мы, как тот Винни-Пух, тянемся за сладостями. Пренебрежительно, когда сласти оказываются коварной приманкой...

LINUX FORMAT

Главное в мире Linux

Журнал зарегистрирован Федеральной службой по надзору за соблюдением законодательства в сфере массовых коммуникаций и охране культурного наследия ПИ № ФС77-21973 от 14 сентября 2005 года. Выходит ежемесячно. Тираж печатной версии 1000 экз., распространение электронной версии 30 000 экз.

РЕДАКЦИЯ РУССКОЯЗЫЧНОЙ ВЕРСИИ

Главный редактор

Кирилл Степанов info@linuxformat.ru

Литературный и выпускающий редактор

Елена Толстякова

Переводчики

Галина Итальянская, Даниил Кривошеин, Светлана Кривошеина, Екатерина Семенова, Валерий Смирнов, Елена Толстякова, Ирина Шулакова

Редактор диска

Григорий Огилько

Верстка, допечатная подготовка

Сергей Рогожников

Технический директор

Андрей Смирнов

Директор по рекламе

Владимир Савельев advert@linuxformat.ru

Генеральный директор

Павел Фролов

Учредители

Частные лица

Издатель

ООО «Линукс Формат»

Отпечатано в типографии ООО «Типографский комплекс „Девиз“» 195027, Санкт-Петербург, Якорная ул., 10, корп. 2, литер А, пом. 44

РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ

Редактор Нейл Мор (Neil Mohr) neil.mohr@futurenet.com

Мистер Робот Джонни Бидвелл (Jonni Bidwell)

jonni.bidwell@futurenet.com

Художественный редактор Эфраин Эрнандес-Мендоса

Efrain.Hernandez-Mendoza@futurenet.com

Выпускающий редактор Клифф Хоуп (Cliff Hope)

clifford.hope@futurenet.com

Директор редакции Грэм Барлоу (Graham Barlow)

Старший художественный редактор Джо Гулливер (Jo Gulliver)

ПОДГОТОВКА МАТЕРИАЛОВ

Майк Бедфорд (Mike Bedford), Джонни Бидвелл (Jonni Bidwell), Нейл Ботвик (Neil Botwick), Кент Ельчук (Kent Elchuk), Дэн Фрост (Dan Frost), Джон Найт (John Knight), Джереми Лэйд (Jeremy Laird), Ник Пирс (Nick Peers), Омри Петитт (Omri Petitt), Шашанк Шарма (Shashank Sharma), Валентин Синицын (Valentine Sinitsyn), Томас Стратфорд (Thomas Stratford), Александр Толстой (Alexander Tolstoy), Михалис Цукалос (Mihalis Tsoukalos), Евгений Балдин, Евгений Голышев, Андрей Гондаренков, Алексей Федорчук

Иллюстрации Шейн Коллиндж (Shane Collinge)

Иллюстрация на обложке magictorch.com

КОНТАКТНАЯ ИНФОРМАЦИЯ

UK: Linux Format, 30 Monmouth Street, Bath BA1 2BW

Тел. +44 (0) 344 848 2852, email: linuxformat@myfavouritemagazines.co.uk

РОССИЯ: Санкт-Петербург, пр. Медиков, 5, корп. 7

Тел. +7 (812) 309-0686, e-mail: info@linuxformat.ru

По вопросам сотрудничества, партнерства, оптовых закупок: partner@linuxcenter.ru

Авторские права: статьи, переведенные из английского издания Linux Format, являются собственностью или лицензированы Future Publishing Ltd (Future plc group company). Все права зарегистрированы. Никакая часть данного журнала не может быть повторно опубликована без письменного разрешения издателя.

Все письма, независимо от способа отправки, считаются предназначенными для публикации, если иное не указано явно. Редакция оставляет за собой право корректировать присланные письма и другие материалы. Редакция Linux Format получает неэксклюзивное право на публикацию и лицензирование всех присланных материалов, если не было оговорено иное. Linux Format стремится оставлять уведомление об авторских правах ввиду, где это возможно. Свяжитесь с нами, если мы не упомянули вас как автора предложенных вами материалов, и мы постараемся исправить эту ошибку. Редакция Linux Format не несет ответственности за опечатки.

Ответственность за содержание статьи несет ее автор. Мнение авторов может не совпадать с мнением редакции.

Все присланные материалы могут быть помещены на диски — CD или DVD, поставляемые вместе с журналом, если не было оговорено иное.

Ограничение ответственности: используйте все советы на свой страх и риск. Ни при каких условиях редакция Linux Format не несет ответственность за повреждения или ущерб, нанесенные вашему компьютеру и периферии вследствие использования тех или иных советов.

Linux — зарегистрированный товарный знак Линуса Торвалдса (Linus Torvalds). "GNU/Linux" заменяется на "Linux" в целях сокращения. Все остальные товарные знаки являются собственностью их законных владельцев. Весь код, опубликованный в журнале, лицензирован на условиях GPL v3. См. www.gnu.org/copyleft/gpl.html За информацией о журналах, издаваемых Future plc group company, обращайтесь на сайт www.futureplc.com.

РОББО КЛУБ™

Образовательные кружки для детей

Робототехника ● Программирование ● 3D-моделирование ● 3D-печать



Обучение: от игры к профессии

для дошкольников

- ★ **Игровая робототехника**
Возраст — от 5 лет.
Требуется уверенный навык чтения

для школьников

- ★★ **Стартовый**
Основы знаний
- ★★★ **Базовый**
Специальность
- ★★★★ **Углубленный**
Профессия, соревнования, собственные проекты

А ТАКЖЕ

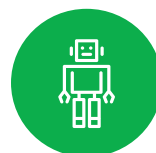
- Праздничные мастер-классы
- Городской лагерь на время каникул
- Курсы повышения квалификации **для педагогов**



Инновационная
программа обучения

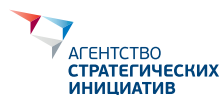


Собственная
методическая база



Российские технологии
и оборудование

ВООБРАЖАЙ! ПРОГРАММИРУЙ! СОЗДАВАЙ!



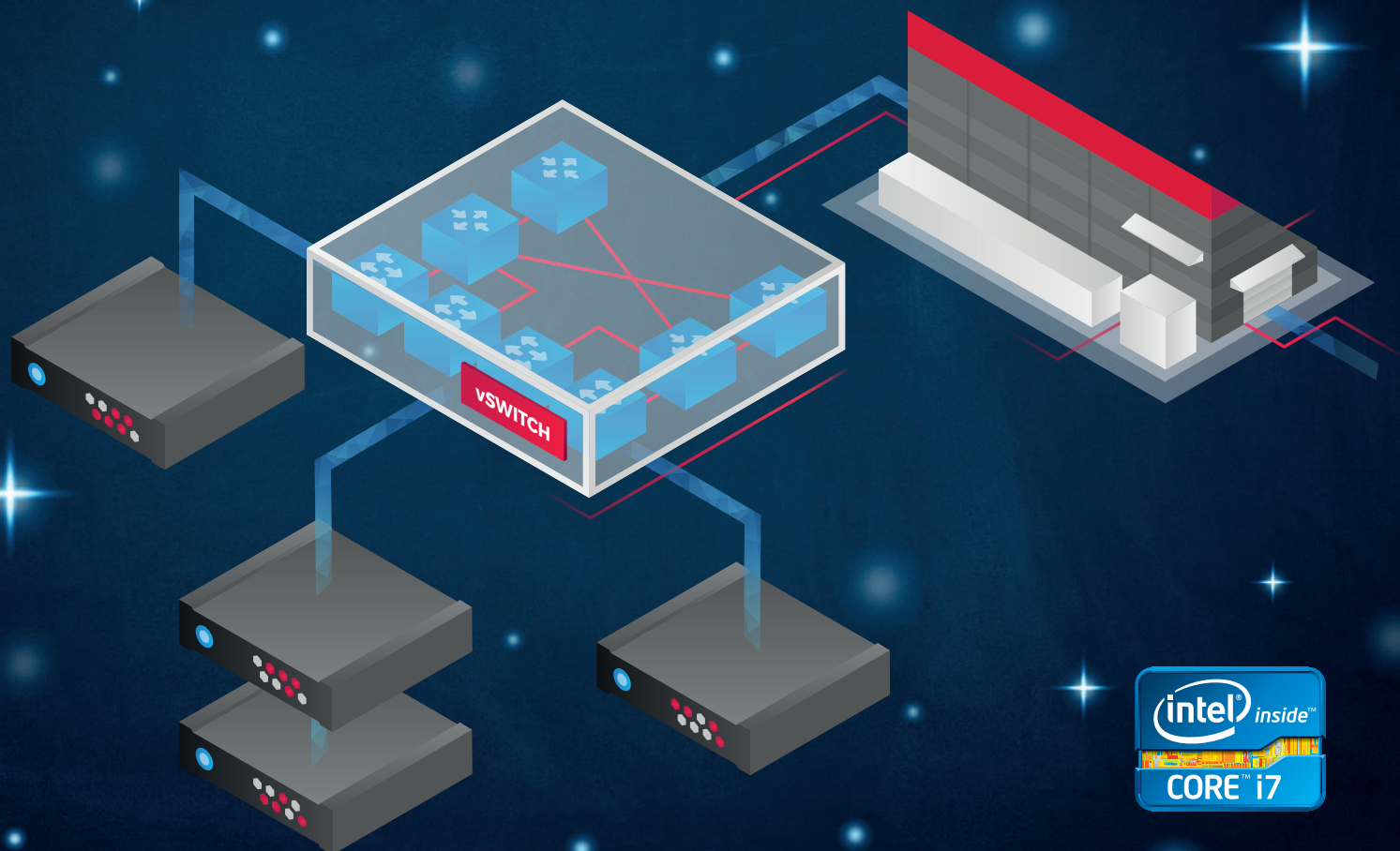
www.robboclub.ru

8 800 777 2985

vk.com/robboclub ● info@robboclub.ru

vSWITCH

ВИРТУАЛЬНЫЙ КОММУТАТОР
БЕЗ ДОПОЛНИТЕЛЬНОЙ ОПЛАТЫ



Выделенный сервер SX61

- ✓ Intel® Core™ i7-3770 Quad-Core
- ✓ 32 ГБ DDR3 RAM
- ✓ 4 x 6 TB SATA 3 Gb/s 7200 rpm
- ✓ 30 ТБ трафик*
- ✓ Без минимального контракта
- ✓ Установка 5400 рублей

5400 рублей в месяц

* Нет платы за превышение. При превышении 30 ТБ/месяц скорость соединения ограничивается (подсчёт ведётся по исходящему трафику, входящий и внутренний трафик не учитывается). Опционально можно снять ограничение, подтвердив оплату 80 руб. за каждый дополнительный ТБ.

vSwitch – Построй свою виртуальную сеть!

vSwitch использует технологию VLAN для объединения нескольких выделенных серверов находящихся в разных дата-центр парках Hetzner Online. Настройки vSwitch производятся через удобный веб-интерфейс панели Robot. Например, можно настроить частную сеть для безопасного обмена данными или заказать публичную подсеть и сделать систему отказоустойчивой.

ru.hetzner.com

Все цены вкл. НДС 18%. Цены могут измениться без уведомления. Все права защищены соответствующими производителями. Intel, логотип Intel, Core и Core Inside являются товарными знаками корпорации Intel в США и других странах.