

DVD ДАРОМ!



РЕМИКС UBUNTU 17.04

# LINUX FORMAT

Главное в мире Linux

Июль 2017 № 7 (225)

**73** страницы  
учебников  
и статей!

- » Ставим Linux на MacBook
- » Охраняем сообщения KeyBase
- » Строим сайты с Varnish



Академия кодига: Перейдем на Swift3 прямо сейчас!

# ХАКЕРАМ — НЕТ!

Сомкните щиты своей системы и станьте экспертом в безопасности

- » Доскональные руководства по Kali
- » Взломы Wi-Fi » Анализ пакетов
- » Инъекция нагрузки » Сканы портов
- » Потренируйтесь на Metasploitable

ЕСТЬ  
ЭЛЕКТРОННАЯ  
ВЕРСИЯ  
ДЛЯ ВАШЕГО  
ПЛАНШЕТА!



## Наш Exercist

“ Научиться беглости, имея только базовые знания — в этом прелесть Exercism ”

Катрина Оуэн — об опасностях при обучении программированию

## Плюс: Для Pi

8 страниц в копилку Raspberry Pi

- » Запускаем сервер Minecraft
- » Рулим мотором и аналоговым вводом



Cordova

## Android внутри Linux

» Простой способ писать мобильные приложения с открытым кодом



Сравнение

## Программы анимации

» Открытые средства для создания сиквела «Истории игрушек»



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

Linux center  
www.linuxcenter.ru



# iTeleRadio

ИНТЕРАКТИВНОЕ РАДИО&ТЕЛЕВИДЕНИЕ

iteleradio.ru

ОТРН 1097847090021



## Что мы делаем

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

## Кто мы

На сей раз мы спросили наших экспертов: в этом номере мы запугиваем наших читателей до бесчувствия — а чего вы сами больше всего боитесь в Linux?



### Джонни Бидвелл

Как ни странно, на днях я понял, что мой компьютер не проснется от сна. Выключаю. Включаю. Корневая файловая система повреждена. Попытка ремонта. Оказывается, когда разработчики btrfs говорят, что не используют btrfsck -repair, это сущая правда. Моя система Arch схлопнулась. И я потратил приятнейшие пять часов на переустановку...



### Найт Дрейк

Я боюсь отмены Linux. Французский парламент в 2007 г. перешел на Ubuntu, но в 2012 г. депутатам был предоставлен выбор, чтобы вернуться в Windows. Город Мюнхен тоже, похоже, серьезно рассматривает возможность отката от Linux к Windows 10.



### Ник Пирс

Решение Ubuntu о прекращении разработки Unity подчеркивает один из страхов: Linux становится менее разнообразным и более однородным. Пусть вам не нравится Unity, но, по крайней мере, это была альтернатива Gnome. Будем надеяться, что другие нишевые рабочие столы, такие как Pantheon в Elementary, не пойдут этим путем.



### Лес Паундер

Мое главное опасение за Linux? То, что навыки и знания, которые у нас есть, не передаются следующему поколению системных администраторов, хакеров и разработчиков. Конечно, в мире найдутся люди, которые этим занимаются, но хватит ли их на следующее поколение линуксоидов?



### Шашанк Шарма

Страхов у меня множество, и некоторые из них, вроде случайного dd в неправильном разделе, даже сбываются. Сейчас меня беспокоят экзистенциальные вопросы, такие как «вдруг дистрибутивы не будут поставляться с тап-страницами»? Или вдруг я забуду, как работает функция раскрытия истории Bash. Ой!



### Валентин Синецын

Самое страшное — это забыть пароль root. В старые добрые времена всегда можно было вбить "linux single" в приглашение LILO и получить оболочку root на блюде. А сейчас, со всеми этими Grub, systemd и sudo, никогда не знаешь наверняка, init=/bin/bash вроде бы должен работать везде, где есть Bash, но я уже ни в чем не уверен на 100%.



## Холодное лето 17-го

» Весна и начало лета нынешнего года были богаты новостями из области информационной безопасности. Несколько крупных вирусных эпидемий и утечек данных, поиски «русских/северокорейских/китайских (нужное подчеркнуть) хакеров» и даже компрометация российского сайта госуслуг... Не уверен, что инцидентов стало больше, но тон их освещения в неспециализированной прессе у слабонервных может вызвать панику.

С другой стороны, явно усиливается желание бюрократии всех стран «по-регулировать» киберпространство. Способ везде одинаков: запретить! Что — VPN и анонимайзеры, как в Саудовской Аравии и России, GitHub, как в Турции — не важно. Ущерб для бизнеса — не наше дело... Главное — обозначить заботу о безопасности граждан и продемонстрировать собственную полезность. Похоже, что угроза распада Интернета по национальным границам, предсказанная аналитиками «Лаборатории Касперского» еще четыре года назад, становится реальной.

С точки зрения теории информационной безопасности, подобные запреты — очередная угроза доступности данных, как и троян-«шифровальщик». Методы противодействия тоже общеизвестны — резервирование служб и бэкап данных. Вот и оффлайновую Википедию я на всякий случай качнул... Кто знает, какими будут завтрашние новости...

### Кирилл Степанов

Главный редактор

» [info@linuxformat.ru](mailto:info@linuxformat.ru)

## Как с нами связаться

Письма для публикации: [letters@linuxformat.ru](mailto:letters@linuxformat.ru)

Подписка и предыдущие номера: [subscribe@linuxformat.ru](mailto:subscribe@linuxformat.ru)

Техническая поддержка: [answers@linuxformat.ru](mailto:answers@linuxformat.ru)

Общие вопросы: [info@linuxformat.ru](mailto:info@linuxformat.ru)

Проблемы с дисками: [disks@linuxformat.ru](mailto:disks@linuxformat.ru)

Вопросы распространения: [sales@linuxformat.ru](mailto:sales@linuxformat.ru)

Сайт: [www.linuxformat.ru](http://www.linuxformat.ru), группа «ВКонтакте»: [vk.com/linuxform](https://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](http://linuxformat.ru/avtoram.phtml)

Зарегистрируйтесь  
на сайте  
[shop.linuxformat.ru](http://shop.linuxformat.ru)



СОМНЕНИЯ

Скачайте бесплатно  
архивные PDF-номера  
журнала с сайта  
[linuxformat.ru/archive](http://linuxformat.ru/archive)  
и загляните на  
[wiki.linuxformat.ru](http://wiki.linuxformat.ru)



Версия для iPad  
и iPhone доступна  
в App Store

Выберите вид подписки

PDF-версия  
на 6 месяцев

990 ₺

PDF-версия  
на 12 месяцев

1800 ₺

Печатная версия  
на 6 месяцев

1890 ₺

Печатная версия  
на 12 месяцев

3480 ₺

Выберите вид доставки

Оплатите

# Читайте Linux Format!

Станьте Linux-гуру

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

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

## Печатная версия Linux Format понравится читателям, которые:

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

## Способы доставки

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

## Способы оплаты

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



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

# Содержание

«Богатым и бедным равно возбраняется спать под мостом и красть хлеб.» Анатолий Франс

## Обзоры

### Kali 2017 ..... 14

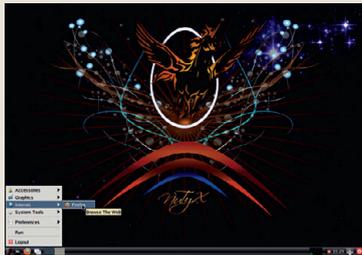
Популярный дистрибутив тестирования на вторжения с возобновляемым релизом предлагает вам стать на позиции хакера.

### Solus 2017.04.18 ..... 15

Почему бы этому дистрибутиву с рабочим столом Budgie — симпатичному и простому в работе — не сделаться вашим новым приключением в Linux?

### NuTux 9.0 ..... 16

Гибкий, опрятный и аккуратный независимый дистрибутив а-ля Linux from Scratch — вы сами заполняете его программами.



➤ Проявив смекалку, вы найдете тут некую связь с названием дистрибутива.

### Devil Linux 1.8.0 ..... 17

Дьявольски заинтриговывает, завлекая в мир дистрибутивов легковесных роутеров/брандмауэров. А потом разочаровывает.

### AMD Ryzen 5 1600X ..... 18

Когда 4 ядра — маловато, а 8 — явный перебор, Ryzen 5 с его 6-ю ядрами будет в самый раз. И притом 12 потоков... конкурентам ловить нечего.



➤ Гексадерный — это по-латыни будет или по-гречески?

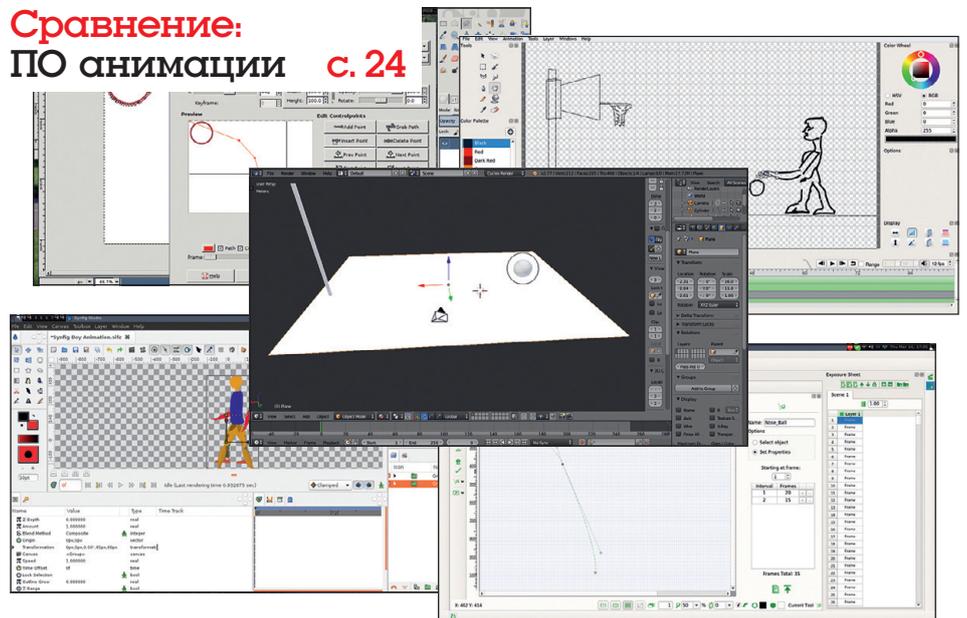
### Full Throttle ..... 19

Классическая приключенческая антиутопия от LucasArts освежила краски и звук ради соответствия антиутопии XXI века.

## Учимся хакингу 2017

Переодеваемся этичными хакерами, запускаем Kali и простукиваем все слабые места в своей сети. Сомнем щиты на с. 30

## Сравнение: ПО анимации с. 24



## Интервью



«Сообщество Ruby очень своеобразное. Они очень любят каламбурить»

Катрина Оуэн — про Automatic Grade Linux с. 38

# На вашем бесплатном DVD

**ЭКСКЛЮЗИВ**  
**ОСОБЫЙ UBUNTU**

Пробуйте будущий Gnome и еще 5 жизненно важных рабочих столов Linux!

UNITY XFCE GNOME CINNAMON KDE MATE

**Solus 2017** **Android 6.0**

**LINUX LIVE-ДИСК: ГОТОВ К РАБОТЕ**  
ВСЕ НЕОБХОДИМОЕ ДЛЯ СТАРТА В LINUX

**Ubuntu Remix 17.04** 64-битный  
**Android x86 6.0** 32-битный  
**Solus 2017.04.18** 64-битный

» Каждый месяц — только лучшие дистрибутивы

**ПЛЮС:** HotPicks, код и библиотека **с. 106**

Побалуйте себя и любимых подпиской на LXF!

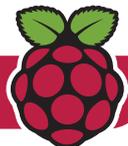


Доступно в AppStore!



[www.linuxformat.ru/subscribe](http://www.linuxformat.ru/subscribe)

## Пользователям Raspberry Pi



- Новости Pi** ..... 88  
Организация Turkey Blocks награждена за отслеживание правительственных блокировок доступа к Internet в Турции, готовится игровая консоль PiPlay, а Pi Zero W наращивает продажи.
- Pimoroni Scroll Bot** ..... 89  
Исследуем забавный комплект для Pi Zero, с помощью которого можно смастерить робота, выдающего сообщения.
- Действуем по аналогии** ..... 90  
У Raspberry Pi нет аналоговых входов? Не беда, мы надставим его GPIO новеньким АЦП.
- Про диоды и моторы** ..... 92  
Рассматриваем четыре проекта, задействующие аналоговые сигналы на полную катушку.

**Змей на дискотеке** ..... 94  
Предлагаем ослепительную прозу про супер-яркие цветные гирлянды из NeoPixel, и кое-какой Python впридачу.

**Без ума от Minecraft** ..... 96  
Редакция любимой игры миллионов для Raspberry Pi улучшается с каждым днем.



## Ищите в номере

- Вся жизнь** ..... 22  
Управляем жизненным циклом нативных приложений OC Tizen.
- Привет, Мак!** ..... 48  
Описываем препятствия и суровые пере(за)грузки, через которые вам надо пройти, чтобы поселить Linux на своем MacBook.



## Учебники

- Скрипты**  
**Проверки на дорогах** ..... 54  
Автоматизируем свои действия в Linux, не забывая справляться с результатами этих действий.
- Основы терминала**  
**Работа в Internet** ..... 56  
Разгуливаем в Internet, забираем почту, загружаем файлы, добываем прогноз погоды — и все это из командной строки.
- Безопасность**  
**IoT под замком** ..... 58  
Незачем допускать посторонних к своей деятельности, так что позаботимся о сохранности потоков с камер видеонаблюдения.
- Web-сайты**  
**Параллельный Internet** ..... 62  
Объясняем, как создать защищенный сайт, который не прочтет никто без вашего ведома.
- Шифрование**  
**Yubikeу** ..... 66  
Расширим свой взгляд на мир, раскопав модель доверия на основе ключей GnuPG.
- Обмен сообщениями**  
**Говорим в Keybase** ..... 70  
Деловым людям требуется простой, но надежный способ для переговоров.
- Управление пакетами**  
**APT** ..... 74  
Учимся устанавливать и обновлять программное обеспечение как истые линуксоиды.
- Интернет вещей**  
**Домашнее видео** ..... 76  
Заделяем в Послушном Доме видеонаблюдение и узнаем, как ведет себя пёс в отсутствие хозяев.

## Академия кодирга

- Swift 3** ..... 80  
Открытый язык программирования, за спиной которого стоит Apple, радикально повзрослел, так что пора устанавливать, обновляться и приниматься за кодирование удобных инструментов.
- Проги для Android** ..... 84  
Настраиваем Linux-машину под создание приложений для Android со всеми удобствами Cordova, Netbeans и Android Studio.

```
1 var aSet = Set<Int>()
2
3 print(aSet)
4 let number = 1
5 if(aSet.contains(number)) {
6     print(number, "is in the set")
7 } else {
8     print(number, "is not in the set")
9 }
10
11 var count = 0
12 aSet.insert(1)
13 aSet.insert(1)
14 for element in aSet {
15     count += 1
16 }
```

## Постоянные рубрики

- Новости** ..... 6  
Вышел Debian 9, Intel и AMD состязаются, Швейцария потеснила США, университеты Японии получают суперкомпьютеры, виртуальная реальность обретает форматы, США пытаются обогнать Китай, диск от Intel слегка разочаровал, а у Raspberry Pi появился конкурент.
- Вести мобильных ОС** ..... 20  
Представлен первый в мире сетевой процессор, стартовала IoT-Академия Samsung в Москве, Qualcomm занялся распознаванием речи, а SoftBank и ZTE замаяхулись на 5G.
- Сравнение** ..... 24  
Оценим пейзаж открытых инструментов анимации и выберем лучший: *Blender, GIMP, Pencil2D, Synfig, Tupi*.
- Интервью LXF** ..... 38, 42  
По мнению Катрины Оуэн, беглость и компетентность не обязаны совпадать, а Вадиму Кулешову очень нравится тестирование.
- HotPicks** ..... 100  
Отведаете горяченького! Лучшие в мире новинки свободного ПО: *CMST, Elisa, GNU Emacs, Libva, Nylas Mail, OpenTTD, Otter Browser, Persepolis, Redshift, Simon, Warzone 2100*.
- Диск Linux Format** ..... 106  
Содержимое двустороннего DVD этого месяца.
- Пропустили номер?** ... 108  
Поторопитесь заказать, ведь к нему прилагаются аж два Ubuntu 17.04 в первоизданном виде (32- и 64-битный), прямо от Canonical, без наносного.
- Через месяц** ..... 112  
Раскручиваем кампанию перехода с Windows на Linux! Расскажите своим друзьям и родным, как подружиться с любезным нашему сердцу пингвином.



**В ЭТОМ НОМЕРЕ:** Помянем » Intel против AMD » Сюрприз от Швейцарии » ИИ — студентам » Стандарт VP » Гонка за лидером » Новый диск » Играем свободно

## ДИСТРИБУТИВ МЕСЯЦА

# Памяти Яна Мердока

Выпущен Debian 9 “Stretch”.

**П**осле 26 месяцев разработки Проект Debian представил новую стабильную версию своего дистрибутива, которая будет поддерживаться в течение следующих 5 лет и посвящена памяти основателя Проекта — Яна Мердока, ушедшего из жизни 28 декабря 2015 г.

Вариантом MySQL по умолчанию в Debian 9 стала MariaDB; Firefox и Thunderbird заменили безбрендовые варианты Iceweasel и Icedove; для запуска системы отображения X более не требуются права суперпользователя. Благодаря проекту по обеспечению воспроизводимых сборок, сборка более 90% пакетов с исходным кодом теперь приводит к созданию побитно идентичных двоичных пакетов; это позволяет

выполнять проверки и защищать пользователей от злонамеренных попыток манипуляции компиляторами и сборочными сетями.

Stretch — первая версия Debian, в которой в пакете *gnupg* используется современная ветка *GnuPG*; она поддерживает криптографию на основе эллиптических кривых, лучшие настройки по умолчанию, лучшую модульную архитектуру и улучшенную поддержку смарт-карт.

Значительно улучшена впервые появившаяся в Wheezy поддержка UEFI, добавлена возможность установки на 32-битную прошивку UEFI 64-битного ядра. Живые образы Debian теперь также поддерживают загрузку UEFI.



» Рубрику готовил  
**АНДРЕЙ  
ГОНДАРЕНКОВ**

Stretch содержит более 51 тыс. готовых к использованию пакетов ПО, собранных из немногим более 25 тыс. пакетов с исходным кодом, и поддерживает 10 архитектур: amd64 и i386; ppc64el; s390x; armel, armhf и arm64; mips, mipsel и mips64el (поддержка 32-битного варианта powerpc прекращена).



» 17 июня состоялся официальный релиз очередного персонажа «Истории игрушек», Debian 9 “Stretch”.

## CPU

# Гонки чипмейкеров

Новые Intel Core-X уже доступны, а ‘Threadripper’ Ryzen 9 от AMD — пока нет.

**I**ntel в очередной раз обошла AMD: в конце июня начались поставки ее новых микропроцессоров X-серии и материнских плат X299 — раньше, чем AMD выпустила свои CPU ‘Threadripper’ Ryzen 9. Новая линейка Core-X начинается с младшей модели Intel Core i5-7640X по цене £216,98, а старшая модель, Intel Core i9-7900X, доступна за £885,98. Базовая тактовая частота четырехъядерного Intel Core i5-7640X — 4 ГГц (4,2 ГГц в турборежиме), в то время как Core i9-7900X предоставляет 10 ядер и 20 потоков, его тактовая частота 3,3 ГГц (4,3 ГГц в турборежиме); все микропроцессоры новой X-серии поставляются с разблокированным

множителем и поддержкой разгона. Одновременно стали доступны материнские платы на новом чипсете X299; Asus Prime-X299 Deluxe предлагается за £409,99, а цена «самой дешевой» из доступных в Великобритании плат с чипсетом X299, Asus TUF X299 Mark 2 — £229,99. Новые микропроцессоры и материнские платы для них предназначены, в первую очередь, энтузиастам — домашним пользователям ПК, либо имеющим экстремальные требования к обработке графического или видеоконтента, либо запускающим новейшие игры с максимально возможными настройками; этими компонентами также будут комплектоваться мощные рабочие станции.

» На момент представления Intel первых процессоров X-Series брендом Core i9 официально был отмечен только 10-ядерный CPU 7900X.



Что же касается 18-ядерных CPU Extreme Core Count (XCC), то они появятся не ранее третьего квартала с предполагаемой ценой около \$2000 (£1800) — к тому времени, конечно же, конкурирующая серия AMD Ryzen 9 уже тоже будет доступна.

TOP-500

# Самые производительные

Швейцарский кластер вытеснил США из тройки лидеров.

Опубликованный в июне 49-й выпуск рейтинга самых производительных суперкомпьютеров планеты Top-500 преподнес несколько сюрпризов, главный из которых — третье место построенной на платформе Cray XC50 швейцарской машины Piz Daint. Модернизация расположенного в Национальном компьютерном центре Швейцарии кластера дополнительными GPU Nvidia Tesla P100 позволила вдвое, по сравнению с предыдущим рейтингом, увеличить его производительность (19,6 петафлоп против 9,8 петафлоп в тестах Linpack). Лидером рейтинга остается Sunway TaihuLight из Национального суперкомпьютерного центра Китая в Уси; его производительность, 93 петафлоп, почти в 3 раза выше, чем у ближайшего соперника, Tianhe 2 (Milky Way-2) из Национального суперкомпьютерного центра в Гуанчжоу (33,9 петафлоп). На четвертое место сместился расположенный в Окриджской национальной лаборатории Ми-

нистерства энергетики США кластер Titan, построенный на платформе Cray XK7 (17,6 петафлоп). Второй раз за 24 года существования Top-500 в тройку его лидеров не вошли суперкомпьютеры США; тем не менее, именно США принадлежат 5 из 10 лучших суперкомпьютеров мира.

Остальные места в Top-10:

- 5 Sequoia: 17,2 петафлоп (IBM Bluegene);
- 6 Cori: 14 петафлоп (Cray XC40);
- 7 Oakforest-PACS: 13,6 петафлоп (Fujitsu);
- 8 Fujitsu: 10,5 петафлоп (Fujitsu);
- 9 Mira: 8,6 петафлоп (IBM Bluegene);
- 10 Trinity: 8,1 петафлоп (Cray XC40).

Говоря об отечественных системах, Lomonosov 2 опустился на 59 место (с 52), Lomonosov — на 164 (с 132), Tornado — на 297 (с 226). В рейтинге — 3 наших кластера; полгода назад их было 5, а 5 лет назад — 12.

498 машин из представленных в рейтинге 500 работают под управлением Linux, и лишь 2 китайских суперкомпьютера IBM POWER используют AIX (493 и 494 места).



Включение в состав Piz Daint графических процессоров Nvidia позволило швейцарской системе занять в новом Top-500 третье место, потеснив Titan Cray XK7 из Oak Ridge National Laboratory.

Процессоры Intel установлены в 464 машинах, Power PC — в 21 машине, AMD — в 6 машинах. Всё большее распространение получают сопроцессоры: GPU Nvidia присутствуют в 74 системах.

Лидерство по количеству суперкомпьютерных систем вернулось снова к США — 168, далее идут Китай — 160 и Япония — 33.

ДЛЯ ИИ

# В стране Восходящего солнца

Университет Кюсю получит от Fujitsu новую суперкомпьютерную систему.

Научно-исследовательский институт информационных технологий Университета Кюсю в Фукуоке (Япония) объявил, что в октябре получит новую суперкомпьютерную систему для всех университетов страны для расширения исследований в областях, связанных с искусственным интеллектом (ИИ). Новую систему от Fujitsu задействуют в качестве вычислительного ресурса в сети JNPCN, объединяющей суперкомпьютеры нескольких японских университетов; ядром сети является Центр информационных технологий Токийского университета. Новый суперкомпьютер будет также использоваться высокопроизводительной вычислительной инфраструктурой (HPCI) — средой, соединяющей компьютер К и ряд крупных суперкомпьютеров по всей Японии.

Сервер суперкомпьютера от Fujitsu будет состоять из трех подсистем: внутренней, внешней и хранения. Внутренние вычислительные узлы образуются 2128 системами

Primergy CX400 на процессорах Intel Skylake Xeon, с общей емкостью памяти 433 ТБ. Каждый из 128 серверов x86 оснащен четырьмя GPU Nvidia Tesla P100. Внешняя подсистема включает 160 базовых узлов с процессорами Intel Skylake Xeon и Nvidia Quadro P4000, а также четыре высокопроизводительных узла с объемом памяти 12 ТБ ка-

**Это будет первая в Японии суперкомпьютерная система.**

ждый. При теоретической пиковой производительности около 10 петафлоп, новый суперкомпьютер имеет в своем составе систему хранения на 24 ПБ (петабайта), соединение 100 Gbps InfiniBand EDR и масштабируемую файловую систему (ФС) Fujitsu для кластеров FEFS. Это будет первая в Японии суперкомпьютерная система с крупно-

масштабной приватной облачной средой на внешней подсистеме, связанной с вычислительным сервером внутренней подсистемы через высокоскоростную ФС.

В апреле в Riken Center for Advanced Intelligence Project запустили суперкомпьютер для глубинного обучения, позволяющий ускорить исследования и разработки реального применения технологий ИИ. Система состоит из двух серверных архитектур с 24 серверами Nvidia DGX-1; каждый из них включает 8 новейших ускорителей Nvidia Tesla P100, интегрированное ПО для глубинного обучения, 32 сервера Fujitsu Server Primergy RX2530 M2, а также высокопроизводительную систему хранения.

ФС здесь также FEFS на шести серверах Fujitsu Server Primergy RX2540 M2 PC; 8 систем хранения Fujitsu Storage Eternus DX200 S3; одна система хранения Fujitsu Storage Eternus DX100 S3 обеспечивает обработку IO, для анализа глубинного обучения.

СОЗДАНИЕ КОНТЕНТА

# Google задает стандарт

Виртуальная реальность обретает форматы.

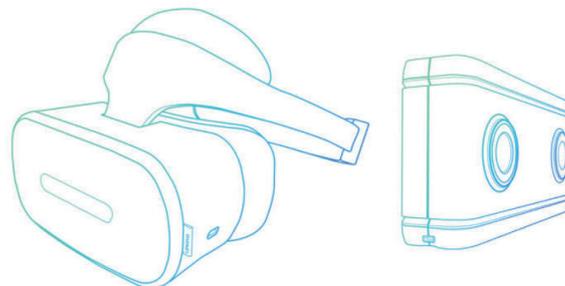
**G**oogle создает новый формат для VR-контента, а Yi, Lenovo и LG уже готовят камеры под этот формат. Google Daydream VR team совместно с YouTube создают новый видеоформат, призванный стандартизировать создание контента для виртуальной реальности. Видео в формате VR180 в высоком разрешении показывает, в первую очередь, то, что пользователь видит перед собой. VR-камеры, снимающие в новом формате, уже готовит компания Yi, Lenovo и LG; Google обещает, что в обращении камеры формата VR180 будут не сложнее обычных компактных камер при приблизительно схожей цене. Такие камеры, оснащенные двумя объективами, появятся в продаже зимой.

Угол обзора у видео нового формата ограничен 180 градусами; на решение разработчиков отказаться от полных 360 повлияли, в том числе, результаты проведенного сотрудниками YouTube исследования, согласно которым 75% времени просмотра видеоролика пользователи уделяют фронтальным 90 градусам. «Реальность

такова, что съемка 360-градусного VR-видео не очень проста для большинства создателей контента, а некоторые VR-камеры получаются дорогими», пояснила CEO YouTube Сьюзан Воджицки [Susan Wojcicki].

Видео в формате VR180 можно просматривать как на десктопах, так и на мобильных устройствах, а эффект 3D достигается с помощью Cardboard, Daydream или PSVR; формат поддерживает и потоковое вещание. В скором времени создателям VR-контента в формате VR180 будет доступно редактирование в Adobe Premier Pro и других подобных приложениях. YouTube запустил сертификационную программу VR180, первым партнером которой стал китайский производитель VR-камер Z CAM.

На прошедшей в мае конференции Google I/O стало известно, что Lenovo и HTC



► На своей партнерской странице с Google VR компания Lenovo демонстрирует прототипы гарнитуры для платформы Daydream и камеру формата VR180.

Vive разрабатывают новые VR-гарнитуры, основанные на прототипах от Google и Qualcomm: высококачественные, и при этом более дешевые, чем существующие VR-комплектеты от Oculus и HTC. Воджицки также сообщила, что в настоящее время YouTube посещают 1,5 млрд зарегистрированных пользователей в месяц, при этом пользователи тратят более часа в сутки на просмотр контента YouTube только на мобильных устройствах.

**VR-камеры, снимающие в формате VR180, готовят Yi, Lenovo и LG.**

ДОГНАТЬ И ПЕРЕГНАТЬ!

# США в роли догоняющего

США выделили \$258 млн на создание эксасфлопного суперкомпьютера.

**В** течение трех лет США потратят \$258 млн на создание суперкомпьютера с производительностью один Эксасфлоп (это более 1 тыс. Пфлоп, или более 1018 операций с плавающей запятой в секунду). Министр энергетики Рик Перри [Rick Perry] объявил, что Вашингтон будет спонсировать компаниям AMD, Cray, HPE, IBM, Intel и Nvidia разработку соответствующих аппаратных технологий, технологий программирования и приложений, при этом каждому участнику будет компенсировано до 40% затрат на разработку.

Финансирование будет поступать из подразделения министерства энергетики Exascale Computing Project, и подпадает под программу PathForward по созданию системы one-exascale к 2021 г. «Продолжение лидерства США в высокопроизводительных вычислениях — основа нашей безопасности, процветания

и экономической конкурентоспособности как нации», заявил Перри. «Работа, финансируемая PathForward, будет включать разработку инновационных архитектур памяти, высокоскоростных соединений, улучшенных систем эксплуатационной на-

**Помимо США, Китая и Японии, работы ведут также в Европе.**

дежности и подходов к увеличению вычислительной мощности без чрезмерного увеличения энергопотребления», пояснил директор подразделения ECP Пол Мессина [Paul Messina].

В последние годы два китайских суперкомпьютера стабильно занимают первые два места в общемировом рейтинге

Топ-500 для суперкомпьютеров. Оба эти вычислительных гиганта находятся в ведении Национального суперкомпьютерного центра Китая; машина Sunway Taihu-Light, расположенная в Уси, имеет производительность 93 петафлоп, а Tianhe-2 из Гуанчжоу — 34 петафлоп.

Помимо США, Китая и Японии, работы над суперкомпьютером нового поколения ведут также европейские проекты CRESTA, DEEP и Mont-Blanc, при этом Китай и Япония планируют создать свои суперкомпьютеры к 2020 г., а европейские команды — к 2022 г. Так, Национальный институт прогрессивной промышленной науки и технологии Японии намерен создать компьютер производительностью 130 петафлоп для развития искусственного интеллекта; ожидается, что на реализацию данного компьютера Япония потратит около 19,5 млрд иен.



redhat.

## 22 СЕНТЯБРЯ RED HAT, ВЕДУЩИЙ МИРОВОЙ ПОСТАВЩИК ПРОГРАММНЫХ РЕШЕНИЙ НА БАЗЕ ОТКРЫТОГО ИСХОДНОГО КОДА, ПРОВЕДЕТ КРУПНЕЙШЕЕ ЕЖЕГОДНОЕ СОБЫТИЕ В СФЕРЕ ОТКРЫТЫХ ИННОВАЦИОННЫХ ТЕХНОЛОГИЙ В РОССИИ — RED HAT FORUM 2017!

Ведущие специалисты Red Hat и представители сообщества Open Source соберутся вместе, чтобы обсудить самые главные тренды цифровой трансформации, поделиться собственными историями успеха и принять участие в технических и бизнес-сессиях.

Тема грядущего Red Hat Forum 2017 — **«Сделай это реальностью»**. В повестке дня будут самые обсуждаемые темы последних лет:

- Платформы.
- Облака.
- Промежуточное ПО.
- Контейнеры.
- DevOps.

Инженеры и архитекторы Red Hat расскажут о том, как проактивно управлять рисками, какую роль в цифровой трансформации предприятий играют контейнеры, и как технология Ansible позволяет автоматизировать большинство задач и операций. Гости форума также узнают подробности о главных продуктовых новинках компании: новой версии системы управления облаками **Red Hat CloudForms 4.5**, а также **Red Hat Hyperconverged Infrastructure** — первой в истории отрасли гиперконвергентной инфраструктуры с открытым кодом, предназначенной для организаций с развитой сетью отделений и представительств.

Среди специальных гостей форума — **Мишель Изнар** [Michel Isnard], вице-президент по продажам Red Hat в Европе, на Ближнем Востоке и в Африке, и **Ян Вилдебоер** [Jan Wildeboer], евангелист Open Source в Европе, на Ближнем Востоке и в Африке.

**Зарегистрироваться на Red Hat Forum 2017 и узнать все самое главное из первых уст можно на официальном сайте компании.**

**До встречи 22 сентября в Отель Ренессанс Москва Монарх Центр!**



ТЕХНОЛОГИИ ХРАНЕНИЯ

# SSD от Intel 545

Его преимущества оказались меньше, чем ожидалось.

Intel начала коммерческие поставки 64-слойного твердотельного накопителя с флэш-памятью 3D-NAND на трехуровневых ячейках TLC. Компания Western Digital еще в начале года анонсировала TLC 64-битную линейку Blue, названную Bics3, на аналогичной технологии; но дебют WD Blue намечен на III квартал года, а продажи Intel SSD 545 начались в США в конце июня.

Цена 512-ГБ SSD 545 — \$179,99 (£140) (стандартный 32-слойный привод Crucial MX300 с ближайшей сопоставимой емкостью 525 ГБ в настоящее время доступен в Великобритании по той же цене). SSD от Intel уступает Crucial MX300 по скорости: 550/500 МБ/с при последовательном чтении/записи, и 75к/90к IOPS при случайном чтении/записи; встроено шифрование AES 256; энергопотребление в активном режиме — 90 мВт (увеличено время



► Intel SSD 545, первый 64-слойный потребительский твердотельный накопитель.

автономной работы портативных компьютеров). Главный менеджер NVM Solutions Group Intel Роб Крук [Rob Crooke] сулит расширение портфеля продуктов, поддерживающих новых бизнес-клиентов и встроенные решения, а взаимодействие поколений на заводах компании позволяет ожидать быстрого роста поставок SSD 545 к середине 2018 г. Похоже, выпуск SSD 545 был затеян для обхода конкурентов, а доводку выполняют в будущих продуктах.

МИКРОКОМПЬЮТЕРЫ

# В пику Pi

PINE64 начинает поставки работающего под управлением Android 7.1 или Debian 4К-совместимого SBC ROCK64.

После успеха PINE A64 — первого в мире игрового компьютера Open Source стоимостью всего \$15, работающего на Android или других ОС на Linux, компания PINE64 создала ROCK64 — еще более мощный одноплатный компьютер (SBC), размерами сопоставимый с обычной кредитной картой. ROCK64 оснащен процессором Rockchip RK3328 Quad-Core ARM Cortex A53 64-bit, до 4 ГБ ОЗУ 1600-МГц LPDDR3, Gigabit Ethernet, 128 МБ SPI Flash, поддержкой USB 3.0, разъемом для модуля eMMC, шинами Pi-2 и Pi-P5+, гнездами 3,5 мм power jack (5V/3A) и A/V jack, двумя выделенными хостами USB 2.0, кнопками питания, сброса и восстановления, ИК-портом и слотом для карт MicroSD. ROCK64 поддерживает разнообразнейшую периферию, что позволяет производителям интегрировать его с многочисленными типами датчиков и устройств. Новый SBC способен работать под управлением различных open-source ОС, включая (но не ограничиваясь) Android 7.1 Nougat, Debian GNU/Linux и Yocto. PINE64



► Главная особенность миниатюрного SBC ROCK64 — поддержка видео 4К 60p HDR.

обещает расширить список ОС, совместимых с ROCK64, и предоставляет пользователям подробную документацию по созданию на основе своего SBC мощного мультимедиа-центра или домашнего кинотеатра. ROCK64 необходим источник питания +5 В/3 А с разъемом Type H 3,5 мм OD/1,35 мм ID "coaxial", который приобретается отдельно в магазине PINE64. Поставки ROCK64 начнутся 31 июля; модель с 1 ГБ ОЗУ предлагается за \$24,95, с 2 ГБ ОЗУ — \$34,95, а с 4 ГБ ОЗУ — \$44,95. LXF

Новости короткой строкой

» В Calculate Linux 17.6 реализованы установка системы в контейнере LXC/LXD и создание тем; повышены безопасность и стабильность двоичных пакетов (автоматизацией зависимостей). Источник: [www.calculate-linux.ru](http://www.calculate-linux.ru)

» На конференции BSides London 2017 исследователь кибербезопасности Росс Бевингтон [Ross Bevington] продемонстрировал процесс взлома ПК при зарядке электронной сигареты (с дополнительной микросхемой) через порт USB. Источник: [techxplore.com](http://techxplore.com)

» На сайте сообщества Steam появилась Linux-версия стратегии реального времени *Cossacks 3* (Казак 3). Источник: [steamcommunity.com](http://steamcommunity.com)

» Конкурс команды Libreoffice Design по созданию талисмана для *Libre-Office* ждет заявки до 31 августа 2017 г. Источник: [design.blog.documentfoundation.org](http://design.blog.documentfoundation.org)

» На аппаратные средства компании System76 вместо Ubuntu предустанавливают новый Linux-дистрибутив Pop!\_OS. Источник: [www.linuxandubuntu.com](http://www.linuxandubuntu.com)

» Минобороны РФ закупает для военных вузов и своих сотрудников по всей стране тысячи защищенных ПК с ОС Astra Linux, на 1,3 млрд руб. Источник: [www.cnews.ru](http://www.cnews.ru)

» Ядро Linux 4.12 включает планировщики ввода/вывода BFQ и Kyber, фреймворк TEE (Trusted Execution Environment), утилиту *AnalyzeBoot*; механизм защиты по умолчанию — KASLR. Источник: [kml.org](http://kml.org)

» Российская компания «Базальт СПО» предлагает очередной релиз своего дистрибутива Simply Linux 8. Источник: [www.basealt.ru](http://www.basealt.ru)

» Компания NAVITEL начала продажи автомобильного навигатора NAVITEL C500 с подробной картой России. Источник: [navitel.ru](http://navitel.ru)

» Microsoft добавила Ubuntu в Windows Store, с опциями установки в сборке Windows Insider и запуска через WSL; на очереди — SUSE и Fedora. Источник: [blogs.msdn.microsoft.com](http://blogs.msdn.microsoft.com)

» «Лаборатория Касперского», НП «ИТЭЛМА», «Сименс» и «СТАН» разработают концепцию цифрового пространства промышленности России. Источник: [www.kaspersky.ru](http://www.kaspersky.ru)

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](http://unixedu.ru)

**UnixEducationCenter**

# ГНУ/Линуксцентр

*Ваш поставщик свободного программного  
и аппаратного обеспечения*

**Комплекты  
легализации СПО**

**Дистрибутивы  
GNU/Linux и СПО**  
на DVD и загрузочных  
флэшках

**Дистрибутивы  
GNU/Linux  
и СПО**  
с сертификатами  
ФСТЭК, ФСБ  
и Минобороны

**Межсетевые  
экраны**  
с сертификатами  
ФСТЭК, ФСБ  
и Минобороны



**Свободное  
аппаратное  
обеспечение**

Arduino, oLinuxino,  
Cubieboard, Raspberry Pi,  
Intel Edison, Digilent,  
3D-принтеры  
и робототехнические  
конструкторы

**Аппаратное  
обеспечение  
с прошивками  
на базе СПО**

**Обучающая  
литература**

**Атрибутика**

## Фирменный магазин и сервис-центр

*Санкт-Петербург, пр. Медиков, 5, корп. 7*

*+7 812 309 06 86 | [www.linuxcenter.ru](http://www.linuxcenter.ru)*



Новинки программного и аппаратного обеспечения в описании наших экспертов



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

## Память и тенденции

В прошлой колонке (LXF224) речь шла о новой версии среды Cinnamon. И о том, что благодаря мерам по ее оптимизации, можно было надеяться на снижение ее потребления ресурсов, вопреки известной эмпирической закономерности: в каждой новой версии любой рабочей среды требования к памяти могут только возрастать.

Увы, проверка гармонии алгеброй показала, что эта закономерность верна. И новая версия Cinnamon затребовала при первом своем запуске памяти больше, нежели предыдущая. Насколько больше — говорить неуместно, ибо это существенно зависит от аршина, которым та самая память измеряется. Однако величина эта была больше статистических погрешностей эксперимента.

Так что же, законы развития софта непреложны, и чудес не бывает? Бывают. И, как ни странно, там, где их можно было меньше всего ожидать.

С давних пор повелось, что среда KDE — самая прожорливая по части ресурсов вообще и памяти в частности. Да на самом деле, почти так и было долгое время (если исключить эпитет «самая»: сородичи от нее по сей части отставали не очень). Однако в версиях KDE Plasma 5.8–5.10 начали наблюдаться противоестественные явления: потребление памяти в каждой последующей версии снижалось. Иногда — немного, но иногда и статистически значимо. Но если тенденция продолжится в следующих релизах этой среды — начнется ведь сплошное безобразие и нарушение законов природы...  
[alv@posix.ru](mailto:alv@posix.ru)

## Сегодня мы рассматриваем:

- |  |  |
|--|--|
| <b>Kali 2017</b> ..... 14  | <b>Devil Linux 1.8.0</b> ..... 17  |
| Настоящий детектив должен уметь вживаться в психологию преступника! Дистрибутив Kali поможет вам почувствовать себя хакером. | Погнавшись сразу за тремя зайцами, этот дистрибутив в итоге не догнал ни одного. Конкуренты обходят его по всем статьям. |
| <b>Solus 2017.04.18</b> ..... 15   | <b>AMD Ryzen 5 1600X</b> .... 18   |
| Дистрибутив с возобновляемым релизом призван обеспечить элегантный рабочий стол с упором на простоту.                        | Процессор вполне заслуживает определения «дешево и сердито» — но стандартная модель для многих окажется выгоднее.        |
| <b>NuTyX 9.0</b> ..... 16  | <b>Full Throttle</b> ..... 19  |
| Не часто встретишь дистрибутивы, идущие своим путем, не полагаясь на именитую основу. Этот — стабилен и надежен.             | Представитель модной категории байкеров Бен — в бегах, поскольку его судят за преступление, которого он не совершал...   |



➤ Отличиться новой модели удалось не особенно.

### Solus 2017.04.18



➤ Сказано — сделано: вот вам и элегантность, и простота в работе.

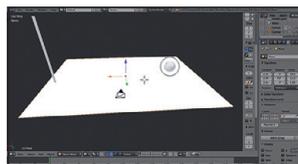
### Full Throttle



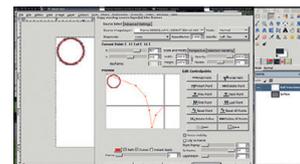
➤ Свобода, аутентичность и крутые кожаные куртки, всё как у больших.

## Сравнение: Инструменты анимации с. 24

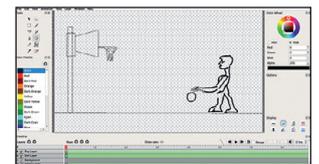
### Blender



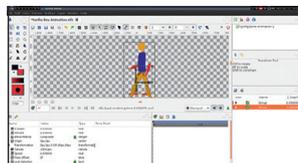
### GIMP



### Pencil2D



### Synfig



### Tupi



Не думайте, что анимация — это всего лишь забавные мультки со схематичными изображениями персонажей. Современные программы позволяют достичь уровня подлинной реалистичности.

# Kali Linux 2017.1

Всегда и во всем подозрительный Нейт «В чем подвох?» Дрейк изучает последнее предложение свыше от своей богини дистрибутивов Kali Linux.

## Вкратце

- » Поставляется с более чем 300 пентестовыми (хакерскими) инструментами.
- » Основан на Debian с рабочим столом Gnome 3.
- » Поддерживает множество платформ, включая Raspberry Pi, CuBox и Chromebook.
- » Может работать в режимах 'Live' или 'Forensic' для локального анализа машин.
- » Теперь предлагает профессиональную сертификацию Kali Linux.

**К**али Linux, который отныне следует модели возобновляемого релиза, повышен до версии 2017.1, насыщен освеженными инструментами и ядром и лучше поддерживает оборудование.

Хотя Kali отнюдь не является единственным в природе дистрибутивом для проверки на вторжение (пентеста), он остается самым популярным среди этичных хакеров. ОС содержит ряд программ вроде *Aircrack-ng*, комплекта инструментов для проверки на вторжение в беспроводные сети. С этой целью последнее предложение Kali теперь поддерживает беспроводные чипсеты RTL8812AU. Они не являются частью стандартного ядра Linux, но были включены в ответ на запрос. Это обеспечивает гораздо более инвазивное тестирование вашей сети, позволяя, например, деаутифицировать устройства, а затем повторить подтверждение связи при их повторном подключении и выявить пароль Wi-Fi.

По части драйверов Kali 2017.1 также включает гораздо более оптимизированную поддержку графических процессоров Nvidia с поддержкой CUDA. CUDA — это программная платформа, созданная Nvidia для прямого доступа к виртуальному набору команд GPU. Говоря по-русски, это позволяет использовать графические процессоры для других целей, помимо видеоигр — таких, как генерация биткойнов или, в случае Kali, взлом паролей.

На сайте Kali подробно описаны все необходимые шаги для получения инструментария CUDA, работающего с Nhashcat (сейчас обновленной до версии 3.2.0); утверждается, что вы можете начать работу



» Последняя версия Kali предлагает огромное разнообразие платформ, инструментов и документации, а также широкие возможности обучения.

за считанные минуты. Если у вас еще нету под рукой удобного арсенала графических процессоров, Kali 2017.1 теперь также поддерживает AWS и Azure Images ОС, со взломом пароля GPU «из коробки».

Хотя репозитории Kali слишком велики, чтобы поставляться вместе с ISO, теперь они включают OpenVAS 9, комплект сканеров уязвимостей, автоматизирующий многие задачи, которые пентестерам приходится выполнять вручную в целевых системах.

## Дружелюбный пентест

Несмотря на множество новых драйверов, образов и инструментов, упрощающих использование Kali, ОС продолжает оказывать поддержку новичкам пентеста. Сайт Kali уже давно предлагает онлайн-обучение OSCP (Offensive Security Certified Professional), включающее изнурительный 24-часовой экзамен. Теперь оно дополнено программой Kali Linux Certified Professional, подтверждающей вашу эффективность именно с самим Kali. Первый четырехдневный курс KLCF состоит в Black Hat USA 2017, где заинтересованные стороны могут стать ниндзя Kali Linux!

Помимо огромного количества платформ, для которых доступен Kali Linux 2017.1, это и документация, склоняющая чашу весов в пользу Kali. Другие пентест-дистрибутивы, такие как BlackArch и Parrot Security OS, могут иметь больше предустановленных или более приятных на глаз инструментов, но на Kali вы западаете сразу. Боковая панель содержит значки для

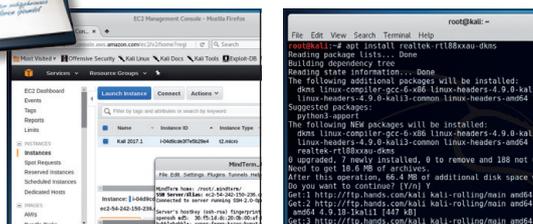
обычных хакерских инструментов, таких как Armitage и Metasploit Framework. Официальная документация Kali рассказывает о шагах по настройке ОС, обычном использовании, а также о настройке собственного ISO. Offensive Security также стоит за бесплатными курсами этичных хакеров Metasploit Unleashed, предлагающими бесплатное обучение Metasploit Framework.

В примечаниях к выпуску Kali 2017.1 также содержалось объявление о том, что Kali Linux Revealed будет опубликован в июле, после чего будет доступна бесплатная онлайн-версия.

Поскольку Kali теперь следует модели возобновляемого релиза, пользователи могут обновляться до версии 2017.1, запуская `apt dist-upgrade` в терминале. **LXF**



## Свойства новскидку



**Представления GPU в Amazon Web Services**  
Offensive Security зарегистрировала новые образы Kali с поддержкой CUDA.

**Поддержка Wireless Card Injection**  
Теперь возможно больше беспроводных атак через драйверы чипсетов.

## LINUX FORMAT Вердикт

### Kali Linux 2017.1

**Разработчик:** Offensive Security  
**Сайт:** [www.kali.org](http://www.kali.org)  
**Лицензия:** GPL

<b>Функциональность</b>	8/10
<b>Производительность</b>	8/10
<b>Удобство в работе</b>	8/10
<b>Документация</b>	10/10

» С улучшенными пакетами, драйверами и теперь даже экземплярами на web-сервере, Kali вне конкуренции.

## Рейтинг 9/10

# Solus 2017.04.18

Шашанк Шарма объясняет, почему Solus Budgie со своей самопальной средой рабочего стола и всяческими инструментами волнует его как садовника весна.

## Вкратце

» Solus предлагает различные собственные основные утилиты, помогающие облегчить новым пользователям путь в Linux. В релизе представлен последний выпуск Budgie, собственной среды рабочего стола, которая фокусируется на простоте и элегантности, чтобы обеспечить уникальный опыт. Сопоставимы — Elementary OS и Pinguy OS, с аналогичной целью привлечь новых пользователей.

**К**ак и большинство дистрибутивов Linux, Solus выпускает несколько редакций, подходящих для разных настольных сред. Помимо релизов с Gnome и Mate, флагманское ответвление и тема нашего обзора — с рабочим столом Budgie.

Гибридный ISO доступен только для 64-разрядных машин, и поддерживает машины как с MBR, так и с UEFI Secure Boot. В отличие от большинства других живых дистрибутивов, Solus не дает пользователям возможность установить дистрибутив из загрузчика. Вместо этого вы должны запустить живую среду, а затем выбрать вариант установки ОС из меню.

При установке Solus вам также не нужен раздел подкачки. Установщик настаивает на разделе в 10 ГБ или больше, но сама установка выполняется быстро. Пользователь, созданный во время установки, получает административные привилегии. Любым другим пользователям, которых вы создаете, также могут быть предоставлены права администратора.

## Необычные вещи

Меню приложений имеет сверху панель поиска и предоставляет список приложений. Нужно щелкнуть по каждой категории, чтобы просмотреть имеющиеся приложения. Еще одна особенность меню — оно реорганизует приложения в каждой категории так, что недавно использовавшиеся находятся сверху. Вы также не можете закрепить любимые приложения на панели.

Другая уникальная особенность рабочего стола Budgie — это Raven, универсальный апплет и центр уведомлений и настройки. Интерфейс с двумя вкладками перечисляет



» Если вы предпочитаете размещать значки и папки на рабочем столе, вы должны вручную включить эту функцию в Budgie с помощью Raven.

уведомления в одной, а на вкладке Applets расположены календарь, регулятор громкости и звуковые устройства. Нажмите значок шестеренки, чтобы открыть настройки Budgie. Вы можете настроить различные параметры на вкладке General, тогда как параметры панели, такие как расположение часов и значков, отнесены на вкладку Panels. Введена обновленная и облагороженная реализация Alt+Tab для ускорения работы.

Для управления программным обеспечением у пользователей есть выбор между инструментом командной строки *eorpk* и надежным *Software Center*. Это однооконное приложение, пригодное для управления программным обеспечением, установки обновлений и даже просмотра сторонних приложений, не включенных из-за проприетарных лицензий.

Обновления подразделяются на три категории. Необходимые обновления обязательны и автоматически отмечаются для установки. Исправления ошибок и новые версии приложений доступны в разделе Other Updates, а Security Updates [обновления безопасности] — отдельная категория. Также можно просмотреть коллекцию программного обеспечения со вкладки Home, где приложения разделены на категории и подкатегории.

Утилита *eorpk* обеспечивает те же функции, что *apt-get* и аналогичные инструменты. Поскольку это дистрибутив с возобновляемым релизом, вы также можете вернуться к предыдущему состоянию с помощью одной команды. В версии 2.0 вы увидите, что *eorpk* уступил место *Sol*: еще одно доморощенное приложение, написанное на C.

Нас приятно удивил Help Center, центр документации на сайте проекта. Он предлагает полезные руководства по различным темам, таким как советы и подсказки по Budgie, установке, настройке, ремонту и восстановлению и многому другому.

Ubuntu также выпустила релиз с Budgie в рамках своих выпусков 17.04. Однако он заметно медленнее по сравнению с Budgie на Solus.

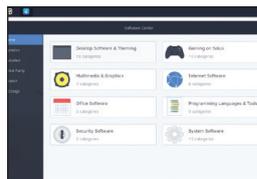
Solus Linux — это дистрибутив с возобновляемым релизом, призванный обеспечить элегантный рабочий стол с упором на простоту. Для большинства дистрибутивов это может показаться сложной задачей, но со своим доморощенным рабочим столом Budgie, Solus легко справляется с этой целью и предоставляет дистрибутив, который действительно прост в использовании, чрезвычайно надежен и интересен для работы. **LXF**

## Свойства навскидку



### Рабочий стол Budgie

Прекрасно настраиваемый, высокопроизводительный рабочий стол наверняка уведет часть пользователей от KDE.



### Собственные утилиты

В Solus множество инструментов разработано самостоятельно, включая установщик и центр программного обеспечения.

## LINUX FORMAT Вердикт

### Solus 2017.04.18 Budgie

Разработчик: Solus Project  
Сайт: [www.solus-project.com](http://www.solus-project.com)  
Лицензия: GPLv2 и др.

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

» За несколько лет дистрибутив кардинально изменился, но по-прежнему обеспечивает пользователю впечатление простоты и элегантности.

## Рейтинг 9/10

# NuTyX 9.0

Шашанк Шарма не аккуратен и не гибок, но любит дистрибутивы такого типа. NuTyX со своим инструментом управления ПО выглядит многообещающе.

## Вкратце

Основанный на Linux From Scratch, NuTyX по-прежнему как минималистская базовая система, которую пользователи наполняют по своему вкусу. Хотя он разработан для пользователей-ветеранов, его весьма интуитивный характер делает его также и хорошим выбором для новичков. Если вы ищете дистрибутив, который дает вам такой же контроль, но при более зрелом сообществе, рассмотрите Arch Linux или Gentoo.

Выпуск Knoppix, live-дистрибутива на CD, стал переломным моментом для Linux. Он породил эпоху производных дистрибутивов, но, несмотря на их изобилие (большинство из них сейчас основаны на Ubuntu), мы иногда сталкиваемся с дистрибутивами вроде NuTyX, независимо разработанного проекта со своим собственным многофункциональным инструментом управления ПО.

Доступный как образ под 300 МБ, ISO обеспечивает скелетную систему, которую пользователи сами заполняют под свои нужды. В противоположность другим дистрибутивам, NuTyX настаивает на установке на одном разделе; пользователям не рекомендуется создавать отдельные разделы, например, для / и /home. Установщик на базе Curses задает ограниченный набор вопросов, таких как раскладка клавиатуры, информация о разделах и т.д.; он тоже разработан своими силами и предоставляет пользователям на выбор fdisk или cfdisk для разбиения на разделы.

## Реально скелетная

В установленной системе не хватает многих полезных и повседневных инструментов. Например, нет графического интерфейса пользователя (GUI), не говоря уже об отдельных инструментах GUI. Не предоставляется даже Lurx, текстовый web-браузер. Нет почтового клиента, текстового процессора, музыкального приложения и т.д., но это и делает NuTyX особенным: пользователи могут свободно заполнять дистрибутив только тем ПО, которое им по душе. Для установки дополнительного ПО



В установке по умолчанию нет ни GUI, ни боев, но кое-что можно получить на сайте проекта, создав себе достойную оправу.

вы должны использовать инструмент управления программным обеспечением Cards. Команда `sudo cards install xorg lxde firefox libreoffice` получает и устанавливает всё указанное программное обеспечение и зависимости. Вы можете запросить список установленных пакетов или выполнить поиск пакетов в репозитории с помощью `cards search package-name-or-description`. К сожалению, в отличие от `apt-get`, Cards не предоставляет список зависимостей или общий объем загрузки. Вместо этого он начинает загружать пакеты сразу. В дистрибутиве также содержится ряд коллекций, включающих основные программы, такие как `cli`, `cli-extra`, `gui-extra` и многое другое. Это упрощает установку пакетов, поскольку вам не надо вводить имя каждого пакета. Но ввиду того, что каждая коллекция содержит сотни пакетов, слепая установка коллекции дает вам систему, полную пакетов, для вас лишних. Чтобы получить список пакетов, предлагаемых в каждой коллекции, выполните `cards search gui-extra`.

Дистрибутив также содержит ряд псевдонимов, определенных в файле `~/bashrc`, чтобы облегчить управление программным обеспечением. Например, вместо `sudo cards install package-name` можно запустить `get package-name`.

Если вы нарушите установку NuTyX, работая с Cards или компилируя пакеты не из репозитория, вы можете вернуться к базовой установке: `cards base -r`. На сайте

представлены руководства по широкому кругу тем, и форумы являются хорошим ресурсом, но для поиска и публикации требуют регистрации. Если вы предпочитаете видеоуроки, разработчик публикует их на специальном канале YouTube (<http://bit.ly/nutyx>).

NuTyX — не просто невероятно стабильная и надежная система, которая легко может работать и как сервер, и как станция медиа-потоков, а дистрибутив, который может стать отличным инструментом обучения для новых пользователей Linux, стремящихся расширить свои навыки за пределы таких дистрибутивов, которые просто работают, наподобие Ubuntu. **ixit**

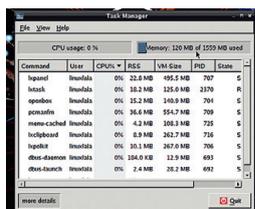


## Свойства навскидку



### Управление ПО Cards

Почти настолько же впечатляющий, как и `apt-get`, и обеспечивает почти такую же функциональность.



### Поджарый и быстрый

Скорость и малый объем — следствие дизайна без излишеств, позволяющего вам решать, какие пакеты установить.

## LINUX FORMAT Вердикт

### NuTyX 9.0

Разработчик: Тьерри Нюттенс  
[Thierry Nuttens]  
Сайт: [www.nutyx.org](http://www.nutyx.org)  
Лицензия: Разные

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

» NuTyX делает для LFS то, что живые устанавливаемые спины Linux — для Arch и Gentoo, обучая новым навыкам.

## Рейтинг 9/10

# Devil Linux 1.8.0

Заинтригованный названием и подходом, **Шашанк Шарма** нырнул в странный мир этого облегченного дистрибутива для роутеров/межсетевых экранов...

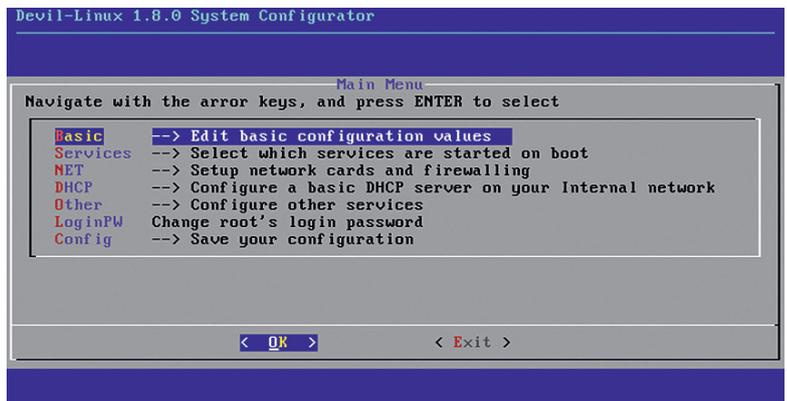
## Вкратце

» Devil Linux — независимо разработанный, легковесный дистрибутив для развертывания межсетевого экрана, маршрутизатора и сервера. Дистрибутив не устанавливается на диск, что, по мнению разработчиков, является дополнительной мерой безопасности, поскольку исключает установку вредоносного кода или ПО. Графического рабочего стола нет, и для настройки различных сервисов и серверов пользователям приходится копаться в конфигурационных файлах. См. также: *Slackware* и *Untangle*.

**D**evil Linux предполагался для использования на роутерах и брандмауэрах, но в 2003 г. релиз 1.0 провозгласил себя пригодным для серверов. Несмотря на это, много лет и релизов спустя, внешне дистрибутив практически не изменился. На самом деле, главным новшеством последней версии является переход на squashfs в качестве основной файловой системы, в попытке уменьшить размер ISO.

Специфика Devil Linux в том, что его нельзя установить на диск. Пользователи могут прожечь ISO-образ на CD или же запускать Devil Linux с USB-накопителя. Любые сделанные вами изменения, такие как определение имени хоста или статических IP-адресов, хранятся на USB-накопителе или любом другом подключенном устройстве. Во время последующей перезагрузки Devil Linux будет искать сохраненные файлы конфигурации, и если он найдет их на любом из подключенных устройств, то система загрузится с учетом ваших параметров. Поскольку дистрибутив работает чисто через ОЗУ, при перезагрузке все изменения теряются, и система загружается заново.

Фактически, в сжатом tar-архиве находится ISO-образ, а также скрипт установки на USB-накопитель (install-on-usb). Последний поможет вам создать на USB-накопителе два раздела. Первый — для хранения ISO-образа, а во второй можно поместить файл конфигурации. Это обеспечивает безопасную настройку, поскольку защищенный от записи раздел исключает возможность любых вредоносных



» Администраторам с супер-навыками этот облегченный дистрибутив для сетевых экранов и роутеров может и понравиться, а на других он, похоже, и не рассчитан.

манипуляций, способных причинить вред вашей системе.

По умолчанию, в Devil Linux нет графического интерфейса. Зато есть разные инструменты, такие как утилита брандмауэра *Shorewall*, *Nagios*, *sshd* и *webmin*. Однако установить дополнительное ПО нельзя, потому что в дистрибутиве нет даже компилятора *gcc*, не говоря уже о каком-либо инструменте управления программами.

Если при запуске скрипта установки на USB-накопитель вы допустите ошибку, установка немедленно прекратится. Например, если на вопрос, хотите ли вы форматировать раздел `/dev/sdb2`, вы ошибочно нажмете `t` вместо `y`, установка оборвется, и вам придется всё начинать с нуля.

## Пароль не нужен

По умолчанию, пользователь `root` не требует пароля. Вы можете скопировать `setup`, чтобы запустить мастер настройки на базе *Curses*. Он позволяет задать основные параметры, такие как имя хоста и часовой пояс, настроить сетевые карты и выбрать, какие службы будут запускаться при загрузке. Правила брандмауэра можно создавать из той же утилиты настройки, однако программа не дает возможности просмотра или изменения конфигурации по умолчанию. Для внесения изменений придется вручную открывать каждый файл конфигурации, будь то сетевые настройки или брандмауэр. Скрипт сохранения конфигурации (save-config) теперь также считывает файл `/etc/sysconfig/save-config.conf`, чтобы определить, какие файлы сохранять.

Кроме того, в дистрибутиве отсутствует утилита *ifconfig*. Вместо нее применяется

инструмент *ip*, который употребляется для настройки сетевых устройств. К сожалению, в неожиданно скудной документации на сайте не упоминается, что утилита *ifconfig* недоступна, и не сказано, какой инструмент можно использовать взамен нее. Не предусмотрен и журнал изменений по новым версиям. Пользователи должны распаковать tar-архив и перейти в каталог `docs`, чтобы получить доступ к файлу изменений, где перечислены все. Аналогичным образом, `sources.lst` предоставляет список программ.

Экосистема Linux может похвастаться множеством превосходных серверных дистрибутивов, таких как Debian и Slackware. Для развертывания брандмауэра и маршрутизатора есть такие проверенные функциональные варианты, как Zentyal, ClearOS и Untangle, которые явно легче в настройке и управлении. **LXF**



## Свойства навскидку

### Надежная конфигурация

Поскольку файл конфигурации хранится на USB-накопителе, при перезагрузке он считывается, давая готовую к работе машину.

### Многофункциональность

Этот облегченный дистрибутив пригоден для использования с File, Mail, Proxy, DNS и разными другими серверами.

## LINUX FORMAT Вердикт

### Devil Linux 1.8.0

Разработчик: Heiko Zuerker  
 Сайт: [www.devil-linux.org](http://www.devil-linux.org)  
 Лицензия: GPLv2

Функциональность	5/10
Производительность	6/10
Удобство в работе	4/10
Документация	4/10

» Предполагается, что Devil Linux — дистрибутив, разработанный администраторами для админов. Но только самых продвинутых.

## Рейтинг 5/10

# AMD Ryzen 5 1600X

Взяв новую шестиядерную красотку AMD покататься, Джереми Лэйрд удивился, что у него сносит крышу...

## Спецификация

- » Разъем AM4
- » Разрядность 64 бит
- » Тактовая частота 3,6 ГГц
- » Разгон 4 ГГц
- » Ядра 6
- » Потоки 12
- » Техпроцесс 14 нм FinFET
- » Кэш L1 576 КБ, L2 3 МБ, L3 16 МБ
- » Память DDR4-2666
- » Каналы 2
- » PCIe 16+4
- » Тепло-выделение 95 Вт
- » Виртуализация AMD-V, AMD-Vi



» Нега-ядерный — это новый quad-ядерный? Только говорящие на латыни знают...

**П**опробуйте это неотразимое предложение. У 1600X шесть ядер и поддержка 12 программных потоков, тогда как у процессора Intel Core i5-7600K с аналогичной ценой — четыре ядра и всего паршивых четыре потока. И как следствие того, что 1600X является лучшей шестиядерной моделью, он еще и работает на 3,6 ГГц номинально и 4,0 ГГц турбо. Другими словами, он выглядит как очень хорошо сбалансированный чип — и всё за £249. Итак, это процессор среднего класса, который мы все ждали, верно?

По большей части — да. Когда дело открыто доходит до многопоточной производительности, он всаживает пулю в голову Intel Core i5. Возьмем тесты *Cinebench*: Core i5-7600K обходится 663 баллами. А Ryzen 1600X? Солидные 1223 балла — почти вдвое больше производительности Core i5.

Признаём, что Core i5 отчасти сохраняет лицо в однопоточной версии *Cinebench*: 179 баллов против 159 баллов у 1600X. Однако чип AMD на первый взгляд выглядит лучшим компромиссом. В других областях, хоть результаты не столь зрелищны, AMD всё же одерживает неплохие победы. Он перемалывает высококачественное кодирование видео при 22 кадрах в секунду, по сравнению с 16 кадрами в секунду у i5, и проходит тест рендеринга всего за три минуты и 46 секунд. Это на добрых две минуты быстрее, чем 7600K.

## Параллели

Проще говоря, если вам нужен процессор для создания контента или действительно какая-либо нагрузка, основанная на параллелизме, то это даже не гонка. Intel как конкурент ужасно, безнадежно проигрывает. Это правда, даже с учетом того факта, что процессор Ryzen несколько более прожорлив: дополнительные 10 Вт на стене, безусловно, окупаются.

Предложение Ryzen чуть менее убедительно на арене игр. 1600X выглядит конкурентоспособным, когда речь идет о средних частотах кадров: он выполняет свое обещание насчет 4 ГГц. Однако субъективный опыт говорит немного другое. *Total War: Attila* — отличный пример этого. При работе на процессоре Intel всё супергладко, но с процессором Ryzen при тех же настройках и с использованием той же видеокарты почти везде наблюдается заметное дрожание изображения.

А причина — в чем? Ну, налицо много вариантов. Во-первых, мы использовали графический процессор Nvidia, и имеются, по крайней мере, признаки того, что драйвер Nvidia плохо оптимизирован для архитектуры Ryzen. Есть также вопросы к базовой архитектуре Ryzen, которая состоит из пары четырехъядерных модулей. Вкратце, во взаимодействии между двумя четырехъядерными модулями присутствуют задержки, что может потребовать отдельных

изменений как в ОС, так и в приложении, ради предотвращения сбоев в производительности. Ryzen — совершенно новый бренд, так что немало программ еще предстоит изменить, с целью учесть подобные проблемы.

Конечно, большинство программ отлично работает без каких-либо изменений. Жаль, что основным классом приложений, который, кажется, особенно чувствителен к архитектурным нюансам Ryzen, оказались игры. Но при малом пространстве для разгона, всё указывает на то, что стандартный AMD Ryzen 5 1600 лучше, чем эта модель X. **LXF**

## LINUX FORMAT Вердикт

### AMD Ryzen 5 1600X

Разработчик: AMD  
Сайт: [www.amd.com/ryzen](http://www.amd.com/ryzen)  
Цена: £249

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

» Поразительная производительность для такой цены, но для многих более дешевая модель не X будет выгодной.

Рейтинг **8/10**

# Full Throttle

Большой фанат ношения кожаных курток и махания кулаками, Энди Келли разбирается, отвечает ли эта игра его запросам.

## Спецификация

- » **Процессор** Intel Core 2 Duo, AMD Athlon X2 2,8 ГГц
- » **Память** 4 ГБ
- » **Графический процессор** Nvidia GeForce GTX 260, ATI Radeon 4870 HD, Intel HD 4000 (OpenGL 3.3 или выше)
- » **На диске** 8 ГБ

**Н**е одну головоломку в *Full Throttle Remastered* можно решить, просто пнув что-нибудь, что прекрасно иллюстрирует ее отличия от большинства квестов, устроенных по принципу наведи-и-щелкни. Протагонист Бен — жесткий, брутальный байкер. Качок в кожаной куртке с подбородком, об который можно полировать камни. То же можно сказать и об игре в целом, от головоломок и сюжетной линии до саундтрека в стиле рок.

Но главное в игре — то, что хотя Бен действительно крепкий орешек, знающий свою силу в бою, у него также есть и сердце. Если *Monkey Island* был одой романтической идее пиратства, *Full Throttle* [Полный газ] — то же самое для байкерства. Свобода, аутентичность и крутые кожаные куртки здесь ощущаются лучше, чем при чтении *Ангелов Ада* [Hell's Angels] Хантера С. Томпсона или просмотре *Сынов Анархии* [Sons of Anarchy].

Мир Бена не назовешь обширным: он несколько раз упоминает, что его дом — это его байк; но и этот мирок — под угрозой. Бен в бегах, поскольку его судят за преступление, которого он не совершал, а что еще хуже, последний производитель мотоциклов в стране, Corley Motors, прекращает



» На все графические объекты был нанесен новый слой цифровой краски.



» Угрюмый тон байкера Бена звучит даже лучше (не сочтите за стилистическую ошибку).

их выпуск, переходя на создание летающих мини-взнов. Игр про конец света много, но *Full Throttle* — это конец света для Бена.

Глубокий монотонный голос Бена, наследие покойного Роя Конрада [Roy Conrad], придает этому персонажу индивидуальности. Он идеально соответствует этой роли, и его невозмутимая манера означает, что в игре полно убийств. В ремастеринге он звучит еще лучше. Double Fine нашли аудиозаписи первоначальных сессий записи голоса, и даже странно слышать все эти знаменитые фразы без фонового шума.

Это был первый квест LucasArts после *Day of the Tentacle*, и отличие кардинально. Головоломки стали гораздо проще, интерфейс более упорядочен, а визуально всё гораздо богаче и кинематографичнее. Вместо старого перечня возможных ответов в нижней части экрана, по щелчку на объекте появляется горящий череп с различными опциями, которые отражают характер Бена: самые востребованные — это кулак в перчатке и кожаный ботинок.

Недостатком такого упрощенного дизайна является то, что игра кажется слишком короткой и проходится за пять-шесть часов, в зависимости от того, на скольких головоломках вы застряли. Заодно это означает, что по сравнению с большинством квестов она является и более динамичной и захватывающей. И уже пост-фактум думаешь, что такой, пожалуй, и должна быть игра про парня вроде Бена. Сильные стороны *Full Throttle* — это сюжет, персонажи, атмосфера и графика, но не ее головоломки. И знаете, порой этой хорошо. Но не исключено, что многие, в особенности же те, кто

не испытывает ностальгии по данной теме, будут слегка разочарованы процессом.

Что касается ремастеринга визуальных эффектов, ясно, что художники Double Fine приложили много усилий, чтобы перекрасить всё игровое пространство и видеовставки. Но, честно говоря, пиксельная графика оригинала показалась нам приятнее, и в конце концов, что удивительно, мы доигрывали в ней.

Эпизод Mine Road, где вы участвуете в байкерских боях в стиле *Road Rash*, выглядит жутко устаревшим. Но помимо одной неудачной мини-игры, возвращение *Full Throttle* пришлось нам по сердцу. Его стильный, в духе *Безумного Макса* [Mad Max] мир байкеров, дальнобойщиков и зверюшек на батарейках полон очарования, а сюжет хоть и недолговечен, но интересен до самого горько-сладкого конца. **LXF**

## LINUX FORMAT Вердикт

### Total War: Warhammer

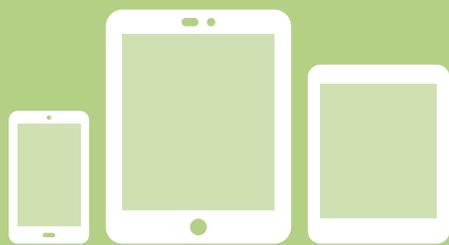
**Разработчик:** Double Fine  
**Сайт:** <http://fullthrottle.doublefine.com>  
**Цена:** £ 10,99

<b>Сюжет</b>	5/10
<b>Графика</b>	8/10
<b>Увлекательность</b>	4/10
<b>Оправданность цены</b>	8/10

» Кинематографичный, мощный, но короткий квест, любовно отделанный для нового поколения.

**Рейтинг 8/10**





## НОВИНКИ NOKIA

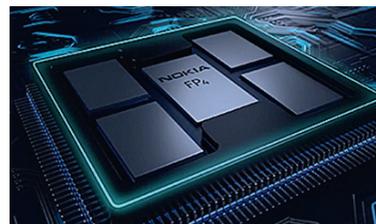
# Скорость и безопасность будущего

Представлены самый быстрый в мире процессор для сетевого оборудования и роутеры на его основе.

Согласно последним оценкам специалистов Nokia и Cisco Systems, в течение ближайших пяти лет объем интернет-трафика вырастет втрое: по прогнозам Nokia, к 2022 г. его общий объем достигнет 330 экзабайт (330 млн терабайт) в месяц, а Cisco полагает, что рост составит 24% в год, с 96 экзабайт в месяц в 2016 г. до 278 экзабайт в месяц в 2021 г. Основными источниками такого роста являются облачные сервисы, мобильные сети 5G и интернет-вещание. В недавнем исследовании Cisco о безопасности IoT содержится прогноз, что к 2020 г. в мире будет насчитываться 50 млрд подключенных устройств; Nokia считает, что к 2025 г. будет создано 100 млрд устройств IoT (по сравнению с 12 млрд в 2017 г.). Cisco полагает, что количество DDoS атак на такие устройства возрастет с 1,25 млн в 2016 г. до 3,1 млн

в 2021 г. Прогноз Nokia более пессимистичен: 10 млн атак только в 2017 г. Nokia считает, что наступает время новых высокопроизводительных решений, способных, помимо прочего, нейтрализовать проблему безопасности. Новый чип Nokia, FP4, основанный на уже используемом чипе FP3, обрабатывает до 2,4 терабит данных в секунду и передает статистику трафика во внешнюю систему аналитики. Плата с несколькими FP4 составляет основу линейной карты с пропускной способностью 12 Тбит/с; такая карта позволит новому роутеру обрабатывать трафик в 6 раз быстрее существующих моделей. Кроме того, по словам СТО подразделения IP/Optical Nokia Стива Фогельзанга [Steve Vogelsang], эти карты могут ставиться в оборудование до 10-летней давности: тогда пропускная способность, например, модели

► **Nokia FP4, первый в мире сетевой процессор со скоростью передачи данных 2,4 Тбит/с, позволяет внедрять масштабные, безопасные и функциональные IP-сети, не жертвуя производительностью.**



7750 SR-12 Service Router будет удвоена, а чуть более новой 7750 SR-12e даже утроена. Дополняющая линейку новая модель Nokia 7750 SR-14s с производительностью до 144 Тбит/с поступит в продажу в IV квартале; одновременно Nokia представила усовершенствованную модель 7950 XRS-XC Extensible Routing System, оптимистично описываемую как "petabit-class", с фактической спецификацией до 576 Тбит/с (т.е. чуть более половины петабита).

## ИНТЕРНЕТ ВЕЩЕЙ

# Кадры для цифровой экономики

В Московском технологическом университете стартует «IoT Академия Samsung».

В Москве состоялась церемония подписания соглашения о сотрудничестве между Samsung Electronics и институтом Информационных технологий — структурным подразделением Московского технологического университета (МИРЭА). В рамках соглашения на базе института в сентябре 2017 года будет запущен новый образовательный проект «IoT Академия Samsung», направленный на подготовку специалистов в быстроразвивающейся области информационных технологий — среде Интернета вещей.

Проект расширяет деятельность компании в области технического образования, которая сейчас представлена программой для старшеклассников «IT ШКОЛА Samsung». Благодаря новому проекту студенты МИРЭА получают возможность пройти

годовой учебный курс, построенный на изучении реальных кейсов по внедрению Интернета вещей, и разработать собственные прототипы устройств. «IoT Академия Samsung» — это уникальная возможность для учащихся приобрести практические

**Это уникальная возможность приобрести практические знания.**

знания в новой сфере информационных технологий и попробовать себя в решении актуальных задач.

Samsung предоставит для проекта учебные и методические материалы, разработанные их Исследовательским центром,

а также расширенный комплект оборудования для оснащения IoT-лаборатории, включая наборы для быстрого прототипирования, микрокомпьютеры Samsung ARTIK, разработанные специально для IoT-приложений, и другие высокотехнологичные устройства. Samsung также обучит преподавателей, которые будут работать на проекте, и окажет методическую поддержку в реализации курса. По завершении учебного курса Samsung проведет конкурс проектов учащихся.

Пилотный этап обучения по программе «IoT Академия Samsung» начнется в сентябре 2017 г. на базе Института информационных технологий МИРЭА. Весной 2018 г. состоится первый студенческий конкурс проектов, и в будущем такие конкурсы будут проходить ежегодно.

## ИНТЕРНЕТ ВЕЩЕЙ

# Масштабные планы Qualcomm

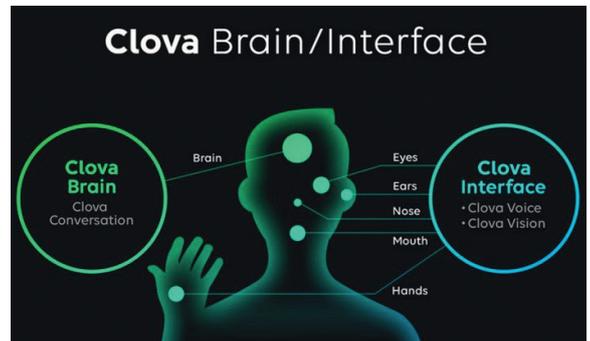
Процессоры Qualcomm будут управлять устройствами IoT, использующими сервис распознавания речи Clova.

Компании Naver и Qualcomm объявили о стратегическом партнерстве, в рамках которого создаваемые южнокорейским поисковым гигантом устройства для Интернета вещей будут оснащаться SoC от Qualcomm; кроме того, AI-платформа Clova (Cloud Virtual Assistant), совместно разработанная Naver и ее японским подразделением Line, позднее будет расширена на смартфоны Qualcomm. «Голосовые ассистенты» «разговорились»: Clova способна осуществлять поиск информации, подбирать музыку, выполнять письменный и устный переводы, вести диалоги на английском языке; необычной функцией для систем подобного рода является умение Clova вести «разговор по душам».

До конца года Line планирует выпустить смарт-колонку Wave с голосовым ассистентом Clova (по функциональности схожую с Apple HomePod), а затем в приложении

Line Chat появится интеллектуальный помощник Champ. В настоящее время Wave оснащена четырехъядерным Qualcomm Snapdragon 212 APQ8009 1,3 ГГц.

Qualcomm всё большее внимание уделяет IoT. Так, в мае компания представила платформу Mesh Networking Platform на сетевом SoC IPQ40x8/9, которая позволит создать новое поколение mesh-систем и расширит использование сетевых сервисов в домашних условиях. Mesh-платформа включает в себя набор Qualcomm IoT Connectivity feature suite, обеспечивающий одновременное использование технологий Wi-Fi, Bluetooth, CSRMesh и технологий на основе стандарта 802.15.4; набор Wi-Fi Self-Organizing (SON), обеспечивающий полную зону покрытия Wi-Fi; облачные средства диагностики и аналитики. В июне Qualcomm анонсировала новую линейку продуктов, включающую платформы audio SoC и smart audio, а также чип-



► Clova располагает мощнейшим инструментом подбора информации, построенном на интеграции поисковой базы данных Naver и AI-технологий.

сет-усилитель для наушников и колонок. Интеллектуальная звуковая платформа оптимизирована для работы с AI-колонками, использующими чипы APQ8009 и APQ8017. Такие устройства поступят в продажу в третьем квартале.

## ТЕЛЕКОММУНИКАЦИИ

## Япония + Китай = ...

SoftBank и ZTE начинают «полевые испытания» сети 5G.

Ширится японо-китайское сотрудничество: вслед за NTT DoCoMo и Huawei, китайский провайдер решений для телекоммуникационных технологий ZTE Corporation и японская телекоммуникационная корпорация SoftBank подписали партнерское соглашение о запуске в Токио пилотной зоны сети 5G в диапазоне суб-6-ГГц, на частоте 4,5 ГГц. SoftBank, третий по величине сотовый оператор Японии, в реальных условиях Токио протестирует технологию 5G, используя сетевые решения ZTE. Сотрудничество ZTE и SoftBank началось с исследований и разработок в области pre-5G технологий, в частности, Massive Multiple-Input Multiple-Output (Massive MIMO); теперь партнерство расширится, охватывая, в том числе, спектр 5G New Radio (NR).

Свои планы выпуска 10-гигабитных базовых станций 5G для диапазонов mmWave и sub-6GHz, соответствующих стандартам 3GPP и 5G NR, ZTE анонсировала еще

в феврале. Тогда же ZTE объявила о достижении максимальных для 4G скоростей 2,6 Гбит/с во время демонстрации технологии Massive MIMO для передачи данных по стандарту LTE FDD, а также одновременном соединении восьми терминалов 4G. В конце прошлого года ZTE представила запатентованное решение FDD-LTE Massive MIMO, основанное на чипе собственного производства для обработки вектор-

### Компании продолжают тестирование ключевых технологий 5G.

ных данных MSC2.0, после испытаний его с China Unicom и China Telecom. Выпущенный ZTE в феврале совместно с Intel модульный 5G IT baseband unit (BBU), благодаря применению программно-определяемых сетей (SDN) и виртуализации

сетевых функций (NFV), совместим с сетями 2G, 3G, 4G и pre-5G, плюс поддерживает сети CloudRAN, распределенные RAN, централизованные и распределенные устройства 5G.

Конкурирующий с SoftBank японский провайдер NTT DoCoMo совместно с китайским сетевым гигантом Huawei в ноябре минувшего года завершил в Йокогаме крупномасштабное полевое испытание технологии 5G в диапазоне частот 4,5 ГГц, достигнув пропускной способности 11,29 Гбит/с и латентности менее 0,5 мс. Компании продолжают тестирование ключевых технологий 5G: Massive MIMO, смешанных численных данных с использованием f-OFDM, совместно работающих технологий Sparse Code Multiple Access (SCMA), Polar Code и f-OFDM. В области 5G NTT DoCoMo сотрудничает также с Nokia: в начале прошлого года достигнута скорость загрузки 2 Гбит/с на частоте 70 ГГц в спектре mmWave. LXF

# Tizen: Цикл жизни

Лада Шерышова рассматривает модель нативного приложения ОС Tizen и управляет его жизненным циклом.



Наш эксперт

**Лада Шерышова** долгие годы работала на коммерческие корпорации, создавая промышленные высоконадежные информационные системы. Но пришло время сбросить оковы и применить свои знания и опыт в работе со свободным ПО.

**М**одель нативного приложения Tizen управляет жизненным циклом приложения и системными событиями. Управление жизненным циклом осуществляется на базе платформы Application API, которая позволяет осуществлять взаимодействие между приложениями и операционной системой. Модель нативного приложения Tizen предполагает, что приложение воспринимает и обрабатывает определенные события, которые переводят его в соответствующее состояние. Для этого приложение должно запускать основной цикл событий, управление которым осуществляется с помощью специальных функций, предоставляемых Application API.

Tizen поддерживает как UI-приложения, которые имеют графический пользовательский интерфейс, так и сервисные приложения, которые не имеют графического интерфейса. При необходимости UI и сервисные приложения могут быть упакованы вместе. Однако, такой комбинированный пакет может иметь несколько сервисных приложений и только одно UI-приложение. Каждое приложение в пакете подчиняется своему собственному жизненному циклу и может быть идентифицировано своим собственным ID.

## Состояния нативного приложения Tizen

Нативное приложение Tizen может находиться в одном из 5-ти состояний, определяемых Application API:

- » **READY** — приложение запущено.
- » **CREATED** — приложение запускает основной цикл.
- » **RUNNING** — приложение работает и видно пользователю.
- » **PAUSED** — приложение работает, но не видно пользователю (приостановленное состояние).
- » **TERMINATED** — приложение завершено.

## Функции обратного вызова

Переход из одного состояния в другое осуществляется через так называемые callback-функции или функции обратного вызова (ФОВ). ФОВ — это функция, которая передается другой функции в качестве параметра, а та, в свою очередь, вызывает переданную функцию. Точка внутри содержащей функции, в которой вызывается ФОВ, как раз и называется «обратным вызовом». Приложение должно соответствующим образом реагировать на каждое изменение состояния. Application API имеет 2 класса ФОВ для управле-

ния состояниями приложения: функции жизненного цикла приложения и системные функции. К ФОВ жизненного цикла нативного приложения относятся:

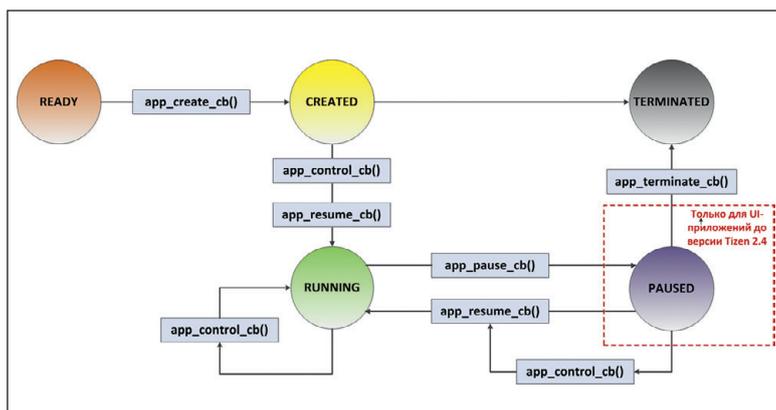
- » **app\_create\_cb()** Используется для выполнения необходимых действий перед запуском основного цикла. Вызывается после функции `ui_app_main()` и используется для инициализации графического интерфейса.
- » **app\_pause\_cb()** Делает окно приложения невидимым и переключает в режим экономного использования ресурсов.
- » **app\_resume\_cb()** Возобновляет работу, делая окно приложения снова видимым.
- » **app\_terminate\_cb()** Сохраняет работу, освобождает ресурсы и выполняет выход из приложения.
- » **app\_control\_cb()** Срабатывает, когда приложение выполняет какие-либо действия, и позволяет приложению получать дополнительную информацию о том, почему оно было запущено и с какими параметрами. Функция может вызываться несколько раз в течение «жизни» приложения и отображает на экране требуемые действия. Application API предоставляет следующие функции управления приложением:

- » **app\_control\_get\_operation()** Возвращает строку, описывающую, для какой ориентации было запущено приложение.
- » **app\_control\_get\_mime()** Возвращает тип данных MIME, такой как `image/jpg`.
- » **app\_control\_get\_app\_extra\_data()** Возвращает строковое значение, связанное с заданным ключом.
- » **app\_control\_get\_app\_extra\_data\_array()** Возвращает строковый массив, связанный с заданным ключом.

К системным ФОВ относится функция `app_event_cb()`, которая управляет следующими системными событиями:

- » **APP\_EVENT\_LOW\_MEMORY** Отвечает за сохранение данных в оперативной памяти устройства или запоминающего устройства для исключения потери данных в случае, когда процесс Tizen Low Memory Killer принудительно завершает приложение для освобождения памяти. Также функция удаляет из оперативной памяти кэшированные данные.
- » **APP\_EVENT\_LOW\_BATTERY** Отвечает за сохранение данных в оперативной памяти устройства или запоминающего устройства во избежание потери данных в случае выключения питания. Функция может также использоваться для остановки процессов, интенсивно использующих ресурсы CPU, или процессов, потребляющих значительное количество питания для экономии оставшейся электроэнергии.
- » **APP\_EVENT\_DEVICE\_ORIENTATION\_CHANGED** Отвечает за изменение ориентации дисплея в соответствии с ориентацией устройства в пространстве.
- » **APP\_EVENT\_LANGUAGE\_CHANGED** Отвечает за обновление дисплея при смене языка.
- » **APP\_EVENT\_REGION\_FORMAT\_CHANGED** Отвечает за обновление дисплея при смене часового пояса.
- » **APP\_EVENT\_SUSPENDED\_STATE\_CHANGED** Отвечает за выполнение необходимых действий перед переходом в приостановленное состояние или после выхода из него (поддерживается, начиная с версии Tizen 2.4).

Для прослушивания системных событий используется функция `ui_app_add_event_handler()`.



» Рис. 1. Переходы между состояниями нативного приложения Tizen.

## Модель жизненного цикла нативного приложения Tizen

Итак, нативное приложение Tizen может находиться в одном из нескольких состояний, переходы между которыми инициируются с помощью соответствующих ФОВ (рис. 1).

Обычно приложение запускается пользователем из программы запуска или другим приложением. При запуске приложения выполняется функция `app_create_cb()`, приложение переходит из состояния READY в состояние CREATED и инициируется основной цикл событий. Запущенное приложение, как правило, открывается поверх всех окон, имеет фокус и состояние RUNNING.

Когда приложение теряет фокус, вызывается функция `app_pause_cb()`, которая переводит приложение в приостановленное состояние (PAUSED). В этом состоянии приложение не закрывается, а продолжает работать в фоновом режиме.

Приложение может быть приостановлено в случаях, когда:

- » Из этого приложения по запросу запускается новое приложение.
- » Пользователь хочет перейти на домашний экран.
- » Происходит системное событие (например, входящий телефонный вызов) или вызов резидентного приложения с более высоким приоритетом, которое становится активным и временно скрывает ваше приложение.
- » Срабатывает оповещение для другого приложения, которое перемещается поверх всех окон и скрывает ваше приложение.

Приложение, находящее в состоянии PAUSED, работает с ограниченными ресурсами CPU, и в этом состоянии Application API не позволяет выполнения фоновых приложений, за исключением таких приложений, как Media и Download, которые должны работать в фоновом режиме.

Когда приложение становится снова видимым, вызывается функция `app_resume_cb()`, и приложение снова переходит в состояние RUNNING. Это происходит, когда:

- » Ваше приложение запускается другим приложением (например, Task Navigator, которое отображает все запущенные приложения и позволяет пользователю выбрать любое приложение для запуска).
- » Все приложения поверх вашего приложения в окне завершились.
- » Для приложения срабатывает оповещение, которое переносит его вперед и скрывает другие приложения.

До версии Tizen 2.4 сервисное приложение, не имеющее графического интерфейса, не могло переходить в приостановленное состояние. Начиная с версии 2.4, сервисное приложение может быть переведено в состояние PAUSED при условии, что для него указана категория фонового приложения. Однако, когда UI-приложение, упакованное вместе с сервисным приложением, запускается в фоновом режиме, сервисное приложение также рассматривается как фоновое и может быть запущено без указания категории фонового приложения.

Когда происходит выход из приложения, вызывается функция `app_terminate_cb()`, и приложение переходит в состояние TERMINATED. Процесс завершения запускается, когда:

- » Само приложение запрашивает выход с помощью ФОВ `ui_app_exit()` или функции `service_app_exit()` для завершения основного цикла событий.
- » Приложение завершается из-за нехватки памяти.

## Программируем жизненный цикл нативного приложения

В Tizen IDE в Project Explorer в папке вашего проекта перейдите в папку `src`, откройте файл `<наименование приложения>.c`, например, `testbasicui.c`, и добавьте код, показанный на рис. 2.

- » Для использования функциональности Application API включите в свое приложение заголовочный файл: `#include "<наименование приложения.h>".` Например, `#include "testbasicui.h"`

```

#include "testbasicui.h"
...
int
main(int argc, char *argv[])
{
    appdata_s ad = {0,};
    int ret = 0;

    ui_app_lifecycle_callback_s event_callback = {0,};
    app_event_handler_h handlers[5] = {NULL, };

    event_callback.create = app_create;
    event_callback.terminate = app_terminate;
    event_callback.pause = app_pause;
    event_callback.resume = app_resume;
    event_callback.app_control = app_control;

    ui_app_add_event_handler(&handlers[APP_EVENT_LOW_BATTERY], APP_EVENT_LOW_BATTERY,
        ui_app_low_battery, &ad);
    ui_app_add_event_handler(&handlers[APP_EVENT_LOW_MEMORY], APP_EVENT_LOW_MEMORY,
        ui_app_low_memory, &ad);
    ui_app_add_event_handler(&handlers[APP_EVENT_DEVICE_ORIENTATION_CHANGED],
        APP_EVENT_DEVICE_ORIENTATION_CHANGED, ui_app_orient_changed, &ad);
    ui_app_add_event_handler(&handlers[APP_EVENT_LANGUAGE_CHANGED], APP_EVENT_LANGUAGE_CHANGED,
        ui_app_lang_changed, &ad);
    ui_app_add_event_handler(&handlers[APP_EVENT_REGION_FORMAT_CHANGED], APP_EVENT_REGION_FORMAT_CHANGED,
        ui_app_region_changed, &ad);

    ret = ui_app_main(argc, argv, &event_callback, &ad);
    if (ret != APP_ERROR_NONE) {
        dlog_print(DLOG_ERROR, LOG_TAG, "app_main() is failed. err = %d", ret);
    }
}
    
```

» Рис. 2. Создаем функцию `main()`.

2 Определите функцию `main()`. Она инициализирует Application API и запускает основной цикл событий с помощью функции `ui_app_main()`.

3 Перед вызовом этой функции создайте структурную переменную типа `ui_app_lifecycle_callback_s`, которая будет передаваться в функцию `ui_app_main()`. Например,

```
ui_app_lifecycle_callback_s event_callback = {0,};
```

4 Задайте функции обратного вызова жизненного цикла приложения (`create`, `pause`, `resume`, `terminate` и `control`), используя структурную переменную, которая была определена на предыдущем шаге — например,

```
event_callback.create = app_create;
```

и т.д.

5 Запустите прослушивание описанных выше системных событий с помощью функции `ui_app_add_event_handler()`, определив для каждого из них соответствующую функцию обратного вызова, например, для события `APP_EVENT_LOW_MEMORY` — функцию `ui_app_low_memory`:

```
ui_app_add_event_handler(&handlers[APP_EVENT_LOW_MEMORY], APP_EVENT_LOW_MEMORY, ui_app_low_memory, &ad);
```

И так — для каждого системного события.

6 Создайте каждую ФОВ для состояний приложения и системных событий, определенные в функции `main()`. Фрагменты кода представлены на рис. 3. LXF

```

static bool
app_create(void *data)
{
    /* Hook to take necessary actions before main event loop starts
       Initialize UI resources and application's data
       If this function returns true, the main loop of application starts
       If this function returns false, the application is terminated */
    appdata_s *ad = data;

    create_base_gui(ad);

    return true;
}

static void
app_control(app_control_h h)
{
    /* Handle the launch request */
}

static void
app_pause(void *data)
{
    /* Take necessary actions before application is paused */
}

static void
app_resume(void *data)
{
}

static void
ui_app_lang_changed(app_event_info_h event_info, void *user_data)
{
    /*APP_EVENT_LANGUAGE_CHANGED*/
    return;
}

static void
ui_app_orient_changed(app_event_info_h event_info, void *user_data)
{
    /*APP_EVENT_DEVICE_ORIENTATION_CHANGED*/
    return;
}

static void
ui_app_region_changed(app_event_info_h event_info, void *user_data)
{
    /*APP_EVENT_REGION_FORMAT_CHANGED*/
}

static void
ui_app_low_battery(app_event_info_h event_info, void *user_data)
{
    /*APP_EVENT_LOW_BATTERY*/
}
    
```

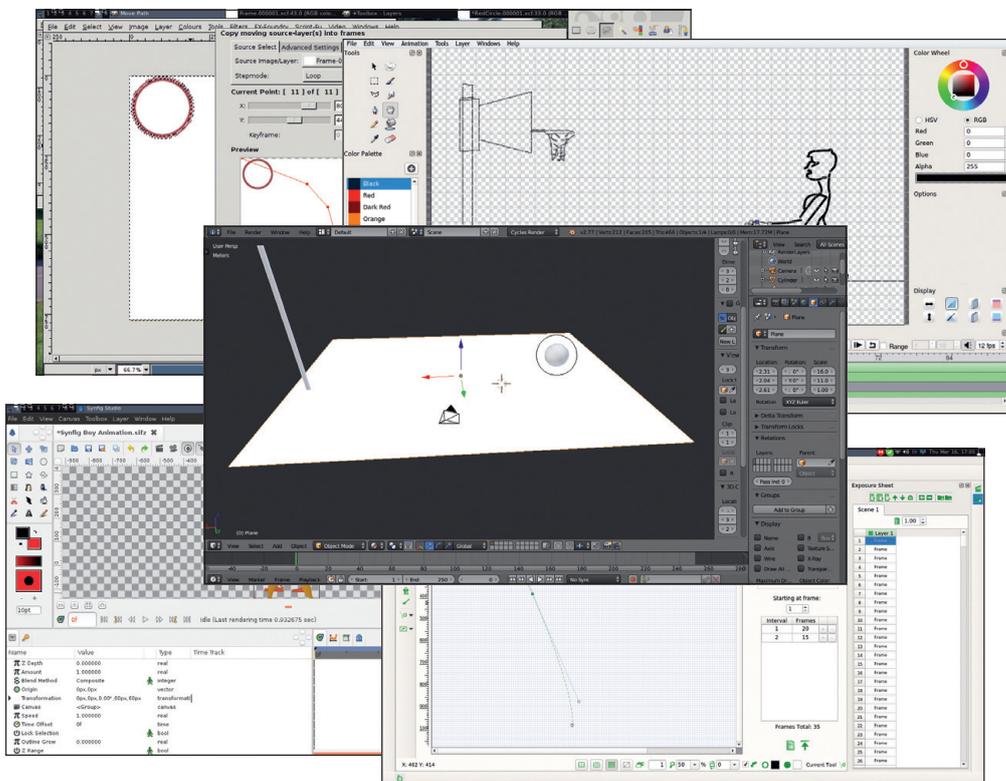
» Рис. 3. Создаем функции обратного вызова: фрагменты кода.

# Сравнение

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

## Средства анимации

Не оставляйте свои наброски томиться в блокноте. Матс-Теге Аксельссон выбирает правильное ПО для анимации, чтобы оживить ваше творчество.



### Про наш тест...

Мы сосредоточились на том, каковы программы в работе. Производительность была вторичным критерием, потому что сложный рендеринг требует отдельного оборудования. Основные функции, которые нас интересуют, относятся к тому, как вы хотите описать движение по экрану и через вашего персонажа. Мы создали сцену, где по экрану прыгает мяч. Это может показаться простым, зато позволяет увидеть, как программа делает его движения реалистичными. Мы также добавили фигуру, чтобы посмотреть, легко ли анимировать части тела.

Если только вы не перфекционист с избытком досуга, вам вряд ли захочется рисовать каждый кадр, поэтому мы также протестировали эффективность опорных кадров, позволяющих контролировать промежуточные кадры вашей анимации. То, насколько успешно вставлены ваши опорные кадры, может привести к успеху или провалу вашей анимации.

**В** этом месяце мы рассматриваем пять разных программ для анимации в Linux. А кроме того, поясним некоторые концепции, чтобы вы осознавали пользу функций, присутствующих в некоторых или во всех программах. Тестируемые программы предназначены для 2D, и для 3D, и используют разные принципы работы для достижения конечного результата. Ваш способ реализации анимационного проекта повлияет на ваш конечный выбор программы. Мы также рассмотрим скорость разработки для каждого пакета. Говоря об анимации, большинство воображает мультфильмы, где, как правило, антропоморфные

животные попадают в забавные ситуации. Это один из способов анимации, однако можно добиться намного более высокого уровня реалистичности: многие рекламные ролики созданы с помощью *Blender*. В официальной версии можно поклясться, что мороженое настоящее. А при виде шоколадной начинки у нас слюнки потекли!

У программ в нашем Сравнении амбиции скромнее. Большинство их создано для

анимации в стиле мультфильмов, поддерживается только 2D и нет интересных опций подсветки. Концепция опорного (ключевого [key]) кадра является основной, поскольку позволяет избежать необходимости рисовать каждый кадр вашей анимации. Мы рассмотрим, как с ним работает каждая программа, а также какие еще инструменты имеются в разных пакетах, тестируемых в нашем Сравнении.

### Наша подборка

- » Blender
- » GIMP
- » Pencil2D
- » Synfig
- » Tupi

**Способ реализации анимационного проекта влияет на конечный выбор программы.**

# Ход работы

## Как вы рисуете на цифровом холсте?

С годами процесс создания анимации сильно видоизменился, однако некоторые принципы остались прежними. В зависимости от программы для компьютеров адаптируются разные методы, имитирующие обычное рисование от руки.

В нашем примере описывается то, как можно импортировать изображения (предпочтительно в формате PNG) и привязывать их к контрольному слою, который поворачивает картинку. Возможны и многие другие трансформации, и даже больше, если ваш файл нарисован в векторной графике.

С помощью функции «bones [костяк]» вы можете трансформировать векторное изображение и разместить костяк в разных местах, создавая зону влияния вокруг костяка и в целом отмечая, на какую часть картинки должна воздействовать функция bones.

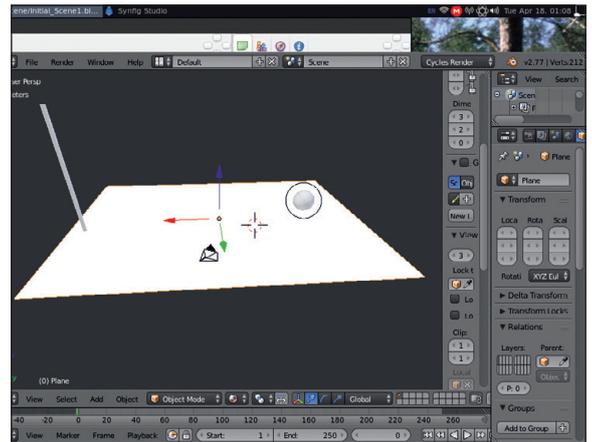
*Blender* использует тот же основной принцип, и тоже именует его «bones». Однако в *Blender* вы создаете «meshes [сетки]», которые представляют объекты во всех

трех измерениях. До известной степени *Blender* также имитирует реальный мир, при посредстве модуля физики. Этот модуль, помимо всего прочего, включает воду, огонь и дым.

Например, чтобы создать качающуюся цепь, зайдите в опцию *Torus primitive*, выделите половину и растяните. Далее выполните копирование, поворот на 90 градусов — и готово.

*Pencil2D* — инструмент попроще, где помощь в анимации ограничена созданием плавного движения объекта посредством рисования нескольких кадров, которые позднее появятся в анимации. Эта программа предназначена для покадровой анимации и использует слои на фоне, разные символы и синхронизацию звука с действием на экране.

В *Tupi* сохранена опция рисования каждого кадра с помощью функции *Onion*. Эта функция показывает несколько последних кадров «сквозь» ваш текущий кадр, как это делают традиционные художники-аниматоры, использующие световой стол и кальку.



➤ Подход в разных программах разный.

Есть также функция рисования объекта и затем отображение его пути, что удобно, например, когда через сцену бросают мяч.

*GIMP GAP* является расширением для *GIMP*. М в *GIMP* означает «манипуляции», однако он прекрасно подходит для рисования благодаря множеству имеющихся эффектов. При использовании *GIMP GAP* вы рисуете один кадр в каждом слое, и функция анимации покажет их в последовательности. Это прекрасно подходит для покадровой анимации существующего произведения искусства или сделанных вами фотографий.

## Вердикт

**Blender**  
★★★★★  
**Pencil2D**  
★★★★★  
**Synfig**  
★★★★★  
**GIMP**  
★★★★★  
**Tupi**  
★★★★★

» *Blender* и *Synfig* круче, но и для *Tupi* найдется применение.

# Возможности стиля

## Каков ваш стиль в анимации — радостно-дурашливый или стильно-крутой?

При выборе программы анимации в первую очередь следует учитывать стиль и требуемый уровень сложности.

Если вы хотите создавать анимацию в стиле мультфильмов, то вам явно понадобится не тот пакет, который применяется

для создания более серьезных и реалистично выглядящих работ.

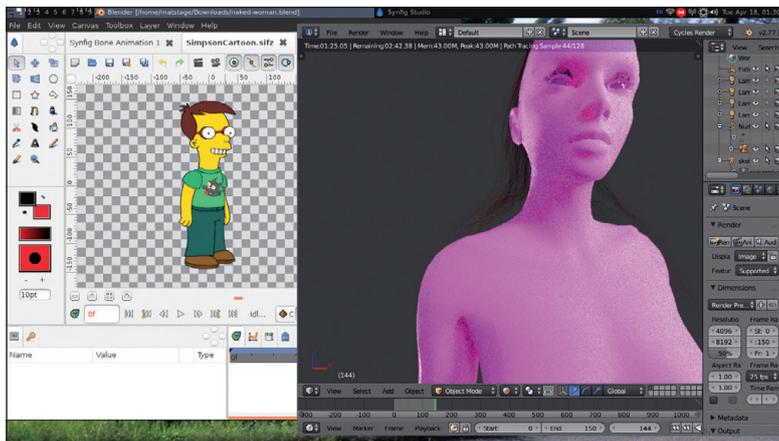
*Synfig* — программа для 2D-рисования, предназначенная для мультфильмов, с расширенными функциями для создания реалистичной среды. Но она отлично подходит для создания графических новелл.

*Blender* — 3D-среда, где вы скорее моделируете вашу сцену, нежели рисуете. Стиль может меняться от мультяшного до фотореалистичного.

Простой интерфейс *Pencil2D* мало чем поможет созданию анимации, хотя применяет опорные кадры. Это пакет покадровой анимации, и функции опорных кадров помогут создать плавное движение по экрану.

*Tupi* работает в 2D и в первую очередь предназначена для рисования мультфильмов. Здесь есть ряд карандашей и инструментов рисования, но нет световых эффектов и других инструментов рендеринга. Чтобы извлечь максимум из этой программы, надо быть серьезным художником. Однако функция раскадровки [storyboard] превратит ее в отличную программу планирования.

В *GIMP* масса расширенных функций для создания прекрасной графики. Но анимировать персонажей непросто: большая часть работы выполняется покадрово. *GIMP* поможет в создании крутящихся шаров и прочих анимационных эффектов из уже существующей картины или графики.



➤ Ваш стиль рисования повлияет на выбор программы.

## Вердикт

**Blender**  
★★★★★  
**Pencil2D**  
★★★★★  
**Synfig**  
★★★★★  
**GIMP**  
★★★★★  
**Tupi**  
★★★★★

» Для полноценного фильма — *Blender*. Остальные — для короткометражек.

# Простота в работе

Начать-то может быть легко, но далеко ли вы уйдете без помощи?

**Н**а первый взгляд все программы в нашем тесте выглядят, как художественные программы, но копнув глубже, вы найдете множество функций, чтобы оживить свое произведение.

Например, бесценны будут инструменты, которые сглаживают движение между опорными

кадрами. В *Blender* стоит начать с моделирования и риггинга, т.е. подготовки виртуального «скелета». На это уходит много времени, но о затратах вы забудете, увидев, как фигура движется по экрану.

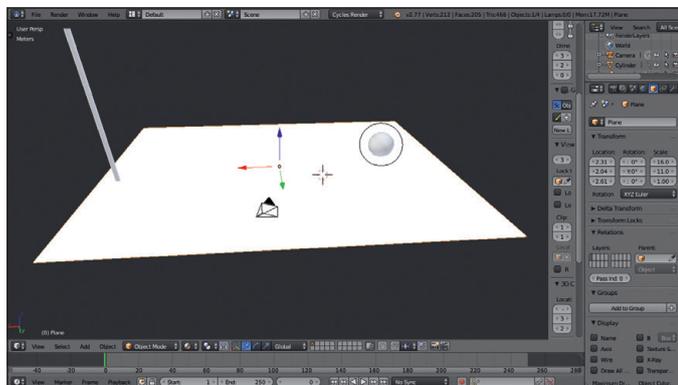
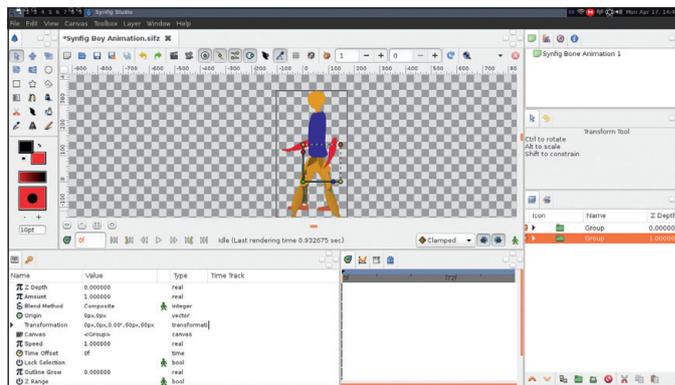
Другая проблема кроется в понимании назначения функций. Скажем, в программе *GIMP* каждый уровень является кадром.

При использовании *Synfig* каждый новый объект на экране является слоем, а точная настройка внешнего вида и того, что идет следом, требует нового рабочего слоя.

В *Blender* отсутствуют специальные слои функций — слои используются только для организации объектов.

## Synfig ★★★★★

Начальный экран *Synfig* смотрится зазывно, однако следите за использованием слоев. Верхний слой влияет на все нижележащие слои, поэтому разбейте их на группы, чтобы изменять только определенные объекты. Даже фон создается с помощью слоев. У каждого слоя есть параметры, управляющие его поведением и следующим слоем. В *Synfig* имеется много способов задать, что будет происходить с вашими фигурами, в зависимости от используемого вами типа файлов. SVG-файлы можно обрабатывать детально, а PNG-файлы — только поворачивать и менять их размер с помощью рисованной анимации. Чтобы добиться этого в *Synfig*, надо импортировать файлы PNG — по одному на каждую движущуюся часть — и затем определить точки поворота и настроить их в определенных опорных кадрах. При использовании SVG-файлов и функции bones можно обработать всё в деталях.



## Blender ★★★★★

*Blender* выглядит как ПО для продвинутых, но через пару часов вы начнете в нем разбираться. Быстрейший способ освоить *Blender* — начать проект и улучшать его по мере освоения новых технологий. Принцип работы *Blender* — добавлять простые формы, именованные примитивами [базовые геометрические фигуры], например, плоскости, шары и т.п., и затем моделировать их, превращая в требуемые фигуры посредством экструзии, масштабирования и соединения частей. Для получения реалистичной поверхности нужна функция materials. Чтобы поверхность выглядела хорошо, придется потрудиться. Инструмент для этой задачи именуется редактор узлов, а узлы — это на самом деле фильтры, которые можно смешивать и соединять для создания нужного материала. В *Blender* масса функций, например, кости, риггинг и материалы, и можно создавать как простые мультфильмы, так и фотореалистичные работы.

# Опции импорта и экспорта

Рисовать бывает проще в других программах, скажем, в *Inkscape*...

**П**ри создании анимации надо создать персонажей, фон и реквизит. Это могут быть фотографии, а может быть, вы решите сначала сделать рисунок в другой программе. Часто можно использовать реальные фотографии ради более реалистичных поверхностей. По вашему требованию ваша программа также осуществит рендеринг в разные форматы.

*Synfig* импортирует PNG, JPG для рисованной анимации и SVG для анимации методом моделирования, экспортирует изображения в ряд форматов и использует *ffmpeg*

для создания видео — но через систему модулей, так что можно подключить и другие видеопрограммы. Есть также функция синхронизации видео и звука под названием Paragayo, для которой нужен WAV-файл.

*Blender* импортирует большинство графических форматов, в том числе DXF-файлы из ваших чертежей CAD, и это делается с помощью модулей, которые следует подключить. Чтобы отправить изображение или клип в другую программу или опубликовать в Сети, сделайте рендеринг файла и выберите один кадр.

*Pencil2D* не работает с векторной графикой, но умеет импортировать последовательности. Можно применить *GIMP-GAP* для создания изначальной последовательности, а отредактировать ее в *Pencil2D*. Также можно импортировать WAV- и MP3-файлы в звуковую дорожку.

*Tupi* импортирует растровую и векторную графику и активы Paragayo. Можно использовать и свою любимую *GIMP* Palette.

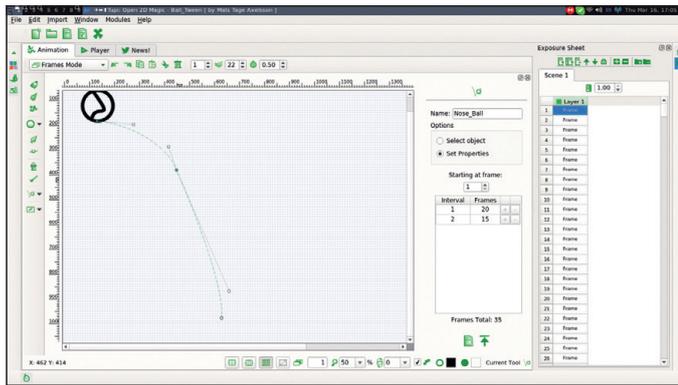
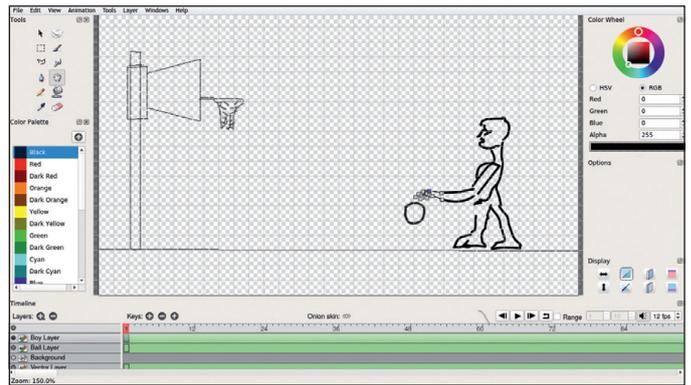
*GIMP* импортирует графические изображения (не векторные). Экспорт — в изображения и последовательности, или GIF.

## Вердикт

- Blender ★★★★★
  - GIMP ★★★★★
  - Synfig ★★★★★
  - Tupi ★★★★★
  - Pencil2D ★★★★★
- » Tupi удивляет функциями синхронизации звука и видео и импортом файлов Paragayo.

## Pencil2D ★★★★★

Начать работать в *Pencil2D* легко, но ваш прогресс может замедлиться по мере освоения более расширенных функций. Пока там нет функции «заполнения промежуточных кадров [tweening]», чтобы рисовать только опорные кадры, позволил *Pencil2D* самому заняться промежуточными стадиями. Вместе с тем, инструмент *Onion* работает хорошо, и легко скопировать кадр и сделать мелкие исправления для создания простой анимации ходьбы. Можно использовать фон, созданный в другой программе, или одно или более изображений, чтобы перерисовывать только персонажей и реквизит. Тут есть ряд преимуществ: например, можно создавать забавные эффекты в стиле мультиков, просто рисуя каждый кадр так, как вам надо. Самая интересная функция — *Opacity*, она работает для всего приложения. С ней вы можете взять любой просмотрщик изображений, поместить его за *Pencil2D* и обвести основные формы и края.

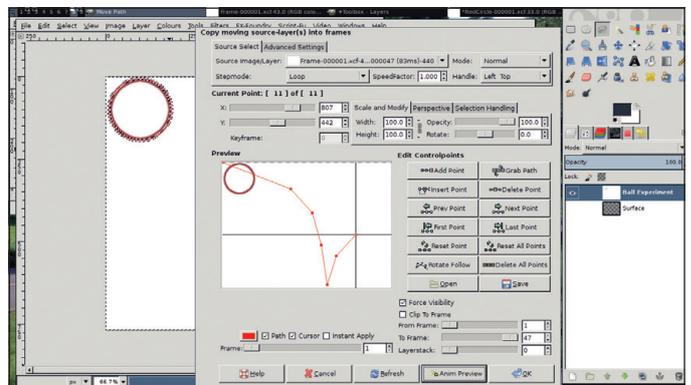


## Tupi ★★★★★

У *Tupi* простой интерфейс, и вы сможете рисовать своих персонажей так же, как это делается ручкой или карандашом. Есть поддержка ломаных линий и базовых элементов (эллипс, прямоугольник, прямая), что очень удобно для рисования простых фоновых объектов типа скамеек или домов. Функция *Tweening* в *Tupi* позволяет перемещать, поворачивать, масштабировать, сдвигать, затемнять и изменять цвета через анимацию, упрощая в вашей работе создание плавного перемещения. Одной из отличных функций является раскадровка, удобная для планирования анимации — даже если вы делаете анимацию в других программах. Набор функций *Tupi* ограничен, но они просты и помогут вам создавать анимацию быстро благодаря *Tweening*. Это придает программе более солидный вид, а с учетом того, что версия пока что только 0.2, я вижу в ней огромный потенциал в качестве инструмента мультипликатора.

## GIMP ★★★★★

В первую очередь этот пакет предназначен для обработки изображений с дополнением, однако дополнение для анимации само по себе очень мощное. Наиболее частое применение этого инструмента — создание текстовых эффектов, но если нарисовать в каждом слое один рисунок, то *GIMP-GAP* пригодится для эффектов подсветки и трансформации, что идеально при создании анимации. *GIMP-GAP* дает вам свободу делать с вашей графикой всё, что нужно, но анимирование предполагает множество копирований и вставок и затем отделки каждого кадра. Процесс анимации будет несколько странным, так как всё делается через плагин, так что подумайте о том, чтобы написать скрипты для многих действий, это поможет вам в создании анимации. Некоторая часть работы выполняется в интерфейсе, что сэкономит вам некоторое время при создании движения, позволяя поправить освещение и подобрать цвета.



# Качество документации

### Бросят ли вас на произвол судьбы после запуска программы?

**Ч**тобы действительно освоить рабочие инструменты, следует начать собственный проект — сделайте его небольшим, однако достаточно детальным, чтобы было интересно.

Файл подсказки и онлайн-руководства ускорят процесс вашего обучения; на беду, большинство программистов ненавидят писать документацию! Вот почему так важно выяснить, хорошо ли задокументирована программа, прежде чем делать свой выбор.

*Synfig* предлагает хорошую подборку руководств, хотя некоторые предполагают

у вас хорошее знание программы с самого начала. Иными словами, когда вы ознакомитесь с несколькими руководствами, вы будете знать инструменты вплоть до мелочей.

В *Blender* колоссальный набор функций, и они хорошо документированы. Кроме того, имеется немало руководств, как видео, так и текстовых.

Чтобы найти документацию по *Pencil2D*, вам придется немного покопаться, потому что на данный момент при попытке доступа к опции программы *Help* вас просто переведут на web-страницу руководств.

У *Tupi* имеется целый буклет подсказок по анимации, являющийся частью справочника. Рекомендую с ним ознакомиться, даже если вы не используете само приложение. Этот документ доступен онлайн, и если вы уделите время на ознакомление с уроками, вы освоите базовую технику, которая поможет вам овладеть искусством анимации.

Справочник пользователя *GIMP* объясняет все функции по одной, и очень удобен, если у вас возникнут затруднения с использованием определенного инструмента. Он также включает краткие руководства.

## Вердикт

- Tupi ★★★★★
- Blender ★★★★★
- GIMP ★★★★★
- Synfig ★★★★★
- Pencil2D ★★★★★

» *Tupi* полезен при освоении основ работы, прежде чем приступать к большим проектам.

# Отношения с сообществом

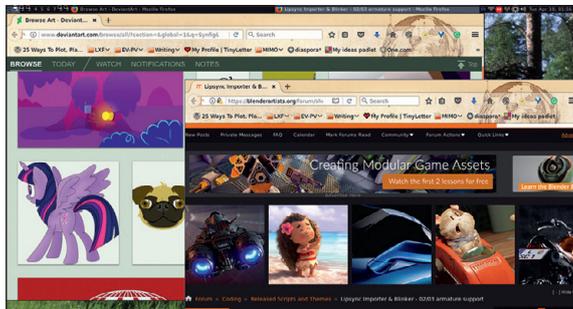
Анимация — искусство, и пусть вас заметят, по душе вы им или нет!

**Н**аши сообщества являются наиболее ценными ресурсами для сотрудничества в наших предприятиях и для поддержки и помощи при возникновении проблем.

Что касается творческих сообществ, они являются отличным местом демонстрации плодов вашего труда. Самое крупное — deviantART, где вы можете найти интересующую вас программу и посмотреть, что в ней создали другие. Это поможет вам оценить возможность достижения поставленных целей в анимации. Однако с самого начала вам нужна помощь с инструментами. Давайте посмотрим, куда обратиться за помощью насчет самих инструментов...

У *Synfig* отличная wiki, также весьма удобная при наличии вопросов по функциям и техникам. Есть также много руководств на YouTube, если вам понадобится дополнительная подсказка.

У *Blender* имеются как технические, так и специальные творческие сообщества;



» Сознайтесь, вы ждете одобрения сообщества!

есть даже те, кто зарабатывает на жизнь, создавая анимацию с помощью *Blender* и обучая его технике.

*Pencil2D* предлагает wiki на своем сайте, который умеренно активен. У программы много преданных пользователей на deviantART. Многие выбирают *Pencil2D* за его простоту. Ведь вы рисуете точно так же, как на бумаге.

*Tupi* родился в Колумбии, поэтому большая часть его контента — на испанском.

Впрочем, у программы не очень много функций, зато определены они хорошо, и имеющиеся на английском руководства достойно написаны.

У *GIMP* есть несколько сайтов, и на некоторых обсуждается анимация с помощью фильтров и расширение GAP. На них вы можете найти подсказки. Помните, что *GIMP* существует уже давно, и все, кто интересуется графикой и открытым кодом, хорошо знают эту программу.

**Вердикт**

- Blender ★★★★★
- Synfig ★★★★★
- GIMP ★★★★★
- Tupi ★★★★★
- Pencil2D ★★★★★

» Здесь блистает сообщество Blender, хотя немало сайтов, независимых от приложения.

# Активность разработки

Расцветет ваш проект или зачахнет?

**З**десь мы рассмотрим состояние разработки по каждому проекту. Некоторыми проектами занимается один человек, у которого руки чешутся поработать, а за некоторыми стоят целые организации. Например, *Tupi* поддерживается одной из школ в Колумбии.

У *Blender* имеется собственный фонд. И это позволило принять на работу персонал с полной занятостью, который исправляет ошибки, и отдельных разработчиков для новых функций. Фонд также сотрудничает с компаниями, у которых есть свои запросы, и те помогают финансировать весь проект.

Подобная организация дает любому проекту большие возможности для стабильной и растущей кодовой базы. Программы, которые не развиваются, очень скоро начнут тормозить ваш прогресс из-за отсутствия определенных функций, которые вам необходимы для создания ключевого эффекта.

Если взглянуть на развитие *Synfig*, активность по-прежнему остается на высоком уровне, хотя своего пика достигла в 2013 г. Это обеспечивает работу над существующими недостатками, однако неизвестно,

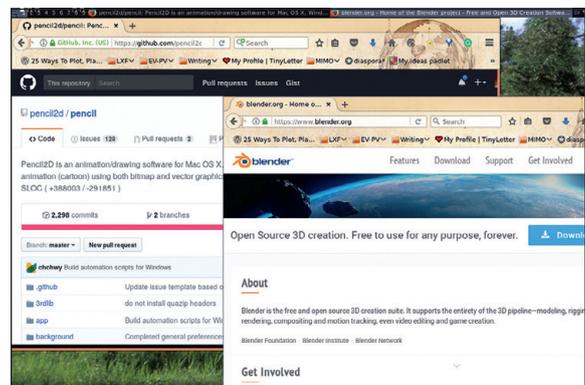
сколько новых функций будет разработано. Правда, с учетом поддержки от нескольких пользовательских организаций, этот проект кажется многообещающим.

*Blender* используется крупными студиями, и нет никаких сомнений, что разработчики будут и дальше добавлять новые функции и исправлять ошибки. У них также имеется собственный репозиторий, демонстрирующий большую активность, и на данном этапе активно рекламируется следующая версия.

Кроме того, для *Blender* имеется много дополнений. Поиск на Github показывает немалое количество тех из них, которые активны. Если у вас есть какие-то особые требования, поищите там плагины.

Разработка *Pencil2D* продолжается, но у него только один активный разработчик, хотя проект получает содействие через Github и еще от двоих разработчиков. Будем надеяться, они сумеют подстегнуть разработку этой простой в использовании программы.

Код *Tupi* вот уже много месяцев не получал добавок, но я рад сообщить, что его дорожная карта недавно обновилась, и при некотором везении новые разработчики



» Разработка при поддержке со стороны фондов, как правило, достигает зрелости, что делает ее более стабильной.

снова возьмется за дело. В этой программе предусмотрен ряд симпатичных функций для автоматизации процесса анимации, которые, к сожалению, отсутствуют в других имеющихся программах покадровой анимации.

Активность разработки *GIMP-GAP*, похоже, довольно низка, однако большая часть активности приходится на сам *GIMP*, что скрадывает усилия на фронте анимации. *GIMP* является уважаемым графическим пакетом, поэтому плагин тоже получает поддержку.

**Вердикт**

- Blender ★★★★★
- GIMP ★★★★★
- Synfig ★★★★★
- Pencil2D ★★★★★
- Tupi ★★★★★

» Blender побеждает, поскольку за ним стоит солидная организация.

## Лучший инструмент анимации

# Вердикт

**О**дин пакет явно выделяется благодаря обилию функций, стабильности и перспективе. И этот пакет — *Blender*, пригодный для создания как простой анимации, так и сложных физических эффектов.

Здесь в одной посуде собраны все инструменты, которые только можно вообразить. На самом деле, вы даже не узнаете об их существовании, пока не приметесь их отыскивать.

Фонд *Blender* даже проводит ежегодную конференцию по всем аспектам создания анимации с помощью этой программы. Это делает *Blender* сильнейшим участником; однако при выборе программы вы можете предпочесть функциям стиль. Важно также помнить, что это — 3D-пакет, и персонажи не рисуются, а создаются с помощью кубов и шаров, способных менять форму.

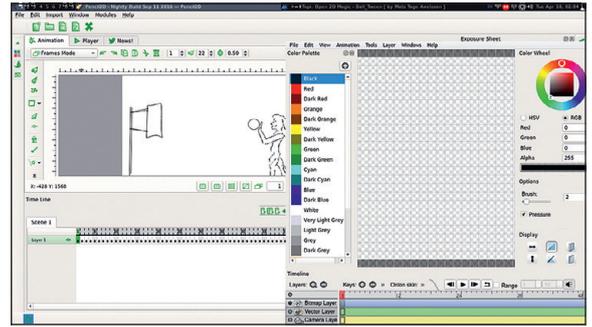
Иногда предпочтительнее простой интерфейс, имитирующий рисование на бумаге. И *Synfig*, и *Tupi* очень близко имитируют рисование на бумаге, благодаря чему

удобны, если вы переходите от рисования на обычной бумаге к цифровому холсту.

*Synfig* также позволяет делать эскизы в растровом поле и с помощью кривых Безье [Bezier]. Подобный подход на самом деле облегчает переход от рисования на бумаге больше, чем полное изменение техники рисования.

Опорные кадры позволяют нарисовать несколько кадров и дать программе возможность заполнить промежутки между ними. Однако данная техника может привести к механическим движениям или движениям с отставанием, так что перед публикацией будьте готовы точно прорисовать все детали — особенно в выражении лиц.

При использовании вида “onion” можно применять загодя подготовленные рисунки



и ивнести их, а затем создать новые кадры, меняя имеющуюся картинку. В *Tupi* это работает особенно хорошо, если начать с одной картинки для каждого объекта в вашей сцене. Затем вы сможете быстро заполнить свою сцену, в итоге создавая анимацию.

В *GIMP* можно использовать простое рисование контуров и затем раскрасить их с помощью множества функций и технологий слоев для добавления цветовых и прочих эффектов. Есть эффекты, позволяющие придать вашим рисункам фактуру разных материалов, например, дерева или металла.

» *Pencil2D* прост в освоении, но результаты примитивные.

### Иногда предпочтительнее простой интерфейс, имитирующий рисование на бумаге.

### I Blender ★★★★★

Версия: 2.77a Сайт: [www.blender.org](http://www.blender.org) Лицензия: GPL v3  
» Blender может привести вас в Голливуд, но подумайте о том, чтобы начать с более простых инструментов.

### IV Tupi ★★☆☆☆

Версия: 0.2 git7 Сайт: [www.maeffloresta.com/portal/](http://www.maeffloresta.com/portal/) Лицензия: GPL v2  
» У *Tupi* привлекательный интерфейс, но он пока что находится на очень ранней стадии разработки.

### II Synfig ★★★★★

Версия: 1.0.2 Сайт: [www.synfig.org](http://www.synfig.org) Лицензия: GPL v2  
» *Synfig* занимает второе место, потому что он только 2D и у него не столь продвинутые инструменты.

### V Pencil2D ★☆☆☆☆

Версия: 7.04, ночная сборка Сайт: [www.pencil2d.org](http://www.pencil2d.org) Лицензия: GPL v3  
» *Pencil2D* не хватает функций, так что на сегодняшний день ему бы не помешало внимание разработчика!

### III GIMP ★★★★★

Версия: 2.8.18 Сайт: [www.gimp.org](http://www.gimp.org) Лицензия: GPL v3  
» GIMP — стандарт в Linux, но анимация отнюдь не принадлежит к его сильным сторонам.

### Обратная связь

Как вы создаете анимацию в одной из этих программ? Или используете другую? Расскажите нам о своих предпочтениях на [linux@futurenet.com](mailto:linux@futurenet.com).

## Рассмотрите также...

**А**нимация — это всего лишь расширение рисования или иных визуальных творческих техник, так что начните с изучения этих дисциплин. Если вы уже умеете рисовать, то вы готовы идти дальше. Отличные программы для рисования — *Inkscape* и *Krita* со свежееоткрытым кодом, которая также годится для анимации.

Рассмотрите шансы применения *Clara.io* — веб-инструмента, который создает анимацию прямо у вас в браузере. Бесплатное членство подарит множество предварительно подготовленных сцен и час рендеринга. *Clara.io* во многом напоминает *Blender*.

Решившись освоить основы анимации, начните с курса академии Khan — Pixar in a Box. Курс дает

и теорию, и практику анимации, то есть это отличная стартовая точка. Преподаватели курса работают в Pixar и являются создателями *Истории Игрушек* и многих других прославленных работ. Они расскажут, как использовать математику кривых Безье для расчета реалистичных моментов, и о многом другом. Определенно стоит взглянуть! **LXF**

# СТАТЬ КАК ХАКЕР

Освойте основы хакерства ради безопасности своей системы. Не волнуйтесь, это законно, уверяет Нейт «Доверься мне» Дрейк...



**П**росмотр телепрограмм вроде *Мистер Робот* создает впечатление, что хакерство — это область бледных от сидения взапери безумцев, которые мошеннически вламываются в дома людей во имя своих интересов, совершая зловредные акты цифрового саботажа.

В реальности хакерство в кругах ИТ считается достойной профессией. Этичных «белых» хакеров корпорации часто нанимают для взлома защищенных систем ради нахождения уязвимостей. Это называется тестом на вторжение.

Ваш первый инструмент, которым стоит обзавестись, чтобы приступить к делу — дистрибутив вторжения в Linux, например, Kali; его можно скачать с [www.kali.org](http://www.kali.org). Обычно дистрибутивы

тестирования на вторжение запускаются с CD или USB-брелка и содержат множество хакерских инструментов, которые можно использовать в целевой сети для определения уязвимых зон.

## Этичных хакеров нанимают для взлома защищенных систем ради нахождения уязвимостей.

Если вы пока еще новичок в хакерстве, вас может шокировать тот факт, что есть специальные дистрибутивы Linux для распространения хакерских инструментов. Разница в том, что вы будете использовать их для защиты, чтобы совершенно законными методами защитить свою систему.

Виртуальная машина Metasploitable (<http://bit.ly/LXFmetasploit>) специально создана с уязвимостями,

чтобы помочь тестированию и безопасно улучшить ваши хакерские навыки. Действия по настройке Metasploitable версии 2 с помощью Kali в *VirtualBox* вы найдете в следующем разделе.

У каждого хакера есть любимые инструменты, которыми он и пользуется, поэтому наше руководство стоит рассматривать как широкий обзор самых общих хакерских методов. Освоившись с хакерскими методами, вы перестанете быть «мелким скриптописцем» и начнете создавать собственные программы.

Большинство описанных здесь хакерских техник пассивны, то есть не влияют на данные в целевой системе. Но тем, кто решит обойтись без Metasploitable, я бы советовал до начала работы сделать резервные копии своих данных.

**П**режде чем делать свои первые шаги на пути превращения в хакера, скачайте файл ISO для Kali Linux с [www.kali.org](http://www.kali.org). Там предусмотрены инструменты, которыми вы будете пользоваться, чтобы овладеть основами хакерства. Kali очень гибкий и может загрузиться прямо с DVD или установиться на жесткий диск или устройство USB.

Вам также понадобится виртуальная машина Metasploitable2 (v.3 несовместима), которую можно скачать с <https://sourceforge.net/projects/metasploitable/files/Metasploitable2>. Файлы Metasploitable2 — в формате ZIP, поэтому прежде всего распакуйте их. В Ubuntu щелкните правой кнопкой по ZIP-файлу и выберите Extract To... [Извлечь в...].

Metasploitable2 предназначен для работы на виртуальной машине, поэтому его можно запустить в окне на вашем компьютере. Он также может содержать совершенно не такую операционную систему и файлы, как на вашем физическом компьютере.

В данном руководстве мы рассмотрим, как настроить отдельные виртуальные машины для Kali и Metasploitable2, а также как безопасно их подключить, чтобы вы могли отточить свои хакерские навыки, не затрагивая остальной сети.

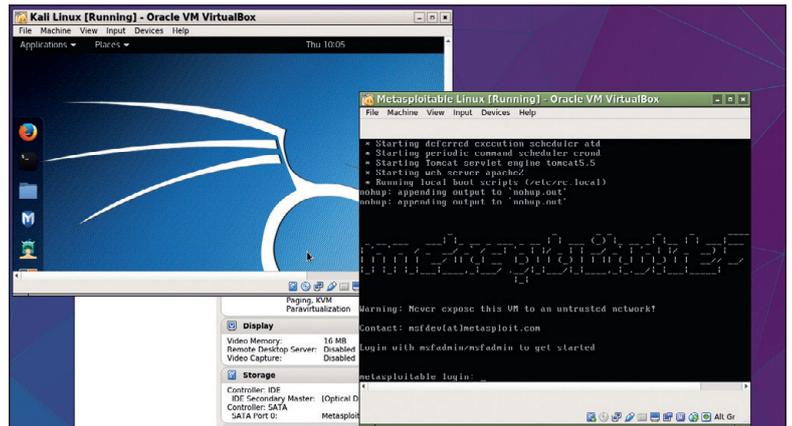
Виртуальные машины управляются специальными приложениями, например, программой VM от Oracle *VirtualBox*. Для начала установите *VirtualBox* с помощью репозитория пакетов:

```
sudo apt-get install virtualbox
```

После установки вы в любое время сможете запустить *VirtualBox* из Терминала, командой `virtualbox`.

Когда *VirtualBox* откроется, нажмите на кнопку New [Новая] вверху слева, чтобы создать виртуальную машину для Kali. *VirtualBox* предложит вам выбрать имя для машины. Введите Kali Linux и нажмите Next [Далее]. Затем *VirtualBox* спросит вас, сколько ОЗУ на вашем компьютере отвести этой машине. Выберите не менее 1024 МБ, но не перемещайте бегунок за пределы зеленого поля. Еще раз нажмите Next.

Следующий раздел чисто опциональный. *VirtualBox* умеет создавать файлы VDI (virtual disk image — образа виртуального диска), которые работают как виртуальный жесткий диск для машины. Kali спокойно загрузится с ISO-образа, но если вы хотите иметь возможность обновиться и сохранить изменения, выберите создание образа динамически отводимого виртуального диска и с помощью ползунка установите его размер не менее



20 ГБ. Сделав это, нажмите Create [Создать] для настройки Виртуальной машины.

Снова нажмите на кнопку New в *VirtualBox*, чтобы создать новую виртуальную машину, на сей раз с именем Metasploitable Linux. Нажмите Next, чтобы продолжить, и затем опять, чтобы принять отведенное по умолчанию количество ОЗУ. (Metasploitable2 работает только в текстовом режиме, и ему не нужно много ресурсов).

В разделе Hard Disk [Жесткий Диск] выберите Use an Existing Virtual Hard disk file [Использовать Существующий Файл Жесткого Диска], затем нажмите на значок папки, чтобы найти образ диска MetaSploitable2 (**Metasploitable.vmdk**). Нажмите Create для завершения процедуры настройки.

На этой стадии вы можете создать виртуальный адаптер хоста для ваших двух машин с помощью *VirtualBox*. Это позволит Kali подключиться к виртуальной машине Metasploitable2, чтобы использовать ваши хакерские инструменты, но сами машины не смогут получить доступ к остальной части сети или Интернету, а значит, всё это совершенно безвредно. Как это сделать — см. пошаговое руководство внизу.

Настроив виртуальный адаптер сети, выделите каждую виртуальную машину и нажмите на кнопку Start, чтобы включить их. Войдите в Metasploitable2 под именем пользователя msfadmin и с паролем msfadmin. Если вы не создали виртуальный диск для Kali, выберите режим Live при загрузке его виртуальной машины.

➤ Идеальный конечный продукт: Kali Linux и Metasploitable2, работающие на отдельных виртуальных машинах, способных соединяться только друг с другом.

»

## Пусть говорит Kali



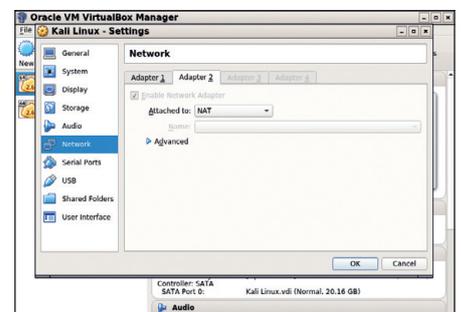
### 1 Создайте свой интерфейс виртуального адаптера хоста

В *VirtualBox* перейдите в File > Preferences [Файл > Предпочтения]. Нажмите на Network [Сеть] в левой панели, затем выберите вкладку Host Only Networks [Сети виртуального адаптера хоста]. Нажмите на зеленую кнопку +, чтобы добавить новый виртуальный адаптер хоста. По умолчанию имя вашей новой сети `vboxnet0`. Затем нажмите OK.



### 2 Подключите Kali к сети виртуального хоста

Выделите виртуальную машину Kali Linux и нажмите Settings > Network [Настройки > Сеть]. В меню Attached to [Прикреплено к] выберите Host-only adapter [виртуальный адаптер хоста]. Появится имя сети виртуального хоста. Нажмите OK, затем повторите все перечисленные шаги для виртуальной машины Metasploitable Linux.



### 3 Добавьте второй сетевой интерфейс

Если вы хотите обновить Kali или добавить новые программы, вам нужен доступ к интернет. Чтобы добавить еще один сетевой интерфейс, перейдите в Settings > Network для виртуальной машины Kali. Нажмите на вкладку Adapter 2 и затем на Enable Network Adapter [Включить Сетевой Адаптер]. В меню Attached To нажмите NAT > OK.

# Искусство боевой езды

**Б**оевая езда (wardriving) означает поиск беспроводных сетей при перемещении пешком или на автомобиле. Иногда так делают хакеры — в поисках уязвимых сетей.

Как этический хакер, вы будете это делать с целью убедиться, что у вас дома или в сети нет вредоносных беспроводных сетей, так как они могут вызывать помехи на законных точках доступа. Возможно, вы также захотите убедиться в том, что ваша беспроводная сеть хорошо скрыта и в безопасности.

Чтобы приступить к работе вардрайвера, вам нужна беспроводная карта или приставка, которые можно перевести в режим «мониторинга». Это используется для пассивного определения беспроводных сетей, без вмешательства в них. Не все сетевые карты поддерживают данную функцию. В Ubuntu Linux вы можете проверить, поддерживает ли ее ваша карта, с помощью команды `sudo ifconfig wlan0 down && sudo iwconfig mode monitor`, где wlan0 — имя вашего беспроводного интерфейса. Наше руководство написано с использованием TP-Link TL-WN823N USB Wi-Fi адаптера (J7 на Amazon UK).

Вам также понадобится устройство GPS — с ним проще точно засечь точки беспроводного доступа. Какую бы модель вы ни выбрали, она должна быть совместима с Linux. Наше тестирование обнаружил, что приемник GlobalSat BU-353-S4 USB GPS Receiver работает с Kali Linux по умолчанию.

Если вы используете Kali на виртуальной машине, обязательно следуйте инструкциям (см. стр. 31) по настройке второго сетевого интерфейса, поскольку вам понадобится доступ к Интернету, чтобы скачать файлы для работы устройства GPS.

Кроме того, *Virtualbox* по умолчанию не работает с устройствами USB. Если вы решили установить Kali на виртуальный диск, можете скачать Guest Additions в *Virtualbox*, что разрешит дополнительные функции, в том числе и поддержку устройств USB. Просто запустите виртуальную машину Kali и нажмите на меню Devices [Устройства], затем на Insert Guest Additions CD Image [Вставить Образ CD Гостевых Дополнений]. Выполнив инструкции, перезапустите виртуальную машину Kali и снова нажмите на меню Devices. Наведите мышью на USB. Отсюда вы можете выбрать подключение вашего Wi-Fi-адаптера USB и устройства GPS, отметив окно-флажок рядом с их названиями.

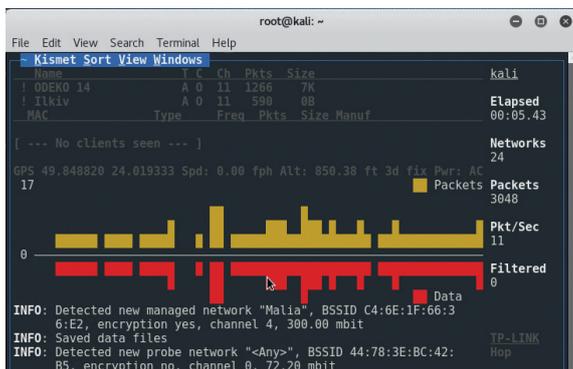
Когда Kali загрузится, следуйте пошаговым инструкциям внизу, чтобы заработало ваше устройство GPS. Далее откройте *Терминал* и запустите команду `kismet -l`. Она запустит *Kismet*, инструмент вардрайвера для мониторинга Wi-Fi сетей. Используйте клавиши со стрелками и нажимайте на Enter, когда делаете выбор.

У вас спросят, хотите ли вы видеть цвета. Для подтверждения нажмите на Enter — и затем снова, игнорируя предупреждение о пользователях root. *Kismet* сообщит, что не может определить источник пакетов. Нажмите на Enter, чтобы добавить имя вашего сетевого интерфейса. (Скорее всего это будет wlan0, но если вы не уверены, запустите команду `ifconfig`). Используйте Tab для выделения кнопки Add [Добавить], затем нажмите на Enter. Сделайте то же самое, чтобы закрылось окно консоли и открылось главное окно *Kismet*.

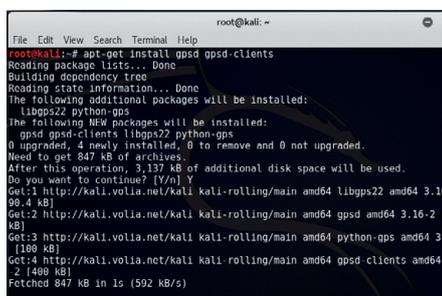
Запустившись, *Kismet* создаст в вашей домашней папке файл с расширением `.netxml`. Вы можете использовать его с картографическим ПО для определения местонахождения Wi-Fi сетей вокруг вас.

Удобный инструмент-компаньон *kismet*, *giskismet*, умеет конвертировать данные, которые вы собираете, в файл данных KML, распознаваемый большинством браузеров Земли, такими как *Google Earth*.

➤ *Kismet* определяет беспроводные сети вокруг вас. Интерфейс с устройствами GPS для определения местонахождения точек доступа.

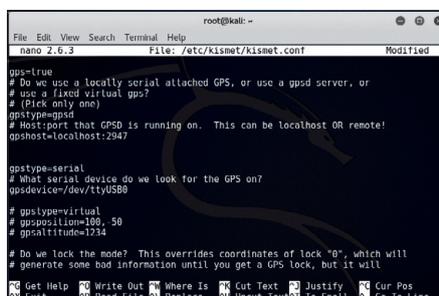


## Поработаем-ка в Kismet



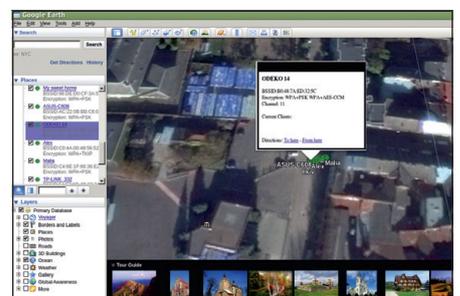
### 1 Настройка устройства GPS

Откройте Терминал в Kali и запустите `apt-get install gpsd gpsd-clients`, чтобы скачать нужные файлы. Затем подключите свое устройство GPS через USB. Если у вас Kali работает на *VirtualBox*, нажмите Devices > USB [Устройства > USB] и включите устройство GPS. Запустите команду `dmesg | tail -n 5`, чтобы посмотреть, где оно смонтировано, например, `/dev/ttyUSB0`.



### 2 Запустите сервис GPS

Запустите GPS Daemon: `gpsd /dev/ttyUSB0`. Далее отредактируйте настройку *Kismet* с помощью `nano /etc/kismet/kismet.conf`. Удалите `#` при `gpsdmode=serial` и `gpsddevice=/dev/rfcomm0`. Также замените `rfcomm0` на местоположение GPS. Нажмите Ctrl+X, затем Y, затем Enter. Теперь запустите *Kismet* с помощью `kismet -l`.



### 3 Экспорт данных GPS

Когда *Kismet* запустится, вы должны увидеть данные GPS. Нажмите Ctrl+C, чтобы остановить сервис, затем снова откройте *Терминал* в Kali. Используйте команду `ls` для поиска имени файла `.netxml`, затем запустите `giskismet -x <имяфайла>.netxml`. Далее запустите `giskismet -q 'select * from wireless' -o giskismet_wardrive.kml`.

# Взлом паролей Wi-Fi

Проверьте надежность ваших паролей Wi-Fi с помощью *Aircrack-ng* и *Reaver*.

Здесь мы сосредоточимся на применении инструментов Kali для проверки надежности ваших беспроводных сетевых ключей, через попытки напрямую взломать пароль. За последние годы эта технология прошла долгий путь, и при условии, что ваша сеть использует шифрование WPA2-AES и сильный пароль и в ней отключен WPS (см. врезку внизу), получить доступ к ней будет крайне сложно.

Один из методов взлома шифрованных паролей Wi-Fi — метод грубой силы, в котором перебираются все комбинации символов, пока не будет обнаружена нужная. Теоретически такое возможно, но на практике на это может уйти не один год, особенно при использовании длинного пароля.

Kali включает пакет инструментов *aircrack-ng*, который могут применять хакеры, чтобы значительно ускорить этот процесс. Обычно это делается перебором паролей из ранее составленного списка слов, содержащего часто используемые пароли и пароли по умолчанию, а также вынуждением устройств заново авторизоваться, чтобы поймать процедуру кэширования, где осуществляется обмен беспроводным ключом с роутером.

Для начала вам нужна беспроводная карта с возможностью инъекции пакетов. Ранее упомянутый ключ TP-Link USB имеет эту возможность.

Потом проверьте, может ли Kali определить адаптер. Откройте *Терминал* и запустите команду `airmon-ng`. Отключите все процессы, способные вмешаться в захват пакетов с помощью `airmon-ng check kill`, и переведите адаптер в режим мониторинга с помощью `airmon-ng start wlan0`.

Запомните имя интерфейса, а затем запустите `airodump-ng <имя интерфейса>` для получения списка сетей вокруг вас, например, `airodump-ng wlan0mon`.

Найдите в списке целевую сеть. Запишите себе "BSSID" и "CH" (Channel). Потом запустите следующую команду:

```
airodump-ng -w <logfile> -c <channel> -bssid <bssid> -ivs <interface>
```

Например, она может выглядеть так:

```
airodump-ng -w log.txt -c 11 --bssid B0:48:7A:ED:32:5C --ivs wlan0mon
```

```
root@kali: ~
File Edit View Search Terminal Help

Aircrack-ng 1.2 rc4

[00:01:43] 44152/9822768 keys tested (437.16 k/s)

Time left: 6 hours, 12 minutes, 56 seconds      0.45%

Current passphrase: alexial23

Master Key   : 65 C7 ED 6E F5 7E 36 57 78 3E 64 7C A4 E2 C3 27
              1D 08 8F 5C 63 2A 16 FF C1 75 2B 01 55 71 75 13

Transient Key: E6 2E F6 81 07 24 B2 AF 97 E8 F2 BB 49 31 8F 40
              F4 E4 E3 73 24 35 D6 09 47 21 44 6E AC 9A B3 42
              68 25 36 1C 40 4A B2 7A 4A 80 00 9C FC FA 07 F2
              BE 6B F9 6C 52 B9 2E E5 17 A0 14 48 BA 31 DF CF

EAPOL HMAC  : 69 67 98 1C 1A F7 B7 DB 4D D4 AD E7 36 DC 8A 56
```

► Собрал данные, попытайтесь взломать пароль с помощью *aircrack-ng*. Это может занять много времени!

Теперь *airodump* начнет перехватывать пакеты данных и сохранять их в ваш лог-файл. Чем больше устройств пользуется беспроводной сетью, тем больше данных перехватывается.

Можно попробовать отключить от сети Wi-Fi все устройства, чтобы им пришлось подключиться заново. Это позволит вам перехватить крайне важное кэширование, содержащее пароль. Откройте новый *Терминал* и запустите

```
aireplay-ng -0 5 -a <bssid> <interface>
```

И, наконец, для завершения процесса по Ctrl+C и запустите ls для отображения точного имени вашего лог-файла.

Для взлома пароля нужен список слов. Вы можете скачать его или извлечь файл [rockyou.txt.gz](http://rockyou.txt.gz), расположенный в `/usr/share/wordlists` в Kali.

Начните взлом пароля с команды

```
aircrack-ng <filename> -w /usr/share/wordlists/rockyou.txt
```



## Поверьте в Reaver

Многие роутеры поддерживают WPS (Wi-Fi protected setup — защищенную конфигурацию Wi-Fi), что в теории облегчает подключение устройств к вашей сети простым нажатием кнопки на вашем роутере.

Обычно подключение WPS обеспечивается только восьмизначным контактом, взломать который намного проще и быстрее, чем полнофункциональный сетевой пароль.

По этой причине в идеале вы должны попытаться отключить WPS на всех роутерах, которыми вы управляете.

Чтобы проверить сеть на предмет поддержки подключений через WPS, запустите команду `wash -l <interface>`.

Однако если владелец сети настаивает на использовании WPS, постарайтесь

повлиять на его мнение, попросив разрешения использовать утилиту *Reaver*.

Сначала откройте *Терминал* в Kali и переведите свой роутер в режим мониторинга, как описано выше, с помощью `airmon-ng start wlan0`.

Далее запустите команду `reaver -i <interface> -b <bssid>`, например, `reaver -i wlan0mon -b B0:48:7A:ED:32:5C`.

Начавшаяся сессия взлома будет обеспечатывать вас регулярными обновлениями по состоянию процесса в процентах. На взлом PIN WPS должно уйти около десяти часов, в зависимости от скорости вашего процессора. Вы можете использовать Ctrl+C, чтобы приостановить *Reaver* и затем возобновить работу с того же места позднее.

```
root@kali: ~
File Edit View Search Terminal Help

root@kali:~# reaver -i wlan0mon -b B0:48:7A:ED:32:5C

Reaver v1.5.2 WiFi Protected Setup Attack Tool
Copyright (c) 2011, Tactical Network Solutions, Craig Heffner <cheffner@tacnetsolutions.com>
mod by t6_x <t6_x@hotmail.com> & DataHead & Soxrok2212

[+] Waiting for beacon from B0:48:7A:ED:32:5C
[+] Associated with B0:48:7A:ED:32:5C (ESSID: ODEKO 14)
[+] Starting Cracking Session, Pin count: 0, Max pin attempts: 11000
[P] E-Nonce: 8a:8a:b9:59:8b:79:1f:52:01:ba:89:e9:ff:50:76:c0
[P] PKC: fb:7d:a3:0d:fc:81:7e:9e:76:38:9b:61:12:53:69:53:44:08:0a:20:ce:6a:4a:2b:6a:56:69:27:d1:ac:06:a3:ce:73:5c:96:73:fe:09:bc:af:aa:5e:9e:2d:71:1d:08:77:92:40:2a:95:b1:e3:2d:4b:e0:00:26:5a:30:a8:30:b7:c9:f0:aa:ab:89:16:4e:f4:96:9a:63:e5:c4:42:7a:17:db:2a:63:d5:d1:98:c0:8e:5c:32:67:2a:c9:ef:3e:a9:09:42:7b:c5:44:e0:a8:96:51:ee:ba:e2:f9:42:29:58:66:3c:47:4a:a7:6a:cf:cf:5d:62:74:11:c0:31:49:48:e9:e0:1:02:37:48:ed:4b:a7:za:04:d2:1b:3a:57:3f:01:0f:46:f8:57:f3:e0:e2:08:b5:37:01:44:02:42:19:e5:36:b2:5a:7e:d7:4f:0a:20:6e:6c:96:70:92:0a:ba:09:a5:d9:70:09:aa:5a:62:3f:15:58:31:43:d4:4a:2c
[P] WPS Manufacturer: TP-LINK
[P] WPS Model Name: TL-WR740N
[P] WPS Model Number: 1.0/2.0
[P] Access Point Serial Number: 1.0
[P] R-Nonce: 52:ce:f4:06:1c:16:ad:fe:a9:ce:d4:75:00:c1:9c:f7
```

► *Reaver* будет запускаться в Kali автоматически и сохранять свое состояние, чтобы вы могли возобновить работу с того же места.

# Подготовка к атакам оффлайн

Берите Резиновую Уточку и проникайте в целевую сеть...

**Н**адеясь, что к данному моменту вы уже обнаружили: ваша целевая сеть защищена от всех попыток взломать пароль. Если это так, то надо рассмотреть следующие шаги, которые может предпринять злонамеренный хакер, то есть получить физический доступ к вашей машине и взломать ее именно таким образом.

Простой формой подобной атаки может быть просто нанесение визита в ваш офис с DVD с Kali и загрузка Kali с вашего компьютера Windows, чтобы исследовать незашифрованные файлы на вашем жестком диске.

Более сложные атаки возможны с помощью USB Rubber Ducky [Rubber Ducky — *англ.* Резиновая Уточка]. Этот инструмент, созданный Hak5, напоминает USB-брелок, однако на самом деле является весьма сложной платформой ввода комбинаций клавиш.



➤ Rubber Ducky выглядит, как безобидный USB-брелок, но обладает скрытыми глубинами...

«Уточка» использует карту microSD и простой язык скриптов для ввода информации: автоматизированных атак, которые, например, дают вам доступ root к целевой машине. Rubber Ducky обманывает компьютер, заставляя его думать, что это клавиатура, и принимать команды. Хотя вы вполне можете создать атаки сами, у Hak5 имеется wiki предварительно подготовленных атак для Windows, Linux и MacOS, которые можно скопировать на Rubber Ducky.

Если вам кажется, что тайком проникнуть в собственный дом, чтобы установить устройство на уязвимый компьютер — не самая интересная задача, можете дать Rubber Ducky (плюс фомку) не слишком разбирающемуся в ИТ другу, который может попытаться взломать ваш компьютер.

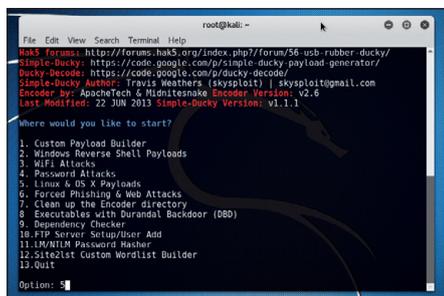
Для начала разоритесь на USB Rubber Ducky от Hak5 (<https://hakshop.com/products/usb-rubber-ducky-deluxe>). На данный момент розничная цена составляет около \$45.

В ожидании прибытия заказанного устройства, запустите Kali и откройте Терминал. Введите команду `git clone --recursive git://github.com/skysploit/simple-ducky.git /usr/share/simple-ducky`, чтобы скачать программу Simple Ducky, которая автоматически генерирует информационную нагрузку.

Далее запустите `bash /usr/share/simple-ducky/install.sh` для выполнения программы установки, затем `simple-ducky` для запуска Simple Ducky. Прежде чем приступить к загрузке информации, выберите Option 9, чтобы скачать все зависимости. Следуйте шагам инструкции (см. врезку внизу), чтобы скопировать информацию на USB Rubber Ducky. Если никакая информация для загрузки в Simple Ducky не отвечает вашим потребностям, можете найти больше на [www.usbrubberducky.com](http://www.usbrubberducky.com).

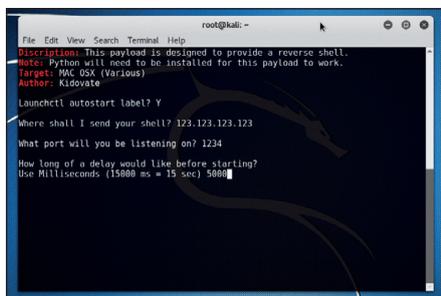
После окончания Rubber Ducky готова проникнуть на целевое устройство. Усложните жизнь коллегам — тестировщикам на вторжение, используя Linux с Full Disk Encryption. Выключайте компьютеры полностью, когда уходите, и запирайте дверь.

## Поиграйте с Уточкой



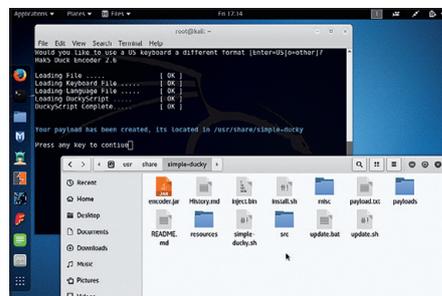
### 1 Выберите информационную нагрузку

Откройте Терминал в Kali и введите `simple-ducky`. Программа спросит вас, с чего начать. Надеемся, вы представляете, на какую систему нацелились. Выберите опции с номерами — так, атака на Linux и MacOS находятся в Опции 5. Затем вы увидите список имеющей информационной нагрузки для целевой системы.



### 2 Настройте информационную нагрузку

В зависимости от выбранного типа данных вам предложат ответить на ряд простых вопросов. В примере выбранная нагрузка «OS X Reverse Shell» предназначена для взлома Mac через Интернет. Данные запрашивают сервер и номер порта для передачи и позволяют настроить отсрочку перед началом процесса.



### 3 Передача данных

После выбора опции настройки ваши данные будут созданы в папке `/usr/share/simple-ducky`. Вам необходимо будет передать файл `inject.bin` на карту microSD, которую впоследствии вы установите в Rubber Ducky. Загляните в файл `payload.txt`, чтобы посмотреть разбор атаки, описанный на собственном простом языке скриптов Ducky.

# Приступите к анализу пакетов

Перехватывайте и анализируйте пакеты данных в целевой сети.

**Т**еперь, когда вы, надо надеяться, подключились к целевой сети, можете предаться тонкому искусству анализа пакетов. Этот термин описывает процесс перехвата и анализа пакетов данных в сети.

Главная проблема при анализе пакетов — огромное количество данных, перемещающихся в сети с использованием разных протоколов. Бывает сложно определить, например, посещал ли пользователь определенный сайт или скачал ли он определенный тип файлов. К счастью, ваша задача сильно упрощается благодаря использованию предустановленного анализатора пакетов *Wireshark* (см. также **LXF218**, **LXF191**). *Wireshark* умеет перехватывать данные в сети и позволяет вам их фильтровать. Самое важное — вы можете перехватывать данные в течение определенного времени и сохранять их в файл, что позволяет выполнить более глубокий анализ.

*Wireshark* использует систему подцветки кода для визуального представления разных типов трафика. Например, TCP-трафик показан зеленым, DNS — темно-синим, а светло-голубой представляет UDP. Вы можете запустить *Wireshark* из Kali, перейдя в Applications > Wireless attacks [Приложения > Беспроводные Атаки]. У вас спросят, хотите ли вы отслеживать определенный сетевой интерфейс, например, wlan0, или весь сетевой трафик.

Помните, что ваш экземпляр Kali, чтобы он мог отслеживать трафик, должен быть подключен к требуемой сети. Для наилучших результатов используйте беспроводную карту, которая поддерживает режим приема всех сетевых пакетов [promiscuous mode]. Ключ TP-Link USB, упомянутый в разделе *Искусство божьей езды* (см. стр. 32), для этого вполне подходит.

При первом запуске *Wireshark* начнет автоматически перехватывать «живые» данные в сети. Используйте устройство, подключенное к сети, для посещения нескольких сайтов, например, [www.linuxformat.com](http://www.linuxformat.com), чтобы потом попрактиковаться в навыке анализа пакетов. Вы можете в любое время нажать на кнопку Stop Capture и выбрать File > Save [Файл > Сохранить], чтобы позднее сохранить собранные данные. Просто выберите File > Open [Файл > Открыть] на любых перехваченных файлах.

Структура *Wireshark* довольно проста. Блок “No” — номер для каждого пакета данных. Второй столбец показывает, сколько секунд прошло с момента начала перехвата данных. Третий и четвертый столбцы показывают исходный и конечный IP-адреса соответственно. Пятый столбец отображает протокол, который

No.	Time	Source	Destination	Protocol	Length	Info
111	12.160000343	192.168.1.102	80.244.178.150	HTTP	550	GET / HTTP/1.1
<p>▶ Frame 111: 550 bytes on wire (4400 bits), 550 bytes captured (4400 bits) on interface 0</p> <p>▶ Ethernet II, Src: Apple_b3:48:0b (10:9a:dd:b3:48:0b), Dst: Tp-LinkT_ed:32:5c (b0:48:7a:)</p> <p>▶ Internet Protocol Version 4, Src: 192.168.1.102, Dst: 80.244.178.150</p> <p>▶ Transmission Control Protocol, Src Port: 37696, Dst Port: 80, Seq: 1, Ack: 1, Len: 484</p> <p>▶ Hypertext Transfer Protocol</p>						
0000	b0 48 7a ed 32 5c 10 9a dd b3 48 0b 08 00 45 00					.Hz.2\..H...E.
0010	02 18 0d d1 40 00 40 06 65 76 c0 a8 01 66 50 f4					...@. ev...FP.
0020	b2 96 93 40 00 50 19 76 31 ed e3 dc 42 f3 80 18					...@.P.v 1...B...
0030	00 e5 26 17 00 00 01 01 08 0a 02 81 3c 3b 74 6b					...&...<...;tk
0040	72 9a 47 45 54 20 2f 20 48 54 54 50 2f 31 2e 31					r.GET / HTTP/1.1
0050	0d 0a 48 6f 73 74 3a 20 6c 69 6e 75 78 66 6f 72					.Host: linuxfor
0060	6d 61 74 2e 63 6f 6d 0a 55 73 65 72 2d 41 67					mat.com, User-Ag
0070	65 6e 74 3a 20 4d 6f 7a 69 6c 6c 61 2f 35 2e 30					ent: Moz illa/5.0
0080	26 28 58 31 31 3b 20 55 62 75 6e 74 75 3b 20 4c					(X11: Ubuntu: 1

отправил пакеты, например, TCP или UDP. Строка Display Filter [Фильтр Отображения] сверху используется для отделения определенного типа пакетов данных.

Полный список параметров, которые вы можете здесь выбрать, имеется в справочнике *Wireshark* ([www.wireshark.org/docsdref](http://www.wireshark.org/docsdref)). Попробуйте простой фильтр, чтобы проверить, получал ли кто-нибудь доступ к сайту Linux Format, введя `http.host contains "linuxformat.com"`. Нажмите Return, чтобы увидеть соответствующий трафик.

Если перехваченные данные содержат незашифрованные протоколы, например, FTP, то вы даже сможете получить доступ к паролям. Например, введите `ftp` в фильтр отображения, чтобы показывать только незашифрованные соединения через FTP, затем щелкните правой кнопкой по списку перехваченных пакетов. Выберите Follow > TCP Stream [Следовать > Поток TCP], чтобы увидеть имя пользователя и пароль. При зашифрованной аутентификации нужен частный ключ, чтобы декодировать данные для их прочтения. Даже если вы не сумеете прочитать определенные типы данных, вы можете легко их найти. Простейший вариант — ввести `ssl` в фильтр отображения, чтобы увидеть зашифрованный трафик.

*Wireshark* также годится для нахождения определенного вида трафика на разных портах. Например, протокол Tor, специально разработанный для того, чтобы скрывать активность в сети, работает через TCP на порте 9001. Введите `tcp.dsport == 9001`.

Перехват и анализ данных — приобретаемый навык, и на изучение всех опций фильтров *Wireshark* нужно время. Постепенно структура и параметры начнут казаться простыми и логичными. »

» Используйте строку Фильтр Отображения *Wireshark* для сортировки перехваченных данных — например, чтобы узнать, посещал ли клиент некий сайт.

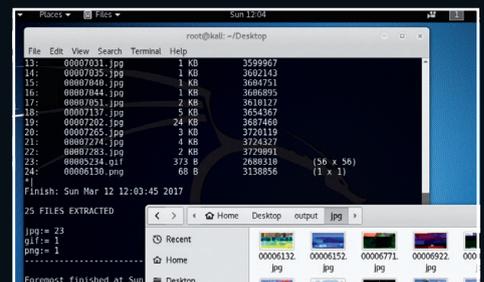
## Первоочередные файлы

Если вы сохранили файл перехваченной информации из *Wireshark* и вам нужен простой способ показать все двоичные файлы, например, изображения JPG, то вы можете использовать инструмент командной строки *Foremost*.

Простейший способ это сделать — сохранить файл в *Wireshark*, выбрав File... > Save As [Файл... > Сохранить Как], затем открыть Терминал Kali и запустить команду `foremost -i -v <имяфайла>`. Она создаст папку с именем **output**, где содержатся все файлы, собранные в сети.

Однако это не самый точный способ решения задачи, поэтому подумайте о том, чтобы

открыть этот файл в *Wireshark* и применить фильтр отображения `http.content_type contains [jpg]`. Замените `[jpg]` на требуемое расширение файла. Если вы знаете имя нужного файла, то это будет намного проще. Щелкните правой кнопкой по списку собранных пакетов данных и выберите Follow > TCP Stream. Откроется новое окно. Нажмите Entire Conversation [Вся Беседа] для выбора полученных данных (обычно это будет больший из двух). В Show и Save Data As выберите Raw и затем нажмите на Save As, чтобы сохранить файл. Запустите *foremost* на этом новом файле как раньше, чтобы извлечь определенные файлы.



» Запустите *foremost* на файле собранных пакетов *Wireshark* для извлечения всех двоичных файлов.

# Скан по всем портам

Сетевые порты — это точки входа между компьютером и интернет. Даже системы, использующие брандмауэры, оставляют некоторые порты открытыми, чтобы могли работать сервисы. Например, компьютер, предназначенный для работы в качестве сервера для *Minecraft*, должен оставить открытым порт 25565.

Одно из первых действий хакеров при подключении к сети — сканирование на предмет открытых портов, чтобы попытаться взломать пароль для этого сервиса или запустить вредоносный код. Будучи начинающим тестировщиком на вторжение, вы тоже должны будете просканировать порты своих устройств, чтобы убедиться в отсутствии уязвимостей. Вам пригодится *Sparta*, графический интерфейс для нескольких инструментов сканирования. Основной среди них — *Nmap*, который отправляет пакеты данных на порты на целевом компьютере. Если порт определяется как открытый, вы также можете попытаться взломать пароль с помощью *Hydra*, которое тоже запускается из *Sparta*. Вашим целевым устройством будет виртуальная машина *Metasploitable2*, которая специально снабжена уязвимыми портами, чтобы вы могли отточить свои хакерские навыки.

Сначала в *VirtualBox* запустите Kali и виртуальную машину *MetaSploitable2*. Войдите на *MetaSploitable2* с именем пользователя

*msfadmin* и паролем *msfadmin*. Далее запустите команду *ifconfig* для получения IP-адреса виртуальной машины *Metasploitable2* и запишите его.

Вернитесь к виртуальной машине Kali и откройте *Sparta* из *Applications > Vulnerability Analysis* [Приложения > Анализ Уязвимостей]. Когда откроется окно *Sparta*, нажмите *Add Hosts to Scope* [Добавить Хосты], введите IP-адрес виртуальной машины *Metasploitable2*, затем нажмите *Add to Scope* [Добавить]. Теперь *Sparta* начнет сканировать порты виртуальной машины *Metasploitable2* на предмет уязвимостей.

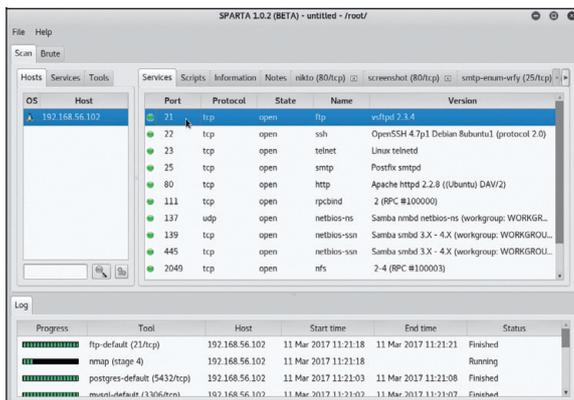
Вкладка *Services* главного окна покажет все открытые порты, а также сервисы, работающие на них — например, SSH работает на порте 22. Нажмите на вкладку *Information*, чтобы увидеть подробную информацию об операционной системе на целевом компьютере. Это очень удобно, если вы хотите запустить эксплойты, о чем мы расскажем в следующем разделе.

Уделите время на ознакомление с остальными вкладками, где дана информация о других программах, запускаемых *Sparta*. Например, *ftp-default* — инструмент, который пытается автоматически подключиться к целевому компьютеру и перебирает некоторые имена пользователя и пароли по умолчанию. Здесь сработали пароли по умолчанию, и вы теперь имеете доступ к машине через порт 21 через намеренно простой пароль "password".

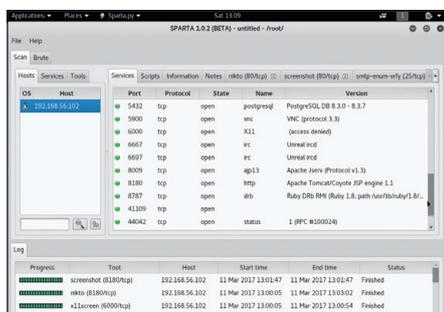
Вкладка *Nikto* показывает результаты для этого инструмента, который предназначен для сканирования web-серверов на предмет общих уязвимостей. Попозже мы расскажем о нем более подробно, а сейчас прочитайте каталог проблем безопасности. У *Sparta* также имеется удобная вкладка *Notes*, на случай, если вы решите записать какую-то информацию.

Поскольку в *Metasploitable2* так много слабостей, получить к ним доступ не составит проблемы, но если вы тестируете *Sparta* на одной из своих машин, вы, возможно, не сможете взломать их с той же легкостью. Вернитесь на вкладку *Services*, чтобы показать открытые порты и щелкните правой кнопкой по сервису под названием *Send to Brute* [Отправить на взлом]. Он позволит вам начать автоматический процесс взлома пароля для этих сервисов на основе списка паролей.

➤ *Sparta* использует *Nmap* для сканирования открытых портов, позволяя вам протестировать их на предмет уязвимостей.

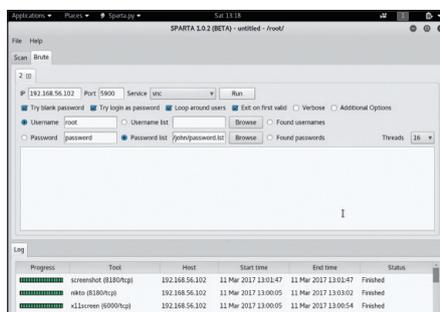


## Выпускайте Hydra!



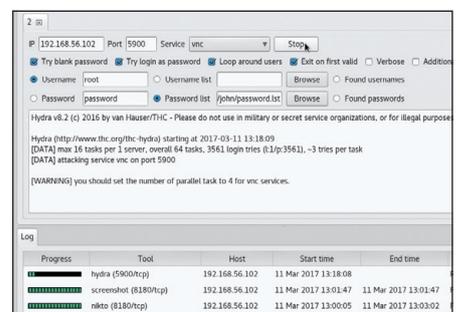
### 1 Выберите свой протокол

Обязательно сначала загляните на другие вкладки в *Sparta*, чтобы посмотреть, не были ли обнаружены пароли по умолчанию, так как это избавит вас от необходимости переходить на вкладку *Services* в *Sparta*, чтобы выбрать сервис для атаки. *Hydra* поддерживает ряд сервисов. Щелкните правой кнопкой по выбранному сервису и выберите *Send to Brute*.



### 2 Настройте свою атаку

Щелкните по вкладке *Brute* в *Sparta*. Опции по умолчанию, такие как проверка пустого пароля или пароля, совпадающего с именем пользователя (как в случае с *Metasploitable2*), уже выделены. Проверьте *Password List* [Список Паролей] и нажмите *Browse* для выделения одного из предустановленных списков Kali, например, файла *rockyou.txt* или */usr/share/john/password.lst*.



### 3 Запустите атаку

Нажмите на *Run*, чтобы запустить атаку на выбранный сервис. Нижняя панель будет показывать все предупреждения. Количество параллельных попыток взлома паролей *Hydra* контролируется через меню *Threads*. Если у целевой системы ограничено число поддерживаемых подключений для определенного сервиса, вы увидите предупреждение.

# Эксплойты

**Э**ксплойты — это крошечные, но высокоспециализированные программы, использующие определенные уязвимости в целевой системе. Они доставляют сведения хакерам, чтобы помочь получить доступ к удаленному компьютеру.

MetaSploit Project — создатели самой крупной в мире открытой базы эксплоитов; их можно получить через мощную среду Metasploit Framework, свободная версия которой предустановлена в Kali.

Если это название кажется вам знакомым, причина в том, что Metasploit Project также поддерживает виртуальную машину Metasploitable, которая позволяет безопасно тестировать эксплойты и загружать данные.

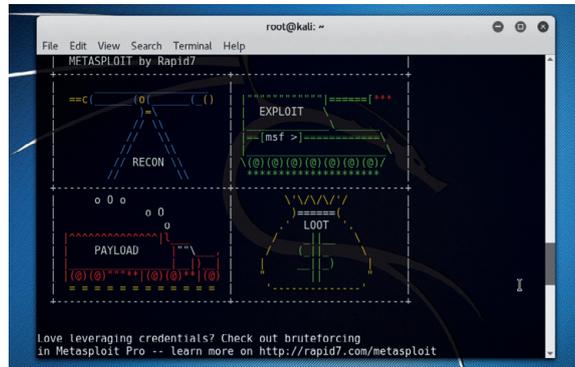
Для начала обязательно проследуйте всем инструкциям в предыдущих разделах для установки контакта виртуальных машин Kali и Metasploitable2 через сеть Host Only. Если вы настроили в *VirtualBox* второй сетевой интерфейс, чтобы Kali имела доступ к Интернету, отключите его. Это обеспечит безопасность от эксплоитов реальных машин в вашей сети.

Пошаговое руководство внизу описывает базовые действия по загрузке и запуску эксплоита для получения контроля за целевым компьютером. Самое сложное — узнать точно, какой эксплойт выбрать из тысяч имеющихся.

Первый шаг — просканировать целевой компьютер на предмет уязвимых портов для атаки. Это можно сделать, повторяя действия из предыдущего раздела по сканированию портов или просто запустив инструмент *Nmap* из командной строки.

Определив уязвимые порты, можете определить правильный эксплойт командой `searchsploit`. Инструмент *searchsploit* является удобной оффлайн-версией Exploit Database ([www.exploit-db.com](http://www.exploit-db.com)). А информацию по самым последним эксплоитам для разных платформ вы найдете в базе данных уязвимостей CVE на [www.cvedetails.com](http://www.cvedetails.com).

Если вы готовы использовать текстовый инструмент терминала Kali *searchsploit*, то базовый синтаксис команды — `searchsploit <условие поиска>`. Например, после использования команды *Nmap* для обнаружения того, что порт 6667 используется Unreal IRC Daemon, вы можете запустить `searchsploit irc`, чтобы увидеть все имеющиеся



► Среда MetaSploit имеет базу данных тысяч эксплоитов, которые вы можете запустить на целевой машине.

эксплойты для IRC, или `searchsploit unreal`, чтобы показать только эксплойты, содержащие это условие поиска.

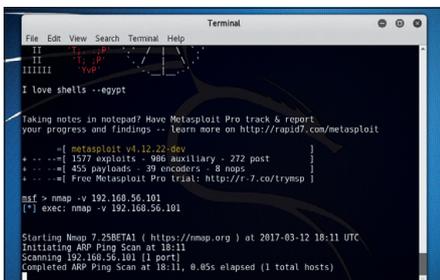
Помните, что не все эксплойты являются результатом чьего-то намеренного саботажа законной программы. Например, DistCC Daemon, который предназначен для облегчения крупномасштабной работы компилятора в системах Unix, может использоваться хакерами для запуска желаемых ими команд, если только заранее не принять мер предосторожности.

Более того, сам факт, что эксплойты попали в открытую базу данных, означает, что скорее всего имеются заплатки для решения соответствующих проблем. Рекомендуется постоянно обновлять систему, чтобы они обязательно были установлены.

Если это базовое введение в хакерство только раздражило ваш аппетит, Offensive Security предлагают Penetration Testing with Kali (PWK). Это 30-дневный курс для желающих освоить хакерские технологии. Команда использует лабораторию тестирования виртуального вторжения, чтобы помочь вам эмулировать разнообразные передовые сценарии тестирования на вторжение. Курс сертифицирован. Если вы успешно сдаете 24-часовой реальный тест на вторжение, вы получаете официальное свидетельство OSCP (offensive security certified professional) — сертифицированного специалиста в области информационной безопасности.

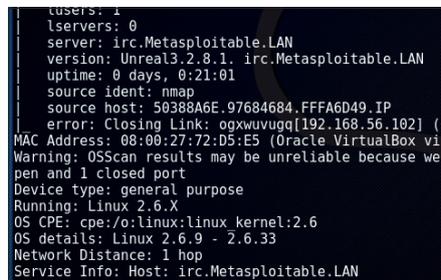
Удачного хакерства! **LXF**

## Эксплуататорское поведение



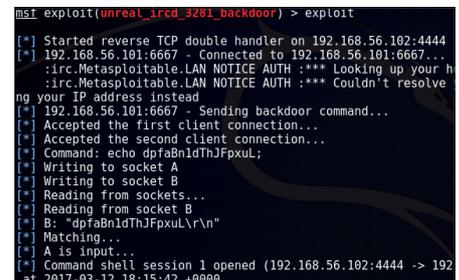
### 1 Запустите Msfconsole

Щелкните по значку консоли Metasploit Framework на рабочем столе Kali для запуска 'msfconsole'. Для работы ей требуется примерно 1,5 Гб ОЗУ — убедитесь, что это пространство отведено под виртуальную машину Kali. Затем повторите шаги из раздела «Скан по всем портам» или запустите команду `nmmap -v <IP-адрес целевой машины>` для определения уязвимых портов.



### 2 Подготовьте свой эксплойт

Выберите сервис, который хотите взломать. Мы попытаемся для получения доступа к целевой системе использовать злонамеренную лазейку [backdoor] в Unreal IRCD. Команда `nmmap -A -p 6667 <IP-адрес целевой машины>` даст вам информацию о демоне, работающем на этом порте. На этой стадии вы можете применить `searchsploit` (см. выше), чтобы выяснить, есть ли эксплойт именно для этой программы.



### 3 Доставьте свои данные для загрузки

Выбрав эксплойт, введите `use` для его загрузки, например: `use exploit/unix/irc/unreal_ircd_3281_backdoor`. Затем настройте, куда будут доставлены данные: `set RHOST <IP-адрес целевой машины>`. И, наконец, попытайтесь прорваться в систему, запустив эксплойт. Если эксплойт не сработает, возможно, он не предназначен для версии программы, работающей на этом порте.



# Наш Exercist

Джонни Бидвелл встретился с **Катриной Оуэн**, создательницей сайта exercism.io, чтобы обсудить Linux, языки и вечные проблемы начинающих программистов...



Так получилось, что, изучая молекулярную биологию, Катрина Оуэн [Katrina Owen] внезапно стала программистом. С тех пор она помогла в этом многим другим, через свой сайт [exercism.io](http://exercism.io). Она также поддерживает развитие Open Source на GitHub и, совместно с Сэнди Метц [Sandi Metz], написала *99 bottles of OOP* — книгу, посвященную тому, как написать эффективный и красивый объектно-ориентированный код. Мы пересеклись с ней на OSCON Europe в октябре 2016 г., чтобы обсудить травматичные воспоминания о Gentoo, дружелюбных роботов для проверки кода и то, чего люди ждут от учебника по кодированию...

**Linux Format:** Расскажите мне вашу историю. В вашей биографии сказано, что вы занимались молекулярной биологией.

**Катрина Оуэн:** Ну, вообще-то, лет до 25 я была секретарем. И примерно тогда же поняла, что в этой работе нет будущего: ни возможностей к развитию, ни должного уважения. Поэтому я решила, что мне надо получить диплом, но понятия не имела, в какой области. Хотелось, чтобы это было что-то осязаемое, что-то научное.

Сперва мне не хватало знаний по математике, физике и химии, чтобы поступить в университет. Поэтому месяцев 18 у меня ушло на самообразование, потом сдала кое-какие экзамены, чтобы получить допуск. Но я всё еще не знала, чем хочу заниматься. Единственным критерием для меня было то, что это должно было привести к реальной работе.

Меня приняли на две специальности: авиа- и ракетостроение и молекулярная биология, я не знала, что выбрать, но потом подумала: «Ну, во всей этой математике я совсем новичок, так что, пожалуй, лучше выбрать биологию». Я подумала, что не потяну тот уровень математики, который требуется на ракетостроении. Теперь, уже задним умом, я понимаю, что оно-то мне бы подошло лучше, потому что это практическая специальность и всё это строится на основных принципах. А с биологией всё выглядит так: «Это произошло в результате эволюции», и с теорией всё гораздо более размыто, в духе того, что «это так, и это надо просто запомнить». Это изрядно напрягает.

**LXF:** Как же вы пришли к компьютерам и программированию?

**КО:** Биологией я занималась три года, за это время я обзавелась своим компьютером на Linux и научилась писать скрипты для *Bash*. Меня увлекло программирование и решение задач, и я начала заниматься фрилансом, в основном просто помогая друзьям в отладке их сайтов и тому подобном. Они были восхищены моей эффективностью, и через них мне перепало много работы, а потом я нашла постоянное место.

**LXF:** Это же, наверное, были еще безмятежные дни юного JavaScript, когда у каждого браузера были свои представления о том, что такое язык?

**КО:** Нет, это было еще в дни DHTML. Но мне с ним, к счастью, дела иметь не пришлось. Я очень быстро стала заниматься системными разработками типа PHP, вообще не связанными с интерфейсом. Несколько лет спустя я перешла на Ruby, и это была совсем другая среда. Я чувствовала более тесное взаимодействие: гораздо больше встреч и конференций. Сообщество Ruby очень своеобразное. Они очень любят каламбурить, все названия проектов у них очень креативны.

**LXF:** Я спрашиваю потому, что сам выбрал Python почти произвольно, придя к необходимости выучить какой-то из «основных» языков. Почему ваш выбор пал на Ruby, а не на Python или что-нибудь еще?

**КО:** Знаете что? Рассматривала-то я оба варианта. И там, и там отличные сообщества, в обеих философия строится вокруг качества и вокруг стремления всё тестировать и делать так, чтобы это радовало программиста. Проработав достаточно с PHP, я стала думать, куда пойти дальше, где я буду чувствовать себя счастливее, но как такового выбора у меня даже не стояло. Я почти ничего не знала ни об одном из них, просто случилось так, что у меня появилась целая компания программистов Ruby, и так уж вышло, что со временем я стала участвовать в их проектах.

**LXF:** Вы упомянули, что установили Linux себе на ноутбук. А когда вы впервые с ним столкнулись?

**КО:** По-моему, начинала я с Debian, а через несколько лет перешла на Gentoo.

## О ЦЕЛЯХ САЙТА

# Можно научиться беглости, имея только базовые знания.

**LXF:** Я когда-то баловался с Gentoo. У всех бывает такой мазохистский период. Хотя тогда, в районе 2004 г., документация у него была гораздо лучше, чем у других дистрибутивов.

**КО:** Помню, как я мучилась с настройкой Wi-Fi. На самом деле, сейчас я вспоминаю, что первые несколько лет я не особо страдала. Потом я переехала в США [из Норвегии], где в гостинице не было Ethernet-сети, а мне очень нужен был онлайн-доступ. У меня, наверное, целый день ушел на то, как это сделать. И до сих пор не помню, как мне это удалось. Но помню, как я сама поразились, что мне это удалось. А потом при каждом обновлении системы приходилось отчаянно гуглить сообщения об ошибках, чтобы узнать, как же звучит заветное заклинание — было очень весело.

**LXF:** И все эти приключения привели вас к созданию [exercism.io](http://exercism.io). Не могли бы вы рассказать нашим читателям, что это?

**КО:** Сама идея Exercism выросла из проекта Language Hunters [Охотники за языками], который по-



священ вымирающим человеческим языкам. Это своего рода межличностный, игровой, интерактивный способ сохранить исчезающие языки. Он провёл для меня границу между беглостью в языке и компетентным знанием — раньше я думала, что эти вещи тесно взаимосвязаны. Но оказалось, что можно бегло говорить, не имея глубоких знаний, и наоборот. Так что, имея не очень высокий уровень, вы всё равно можете эффективно общаться. В этом и заключается цель Exercism: достичь минимального уровня беглости в языке программирования, научившись при этом решать на нем мелкие тривиальные задачи.

И делать это совершенно естественно, не задумываясь о том, как и что сформулировать, или где поставить скобки, или какой должен быть синтаксис. Если вы освоите азы синтаксиса, у вас останется больше умственных ресурсов на решение более сложных задач или решение каких-то более насущных вопросов. Множество людей использует Exercism для каких-то своих целей. Основная идея в том, чтобы научиться беглости, имея только базовые знания — в этом вся прелесть Exercism.

**LXF:** А какие языки люди учат на Exercism? Каков принцип работы сайта?

**КО:** Самые популярные, пожалуй, Ruby, Python, JavaScript, Haskell и Rust. И в последнее время — Elixir и Elm. Есть несколько нишевых языков, у которых находятся свои рьяные энтузиасты. У нас есть клиент командной строки, написанный на Go. Это хорошо, поскольку мы можем распространять свой клиент без дополнительных сред выполнения и зависимостей. Лишь бы он был у вас на каком-то пути, тогда и будет работать. Клиент генерирует упражнения, а когда ученик их выполняет с помощью любых инструментов, клиент отправляет результаты обратно нам. Само обсуждение проблем происходит на сайте, независимо от клиента.

**LXF:** Я просмотрел сайт и хотел бы спросить вас о Rikki, дружелюбном встроенном роботе для проверки кода...

**КО:** Случилось так, что первые пару лет я сама отвечала на множество вопросов по обратной связи. И они снова и снова повторялись. И я поняла, что для каждого упражнения есть свои потенциальные проблемы, ряд подходов и достаточно типичных ошибок, которые люди допускают. И мне подумалось, что проведя небольшой обратный анализ, многое можно автоматизировать. Отсюда и идея: посмотреть на дерево синтаксиса и проверить, нет ли каких-либо красных флажков, которые я закодировала. Это и делает робот Рикки.

**LXF:** Проект Exercism оказался довольно успешным, кажется, вы упоминали цифру в 180 000 посетителей за два дня. Как это выдерживала инфраструктура на раннем этапе?

**КО:** Ну, я думаю, большинство тогдашних пользователей не регистрировались, не входили в учетную запись и не делали ничего особо сложного. Домашняя страница справлялась с таким уровнем трафика. Сегодня бы это наверняка был обвал, поскольку в базе данных слишком много записей и она недостаточно оптимизирована.

Сейчас у нас всё работает на Негоки. До прошлого месяца это была бесплатная Негоки, но на одной из конференций я познакомилась с одним разработчиком Негоки, и он бесплатно предоставил мне дополнительные возможности пользования платформой, так что я обновилась до профессиональной версии. Мне также дали привилегии и в использовании базы данных, но чтобы ее обновить, требуется полное резервное копирование и пере-



настройка, а у меня пока на это не хватает времени и отваги.

**LXF:** В своем докладе вы рассказывали про три категории пользователей: новички, полиглоты и мастера. Как они распределяются по уровню владения кодом?

**КО:** Новички — это те, для кого изначально предназначался сайт. Это люди, которые только начинают учиться программированию, обычно на Java, JavaScript или Python. Они уже прошли ряд учебников или упражнений под чьим-то руководством и теперь хотят решать проблемы самостоятельно.

Как правило, их проблема состоит в том, как сделать этот шаг, и мы учим, как этого достичь.

Мы хотим, чтобы это не заняло у них много времени, но чтобы у них было ощущение, что они повысили уровень. Потом мы даем им более сложные задания. Мы направляем их в этом процессе, и порой они вдруг обнаруживают: «Ой, а я же не знаю, как работают циклы». Потом им приходится взяться за это, вникнуть и разобраться. Так что всё идет через маленькие победы.

Полиглоты — это уже опытные программисты, они уже знают по крайней мере два языка и могут профессионально ими пользоваться. Иногда им приходится для работы выучить Clojure или что-нибудь еще, и проще всего это сделать, выполнив много небольших упражнений.

От них можно услышать что-то вроде: «Хм-м, у вас Clojure выглядит как Ruby. Этого быть не должно. Вот так можно сделать его более похожим на настоящий Clojure». Другие полиглоты выполняют упражнения просто чтобы выучить еще один язык, или просто ради удовольствия.

Что касается мастеров, их интересует глубинное понимание и дизайнерские тонкости. Даже если это просто 20- или 30-строчная программа, им интересно, как она будет выглядеть в каком-нибудь функциональном стиле, или что будет, если взяться за объектную ориентированность и обойтись при этом без условий. Они будут ставить себе интересные цели, например, как сделать идеально красивый читаемый код, или: «А что если написать вот эту строку на Python?»

**LXF:** Вы, наверное, получили уже немало благодарностей от пользователей Exercism?

**КО:** Мне приходило множество прекрасных электронных писем. В одном из них говорилось: «Год назад я вообще не умел программировать, а теперь, благодаря Exercism, работаю младшим разработчиком». Еще одно было от программиста на Lua, она занималась играми и в конце концов устроилась в [fast.ly](http://fast.ly) для работы с Go. Самое потрясающее здесь то, что когда она начинала работать в [fast.ly](http://fast.ly), она была там единственной, кто знал, как выполнять наборы тестов. Потому что на Exercism любое упражнение сопровождается набором тестов, и она хорошо представляла, что это такое. **LXF**



# 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](https://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](https://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!**



# Тестировщик Linux

Игорь Штомпель поговорил с Вадимом Кулешовым о процессах тестирования, жизненных циклах ошибок и верификации...



**Вадим Кулешов родился в городе Брянске в 1991 г. Закончил (с отличием) Брянский государственный технический университет (БГТУ) по специальности «Математическое обеспечение и администрирование вычислительных систем». Свою карьеру в сфере контроля качества ПО (QA) начал в 2011 г., в брянской компании «НАНО Секьюрити», занимающейся производством антивирусного ПО. С апреля 2015 г. работает в Virtuozzo (о Virtuozzo — см. LXF204).**

**Linux Format:** Расскажите, чем вы занимаетесь в Virtuozzo.

**Вадим Кулешов:** Я работаю в отделе, который занимается автоматизированным тестированием продуктов компании. Начиная с должности «инженер-программист» в отделе QA (контроля качества); сейчас мой круг задач расширился, и я стал выполнять также роль тимлидера команды автотестов. Дело это достаточно увлекательное, т. к. занимается наша компания созданием гиперконвергентной платформы на основе решений для виртуализации и хранения данных.

Вопреки всеобщему заблуждению, мы (QA) не занимаемся генерацией головной боли для отдела разработки — мы проверяем, насколько наш продукт соответствует тому, что от него желают получить наши пользователи.

В основном наш отдел занимается развитием нашего собственного тестового фреймворка, написанием новых автоматических тестов, доработкой существующих, ну и, конечно же, разбором результатов их работы с исследованием найденных ошибок. Я пришел в компанию в очень интересный момент, как раз когда только начиналось тестирование Virtuozzo 7; в новой версии продукта гипервизор собственной разработки был заменен на *KVM/QEMU*, в связи с этим пользовательские утилиты *prlsrvctl*, *prlctl* были интегрированы с *QEMU* с помощью *LibVirt*. Сейчас они позволяют управлять как контейнерами, так и виртуальными машинами; также имеется возможность использовать стандартные утилиты *virsh*, *virt-manager* и другие приложения, интегрированные с *LibVirt*. Всё это вылилось в то, что с самых первых дней я погрузился в увлекательный процесс адаптации тестового фреймворка к работе с *Virtuozzo 7* — конечно же, с сохранением обратной совместимости. Прямо сейчас начался еще один знаковый момент для нашего отдела; мы погрузились в тестирование наших новых продуктов, основанных на *OpenStack* и *Kubernetes*. Это очень перспективное направление, а главное, еще и очень интересное. К примеру, прямо сейчас мы адаптируем наш фреймворк для тестирования REST API этих продуктов, попутно изучая всю их подгоготовку.

**LXF:** Почему вы стали тестировщиком ПО?

**ВК:** Потому что мне это очень интересно. Работая тестировщиком над линейкой продуктов для серверной виртуализации, каждый день можно получить какие-либо новые знания из различных областей. В один день можно изучить «Протокол

Обнаружения Соседей» в IPv6, а в другой — работу *sarabilities* в ядре Linux. В разработке, как правило, есть четкое разграничение, кто и чем занимается: есть команды, занимающиеся разработкой в пространстве пользователя, есть команды, которые занимаются разработкой в пространстве ядра, есть команды, которые пишут *web-интерфейсы* для управления виртуализацией, и т. д. Более того, в каждой команде, как правило, тоже бывают разграничения — кто-то занимается сетевыми проблемами, кто-то занимается интегрированием. У тестировщиков таких ограничений нет — ты должен знать и постоянно изучать множество вещей. Помимо возможности расширения кругозора, существует возможность применять и постоянно совершенствовать свои умения в качестве разработчика. При написании автоматических тестов есть возможность писать код, который разнообразными способами взаимодействует с операционной системой, продуктами нашей компании и всякими сторонними компонентами. Архитектура, возможно, получается не такая сложная и интересная, как, например, в основном нашем продукте, но зато есть шансы попробовать различные технологии. А также, мы временами сами исправляем некоторые ошибки в наших продуктах, т. к. от нахождения проблемы до ее исправления большая часть времени уходит на понимание причины ее возникновения, а как раз этим мы и занимаемся.

## О КОНТРОЛЕ КАЧЕСТВА

### Вопреки заблуждению, мы не занимаемся генерацией головной боли для отдела разработки.

**LXF:** Не могли бы вы рассказать подробнее об исследовании *sarabilities* в ядре Linux?

**ВК:** Для начала надо дать определение. *Sarabilities* — это средства для управления привилегиями. Данный механизм позволяет дать процессу права на выполнение каких-либо действий, недоступных обычному пользователю, но в то же время не предоставлять этому процессу полный набор прав *root*. Основной целью было получение информации о том, при каких условиях проверяется наличие у процесса каждой из привилегий, чтобы затем подготовить тестовые сценарии, в которых будет проверяться корректное поведение при наличии и отсутствии соответствующих привилегий. Началось всё с запуска в терминале `man 7 sarabilities`, затем в некоторых случаях приходилось обращаться к исходному коду ядра Linux, ну и завершилось написанием пары десятков простейших программ на C, выполняющих по паре нужных функций.

**LXF:** На ваш взгляд, что такое тестирование ПО? Как вы понимаете процесс тестирования?

**ВК:** Зачастую под тестированием ПО подразумевают поиск ошибок в его работе. Но в действительности, на мой взгляд, процесс тестирования служит большей цели — проверке ПО на соответствие предъявляемым к нему требованиям, причем

не только явно описанным, но и подразумеваемым, которые основываются на здравом смысле или опыте использования подобных продуктов.

Лично для меня процесс тестирования состоит из нескольких основных этапов.

Первый этап — это анализ той функциональности ПО, которую предстоит тестировать. На данном этапе выявляются требования, и на их основе формируются тестовые сценарии. Причем зачастую нет необходимости проверять все возможные сценарии использования ПО, поэтому для реализации производится отбор по принципу необходимой достаточности. Следующий этап — подготовка необходимого тестового окружения, написание кода тестов, если мы говорим об автоматическом тестировании, и, наконец, получение результатов проведенных проверок. Последний этап — это анализ результатов, формализация всех полученных данных в информацию о том, насколько продукт соответствует предъявляемым к нему требованиям.

**LXF:** Какой основной рабочий инструментарий используете? Для решения каких задач?

**ВК:** Для разработки тестов мы используем Python второй версии и наш собственный фреймворк, который, к сожалению, на данный момент не находится в свободном доступе. В качестве IDE я использую *PyCharm Community Edition*. Впервые я опробовал эту среду разработки на прошлой

работе, и мне она очень понравилась — с тех пор использую только ее. Коллеги, пользующиеся *vim*'ом, смотря на нее неодобрительно.

Иногда приходится писать небольшие утилиты на C, тут помогают *gcc* и обычный текстовый редактор (*vim*). Для отладки и исследования ошибок используются утилиты *gdb*, *strace* и *tcpdump*. Ну и, конечно же, практически все основные утилиты для администрирования, такие как *top/htop*, *free*, *vmstat*, *iostat*, *ps*, *ping*, *traceroute*, *ip*, *ethtool*, *nslookup*, *netstat*, *Nmap*, *lsuf*, *df* и т. д.

**LXF:** Что представляет собой ваш тестовый фреймворк?

**ВК:** *VZT framework* — это фреймворк для автоматизированного тестирования и набор автоматических тестов для тестирования функциональности продуктов *Virtuozzo*. Основа фреймворка *vztlib* — набор модулей для описания тестов (*test suite*, *test case*), управления контейнерами, виртуальными машинами, окружением гостевых ОС и окружением серверов. Среди особенностей, которые мы реализовали — это мультиплатформенность в плане поддержки гостевых ОС, как семейства Linux, так и Windows, единый интерфейс для работы с контейнерами и виртуальными машинами, а также возможность управления ими как с помощью SDK »

нашего продукта, так и через интерфейс командной строки. Всё это позволяет, изменяя лишь опции запуска теста, проверять одни и те же сценарии использования через различные интерфейсы управления продуктом, с различными гостевыми системами, а также с различными продуктами компании.

**LXF:** Вы упомянули автотесты. Что включает система автоматического тестирования? Какие требования к ней предъявляются?

**БК:** Система включает три основных компонента: VZT framework, Deploy и VZT-results.

VZT framework — это разработанный нами фреймворк для автоматизированного тестирования, в состав которого входят сами тесты, а также система для их регулярного автоматического запуска. Набор тестов покрывает 99% всей функциональности наших продуктов, и мы постоянно поддерживаем данное соотношение. Теперь не должно вызывать удивления то, что основной состав QA в нашей компании — это разработчики.

Deploy — система разворачивания подготовленных образов ОС, удаленной установки различных ОС как на реальное железо так и в виртуальные машины; также поддерживается выборочная установка продуктов компании, сред разработки и тестирования. Кроме этого, он хранит информацию об оборудовании, принадлежащем нашему отделу, и выполняет функции менеджмента ресурсов, таких как сервера, на которых производится запуск тестов, их IP-адресов, имен и IP-пулов для тестов.

**LXF:** В чем преимущества использования системы непрерывной интеграции?

**БК:** Суть непрерывной интеграции (Continuous Integration, сокращенно CI) заключается в постоянной, после каждого коммита, сборке и тестировании разрабатываемых продуктов в среде, максимально приближенной к боевой, с целью как можно быстрее обнаружить ошибки и оповестить разработчиков. Когда продукт состоит из множества компонентов, разрабатываемых различными командами, общей численностью свыше ста человек, такой подход позволяет значительно снизить трудоемкость интеграции этих компонентов.

**LXF:** Используете ли в работе матрицу соответствия требований (Traceability matrix)?

**Если да, в каких случаях? В чем особенности ее использования?**

**БК:** В явном виде — не используем, но фактически, если рассматривать тест-план на VZT-results, то он имеет с ней много общего. Используя его, можно увидеть соотношение между тестами, которые проверяют определенные функциональные требования к продукту, и количеством сценариев, которые продукт успешно проходит.

**LXF:** В чем особенности процесса интеграции готовых решений и своего кода для автоматического тестирования?

**БК:** Мы практически не используем какие-либо готовые решения для тестирования, лишь иногда — отдельные их компоненты. Но в любом слу-

исправляющих данную проблему, баг закрывается, а в информацию о баге заносится версия компонента, в которую попал этот коммит — это позволяет перезапустить тест в автоматическом режиме для верификации его исправления.

**LXF:** Как в Virtuozzo понимаются «критерии входа»? Не могли бы вы привести несколько практических примеров?

**БК:** Критерии входа определяют момент, когда можно начинать тестировать новую функциональность. Для нас таковыми критериями являются наличие описания функциональности и появление сборки, где он присутствует хотя бы в минимально-рабочем состоянии. В нашей системе управления проектом задачи для разработчиков подвергаются декомпозиции, и мы в удобном виде можем отслеживать момент завершения подзадач и начинать их тестирование. Например, функциональность резервного копирования мы начали тестировать, когда было готово сохранение и восстановление резервных копий виртуальных машин в выключенном состоянии в пределах одного сервера. А по мере завершения работы над задачами сохранения резервных копий включенных виртуальных машин или функциональности восстановления резервной копии на удаленный сервер мы просто разблокировали уже готовые тесты. Для этого у нас используется механизм, который позволяет указать в тесте определенного сценария связь с идентификатором задачи, до закрытия которой результат теста независимо от наличия ошибок будет считаться положительным. Когда же задача закрывается, в случае наличия ошибки эта функциональность рапортует о найденном несоответствии статуса задачи и статуса прохождения теста.

**LXF:** Расскажите о процессе верификации в Virtuozzo. В чем в вашей компании особенности его организации?

**БК:** Я объединю ответы на оба вопроса, т.к. мне сложно говорить о процессе верификации в нашей компании и не упоминать о его особенностях.

Все баги в нашем баг-трекере можно разделить на две категории: созданные вручную и заведенные нашей системой запуска автотестов. Последние помечены специальным тегом и имеют привязку к конкретному тесту в тест-плане. Соответственно, в дальнейшем как раз эта связь и используется при верификации.

Пока баг находится в открытом состоянии, он блокирует запуск теста или нескольких тестов, если они имели схожую проблему и были помечены как его дубликаты. При готовом решении проблемы разработчики либо вручную указывают версию компонента, в которой он был исправлен, либо, при наличии коммитов в системе контроля версий, исправляющих данную проблему (идентификатор бага указывается в описании коммита), просто закрывают баг без каких-либо дополнительных действий, и тогда версия проставляется автоматически при попадании коммитов в очередную сборку. После очередной сборки продукта и прохождения им валидации автоматически выполняется поиск закрытых и еще не верифицированных багов

## ОБ УПРАВЛЕНИИ ПРОЕКТОМ

### В нашей системе управления проектом задачи для разработчиков подвергаются декомпозиции.

VZT-results — система учета и обработки результатов автоматизированного тестирования. Именно благодаря этой системе мы всегда можем понять, когда (в какой версии сборки) появилась проблема, сравнить поведение продукта и окружения при использовании различных гостевых ОС или при использовании различных интерфейсов управления продуктом, а также проследить динамику изменения скорости выполнения различных сценариев использования продуктов.

Из перечисленных компонентов системы только фреймворк и автотесты находятся в состоянии постоянного совершенствования; остальные компоненты уже хорошо отлажены, ими удобно пользоваться, и они предоставляют всю необходимую функциональность.

**LXF:** А методики тестирования?

**БК:** Мы занимаемся автоматическим функциональным тестированием, с использованием системы непрерывной интеграции. Полный цикл тестирования продуктов занимает порядка нескольких дней, при этом в день прогоняется несколько сотен тестов для контроля деградации.

чае мы стараемся встроить эти компоненты в наш фреймворк для облегчения их дальнейшего использования в других сценариях. Такие компоненты мы храним в виде исходных кодов и собираем их прямо в тестовом окружении. Это позволяет при необходимости легко вносить в них изменения, а также всегда иметь под рукой их исходный код, чтобы при необходимости посмотреть на их внутреннюю логику.

**LXF:** Что представляет собой жизненный цикл ошибки в Virtuozzo, от выявления до закрытия?

**БК:** В большинстве своем ошибки-баги выявляются регрессионными тестами. В случае выявления неверного поведения продукта VZT framework самостоятельно заводит запись в баг-трекере. Затем подключается отдел QA: мы проводим расследование, выявляя конкретный компонент, в котором произошел сбой, а также последовательность действий, которая к нему приводит. В зависимости от знаний по этому компоненту и сложности проблемы, мы можем сами попытаться найти в исходниках причину проблемы, либо сразу же передаем ее в отдел разработки. После коммита изменений,



с версиями компонентов, меньшими либо равными версиям, попавшим в сборку. Затем разблокируются тесты, которые были ими заблокированы. В дальнейшем, при успешном проходе теста, все закрытые баги, которые были с ним связаны, помечаются как верифицированные.

Ну и конечно же, никто не мешает, при заведении бага вручную, указать для него название теста, где присутствует аналогичный сценарий использования, которым он может быть верифицирован, и пометить его соответствующим тегом; тогда для него алгоритм верификации будет соответствовать уже описанному алгоритму автоматической верификации.

Если же баг заведен вручную и не был связан с одним из тестов, тогда его верификация ложится на плечи отдела ручного тестирования.

Они с некоторой периодичностью делают выборку по таким багам и выполняют их верификацию.

**LXF:** Не могли бы вы привести несколько практических примеров верификации с привязкой к уровням тестирования?

**ВК:** Специфика тестирования в нашей компании такова, что мы не используем модульное тестирование как таковое. В качестве уровня интеграционного тестирования можно было бы назвать валидацию сборки — это один из тестов, которые выполняет минимальный набор действий, проверяющий, что все компоненты взаимодействуют корректно и базовый набор функциональности нашего продукта работает без явных ошибок. Соответственно, все остальные тесты и верификация выполняются на версиях, которые прошли

валидацию и примерно соответствуют уровню системного тестирования.

**LXF:** В чем особенности процесса отладки при тестировании?

**ВК:** Я полагаю, что главной отличительной особенностью является то, что при тестировании приходится выполнять отладку различных сторонних компонентов, а это влечет за собой ряд проблем, главная из которых — это незнание архитектуры и логики работы исследуемого компонента. Даже при наличии исходных кодов иногда не просто бывает разобраться в той или иной логике.

Например, из нашей недавней практики могу вспомнить случай, когда потребовалась отладка Network Manager'a. При настройке интерфейса с его помощью через интерфейс командной строки »



(утилита *nmcli*) используется такая сущность, как *connection*, у него есть поле с текущим состоянием. Мне был интересен переход из состояния «активируется» в состояние «активирован», чтобы контролировать момент получения IP-адресов от DHCP-сервера. В некоторых случаях *connection* получал статус «активирован» задолго до получения IP-адресов. Причиной тут было то, что при одновременном наличии IPv4- и IPv6-конфигурации было достаточно, чтобы был получен хотя бы один из адресов. Соответственно, при конфигурации статический IPv6- и динамический IPv4-адрес

В большинстве случаев продукт рассматривается в качестве «черного ящика [black box]», т. е. проверяется правильность поведения программного обеспечения при различных входных данных и различном внутреннем состоянии ПО, без предварительного изучения кода (основываясь на спецификации). Однако для лучшего покрытия тестами различных специфичных ситуаций иногда используется так называемое тестирование «серого ящика [grey box]». В таком случае для написания тестов на правильность поведения используется изучение кода, это позволяет написать более

затраты на поддержку уже готовых тестов минимальны, а также они могут выполняться в любое время суток и всегда предоставляют отчеты о ходе выполнения.

#### **LXF: В чем вы видите особенности автоматического функционального тестирования в Virtuozzo?**

**БК:** Одна из особенностей в том, что у нас очень мало ручного тестирования, и оно в большинстве своем не заменяет автоматическое, а дополняет его для нахождения тех проблем, которые пропустили тесты, в силу того, что их не смогли учесть во время разработки теста или реализации их проверки потребовала бы слишком большого количества ресурсов. Другая особенность в том, что мы сами анализируем функциональность продукта и реализуем в тестах те сценарии, в которых, на наш взгляд, вероятнее всего будут обнаружены проблемы. Это не значит что мы не используем Test Requirements Document (TRD) — это значит, что мы используем не только его, но и сами проводим некоторый анализ функциональности продукта, зачастую заглядывая в исходный код.

#### **LXF: Интересные случаи, выявление недокументированных возможностей...**

##### **Не могли бы привести несколько примеров?**

**БК:** Если говорить о наших продуктах, то выявление таких возможностей было скорее связано с появлением большого объема новой функциональности и проблемам со скоростью обновления документации в связи с выходом Virtuozzo 7, но сейчас все проблемы уже решены. А что касается сторонних компонентов, то чаще всего в качестве документации сразу используется исходный код, а в нем все возможные варианты использования документированы. **LXF**

## **О НЕДОКУМЕНТИРОВАННЫХ ВОЗМОЖНОСТЯХ**

### **Чаще всего в качестве документации сразу используется исходный код, а в нем — все варианты.**

*connection* всегда имел статус «активирован». Если бы я входил в команду разработчиков утилиты *nmcli* и *Network Manager'a*, то такое поведение, вероятно, было бы мне понятно, а так пришлось потратить некоторое количество времени на выяснение и придумывание обходного пути для решения возникшей проблемы.

#### **LXF: Расскажите подробнее об автоматическом функциональном тестировании.**

**БК:** Функциональное тестирование рассматривает соответствие разрабатываемого продукта тем требованиям, которые к нему предъявляются заказчиком по спецификации, т. е. проверяются различные сценарии использования, исходя из функциональности, предоставляемой продуктом.

эффективные тесты, которые смогут выявить проблемы с использованием слишком коротких/длинных наборов данных, данных в неверном формате или с недопустимыми символами.

В автоматизированном функциональном тестировании, в отличие от ручного, все основные функции, такие как развертывание окружения, инициализация состояния, выполнение определенного сценария использования и получение результатов, происходят в автоматическом режиме.

Несмотря на сложность и начальные затраты ресурсов на написание и внедрение автоматического тестирования, в дальнейшем проверка одних и тех же сценариев занимает гораздо меньше времени, гарантированно проверяются все заданные параметры (исключен человеческий фактор),

# softline®

Cloud Software Hardware Services

# 20+

Years in IT

## IT-архитектура вашего бизнеса



корпоративные  
мобильные решения  
на платформе Microsoft

**SaaS**

### CLOUD

PROJECTS

гибридные облака

**SOLUTIONS**

remote workplace

**SAM**

поставка аппаратных  
решений

юридическая  
поддержка

education

Mirapolis

**SECURITY**  
учебный  
центр

разработка IT-стратегий

hardware

ERP&CRM

CAD  
CAM  
CAE

### VIRTUALIZATION

web-разработка legal support

**ЗАЩИТА  
персональных данных**

### SOFTWARE

публичные облака

САПР и ГИС

LINUX

**программное  
обеспечение**

BI & BigData

### SERVICES

WinSpace

**информационная  
безопасность**

решения Microsoft

data leak  
prevention

**PaaS**

разработка информационных систем

**корпоративный портал  
DeskWork**

open  
source

техническая поддержка

search

flexible

**CONSULTING**

PLATFORM

**IaaS**

частные  
облака

protection

hosting

technical support

16+

8 (800) 232-00-23

+7 (495) 232-00-23

www.softline.ru

info@softline.ru



# Ставим Linux на MacBook

Ник Пирс вдыхает новую жизнь в Mac, установив Linux вместе с OS X с разделом данных совместного пользования.

**У**читывая, что macOS основана на Unix, можно подумать, что Mac будет куда дружелюбнее к Linux, чем на самом деле. Но, наверное, справедливо и обратное: большинство итераций Linux не устроят пользователей Mac, которые верят в мантру от Apple «это просто работает [it just works]», которая застряла в мире с одной кнопкой.

Тем не менее, зачем ограничивать себя рамками запертого сада Apple, если вы можете — с помощью этой статьи — установить дружелюбную к Mac версию Linux параллельно OS X?

Наш принцип двойной загрузки позволит вам сохранить доступ к OS X, и мы даже проведем вас по процессу настройки раздела данных совместного пользования — и переноса домашней папки Mac

в него — поэтому и OS X, и Linux смогут легко получать доступ к главным папкам. Мы также взяли версию Linux — Elementary OS — разработанную, чтобы пользователи Mac чувствовали себя комфортно, и способную похвастаться стильным рабочим столом со знакомыми элементами, включая док внизу, строку меню и минималистский дизайн.

Elementary является особенно хорошим выбором для пользователей Mac, потому что здесь имеется встроенная поддержка чтения/записи для файловой системы Mac HFS+. Есть и подводные камни: ей не нравится встроенное в Mac шифрование FileVault, поэтому обязательно отключите

его перед началом работы, и вам также понадобится отключить журналирование на разделе данных — мы проведем вас по быстрому процессу. В нашей статье мы не только покажем вам, как получить доступ к домашней папке Mac из Elementary, но и объясним, как создать символичные ссылки из папок пользователей Elementary на ваши папки Mac в разделе данных общего пользования.

Всё, что вам нужно, чтобы настроить ваш Mac (старый или новый), установить Elementary и приступить к работе, о чем мы всегда говорим: прежде чем приступить к делу, стоит позаботиться о создании

полной резервной копии вашего Mac с помощью Time Machine. А после этого можете начинать читать следующую страницу и приступить.

**Мы взяли версию Linux, разработанную, чтобы пользователи Mac чувствовали себя комфортно.**

**М**ы создадим двойную загрузку Elementary с OS X. А также настроим раздел данных для общего пользования между этими двумя ОС, чтобы у вас был простой доступ к вашим данным. Затем мы переместим вашу пользовательскую папку Mac на раздел данных.

В нашу задачу входит разбиение диска на разделы. Первый шаг — создать два раздела, выделив достаточно места на раздел данных общего пользования и — далее — разделы для Elementary. Разбиение на разделы можно провести из самой OS X, с помощью инструмента *Disk Utility* — весьма иронично, но чем древнее ваша версия OS X, тем лучше этот инструмент.

Сначала определитесь, сколько места на диске вы хотите отвести под свой раздел данных. Откройте *Finder*, щелкните правой кнопкой по папке пользователя в разделе *Favorites* в панели навигации и выберите *Get Info*. Запишите его текущий объем и спросите себя, какой добавочный объем вам понадобится — скажем, 20 ГБ к текущей цифре. Затем надо добавить места для установки Elementary — минимум 12 ГБ (8 ГБ плюс 4 ГБ на раздел подкачки), но если вы планируете в Elementary играть, рекомендуем отвести 24 ГБ или больше.

Далее выясните, сколько свободного места на диске. Если там как минимум на 2 ГБ больше, чем вы хотите отвести под свои новые разделы, перейдите к следующему шагу; если нет, временно переместите файлы в вашей домашней папке на диск резервного копирования, чтобы освободить место (или — если большинство ваших файлов в iCloud — временно отключите его, следуя рекомендации в следующем разделе). Мы скажем вам, когда можно будет безопасно вернуть их назад. Затем разбейте свой диск на разделы в соответствии с инструкцией во врезке внизу.

По умолчанию на вашем диске данных включено журналирование — это означает, что вы ограничены в Elementary доступом только на чтение. Чтобы записывать файлы на диск — например, через папку общего доступа **Documents** — откройте *Applications > Utilities > Terminal* на своем Mac и введите

```
$ diskutil list
```

```
Last login: Mon Apr 17 15:19:56 on console
Nicks-MacBook-Pro:~ nickpeers$ diskutil list
/dev/disk0 (internal, physical):
#:          TYPE NAME             SIZE IDENTIFIER
0:          GUID_partition_scheme *251.0 GB disk0
1:          EFI EFI                 209.7 MB disk0s1
2:          Apple_CoreStorage Macintosh HD 88.4 GB disk0s2
3:          Apple_Boot Recovery HD   650.0 MB disk0s3
4:          Apple_HFS Data           129.5 GB disk0s4
5:          Linux Swap                4.1 GB  disk0s5
6:          Linux Filesystem          28.1 GB  disk0s6

/dev/disk1 (internal, virtual):
#:          TYPE NAME             SIZE IDENTIFIER
0:          Macintosh HD             +88.0 GB disk1
           Logical Volume on disk0s2
           05F36811-C046-4296-8F78-8F33A071DAC8
           Unencrypted

Nicks-MacBook-Pro:~ nickpeers$ diskutil disableJournal /dev/disk0s4
Journaling has been disabled for volume Data on disk0s4
Nicks-MacBook-Pro:~ nickpeers$
```

» Если хотите делать записи на диск в Elementary, отключите журналирование на своем разделе данных.

Найдите свой раздел данных в списке и заметьте его идентификатор. Введите `$ diskutil disableJournal /dev/disk0s4`. Это отключит журналирование для раздела. Обязательно замените `disk0s4` на отмеченный вами идентификатор раздела.

## Переместите свою домашнюю папку

После этого вам надо переместить свою домашнюю папку на раздел данных, чтобы вы могли делить его содержимое с установкой Elementary. Перед этим рекомендуем временно отключить iCloud Drive, если вы его используете: откройте *System Preferences* из меню *Apple* и выберите *iCloud*. Снимите отметку с *iCloud Drive* и выберите *Remove from Mac [Удалить с Mac]*, когда вам это предложат.

Затем откройте два окна *Finder*, одно — указывающее на ваш главный раздел, а другое — на ваш раздел данных. Откройте папку **Users** и перетащите свою пользовательскую папку на раздел данных. Велите копировать файлы, а не перемещать их. Если копирование файлов прошло нормально, снова откройте *System Preferences*, но на сей раз выберите *Accounts* или *Users & Groups*.

Нажмите на замок, чтобы внести изменения, введите свой пароль, затем щелкните правой

кнопкой по своему имени пользователя и выберите *Advanced Options*. Нажмите *Choose* рядом с Домашней директорией, чтобы выбрать свою пользовательскую папку на разделе данных. Пока вы находитесь здесь, запишите свой пользовательский ID и имя учетной записи. Они вам понадобятся позже. После выбора нажмите *OK*.

Прежде чем закрыть *System Preferences*, создайте нового пользователя-администратора, чья папка находится на вашем разделе OS X — это позволит вам загрузиться в OS X, даже если что-то случится с вашим разделом данных. Нажмите на **+**, дайте какое-нибудь имя вроде *Emergency* (с тем же паролем, что и у вашей основной учетной записи пользователя) и сделайте администратором. Нажмите *Create User*.

Теперь перезагрузите свой Mac; если всё идет хорошо, то по нажатию на свое имя пользователя в строке *Favorites* в *Finder* вы будете направлены на свой раздел данных. Вы должны увидеть, что это действие ни на что не повлияло.

Удовлетворившись увиденным зрелищем, удалите пользовательскую папку с раздела основной системы, снова включите *iCloud Drive* и восстановите все файлы, которые вы временно переместили туда с диска.

## Разбейте свой диск на разделы

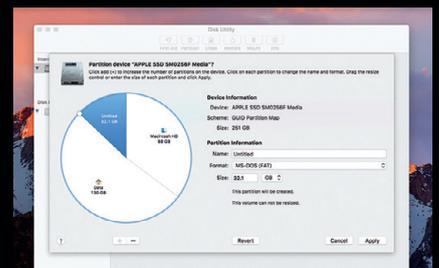
Процесс разбиения на разделы отличается в зависимости от того, какую версию OS X вы установили на MacBook. Единственное, что не меняется — способ запуска *Disk Utility*, это делается через папку *Applications > Utilities*.

Если вы работаете на OS X 10.10 или более ранней версии, то выберите свой физический жесткий диск и перейдите во вкладку *Partition*. Выберите раздел *Mac* и сократите его раздел на всё необходимое вам место с помощью ползунка. Нажмите на кнопку **+**, чтобы создать второй раздел, назовите его *Data* и затем сократите его объем, чтобы оставить достаточно места для Elementary. Нажмите *Apply*.

Если вы работаете на версии OS X 10.11 (El Capitan) или более поздней, всё будет сложнее, потому что

*Disk Utility* не позволит вам создать свободное место. Просто создайте добавочные разделы. Пока что выделите весь физический диск (не раздел OS X) и нажмите *Partition*. Нажмите на кнопку **+**; появится новый, незаглавленный раздел. Назовите его *Data*, оставьте его формат заданным как *Mac OS Extended (Journaled)* и затем настройте его размер на дополнительный требуемый вам объем — иными словами, объем раздела данных плюс место, которое вы хотите отвести под Elementary. Теперь нажмите *Apply*.

После этого выделите раздел *Data* и снова нажмите **+**. Создается третий раздел. Это временная мера, поэтому пока просто отведите ему то место, которое планируете отдать Elementary, измените его формат на *FAT32* и снова нажмите *Apply*.



» Разбиение на разделы осуществляется с помощью инструмента *Mac Disk Utility* — ирония в том, что в более поздних версиях OS X он менее мощный.

## Новый менеджер загрузки

Пора установить менеджер загрузки *rEFInd*, который позволит вам легко переключаться между OS X и Elementary — у него графический интерфейс со значками, который вы контролируете с клавиатуры с помощью клавиш со стрелками, и по умолчанию в нем стоит та ОС, которую вы загрузили последней. Скачайте *rEFInd* с [www.rodsbooks.com/refind/getting.html](http://www.rodsbooks.com/refind/getting.html) — нажмите на ссылку Binary Zip File и сохраните его в папке **Downloads**.

Если вы работаете на El Capitan или более поздней версии, вам надо временно отключить System Integrity Protection — перезагрузите свой Mac, затем нажмите и удерживайте Cmd+R, когда услышите сигнал запуска. Отпустите, когда появится логотип Apple, и должна запуститься среда восстановления. Когда это произойдет, выберите Utilities > Terminal. Введите `csrutil disable` и нажмите Enter, затем перезапустите.

Вернитесь в папку **Downloads** и найдите папку с именем **refind-bin-0.10.4** или наподобие (если ее там нет, просто дважды щелкните по ZIP-файлу с тем же именем). Откройте папку, потом откройте Терминал через Applications > перед тем, как перетащить файл скрипта **refind-install** из окна *Finder* в Терминал.

Скрипт появится в командной строке, поэтому нажмите Enter, введите пароль и снова нажмите Enter. В конечном итоге должно появиться сообщение «Установка успешно завершена»; перезагрузите свой Mac — и вы должны увидеть менеджер загрузки *rEFInd* во всей красе. В этот момент, если вы работаете на El Capitan или более поздней версии, используйте клавиши со стрелками, чтобы выбрать **Apple Recovery**; затем нажмите Enter. Снова откройте Терминал, как раньше, но на сей раз введите `csrutil enable` и нажмите Enter, чтобы снова активировать SIP. Перезагрузитесь в OS X.

## Создайте средство установки

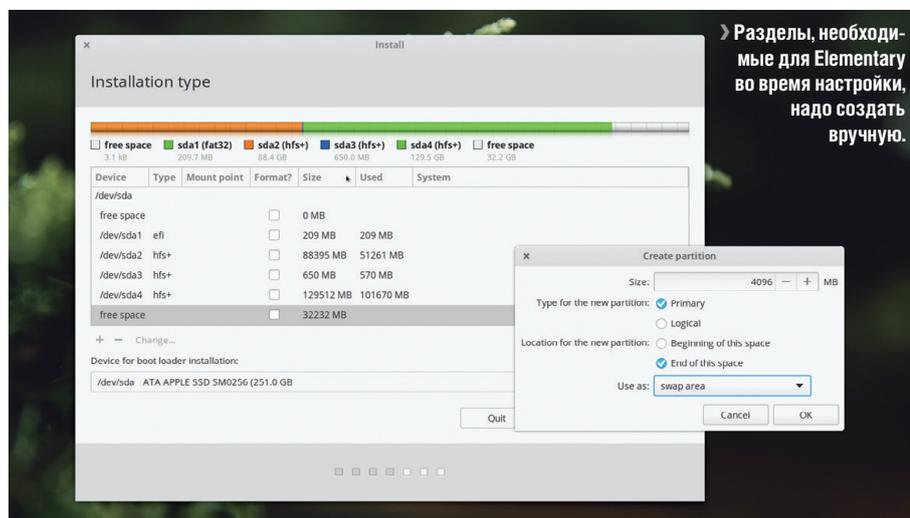
Финальная стадия подготовки включает создание средства загрузки Elementary. Перейдите в <https://elementary.io> — если вы не планируете платить, введите 0 в поле \$ Custom и нажмите Download Elementary OS. Сохраните файл ISO на жесткий диск. Пока вы ждете окончания скачивания 1,2 ГБ, скачайте и установите *Etcher* с <https://etcher.io> и используйте в качестве источника 2-ГБ USB-брелок (при желании вы можете прожечь ISO на DVD, щелкнув по нему правой кнопкой и выбрав Burn Disc Image, но это очень медленно).

Когда ISO скачается, запустите *Etcher* и выделите ISO-файл. Подключите свой USB-брелок, если вы этого еще не сделали, и он должен определиться, поэтому нажмите на **Flash!** и подождите, пока создается средство загрузки. Нажмите **Ignore** в конце процесса — OS X не будет распознавать файлы на диске.

## Время установки

И вот момент истины: пора устанавливать Elementary OS вместе с OS X. Установив средство загрузки, перезапустите свой Mac. В меню загрузки *rEFInd* вы должны увидеть значок с пингином; выберите его и нажмите Enter. Когда вам это предложат, выберите **Install Elementary OS**.

Выберите **English** и нажмите **Continue**, когда вам предложат, затем при необходимости подключитесь к сети Wi-Fi. Отметьте окошки для скачивания обновлений и сторонних программ и снова нажмите на **Continue**. Следующая стадия — самая запутанная часть установки. Elementary не определит OS X, и поэтому не предложит установить ее вместе с ней. Вместо этого вам надо выбрать **Something Else [Другое]** и нажать **Continue**. Когда появится таблица разделов, вы должны увидеть свободное место или — если вы разбивали



## Первые шаги в Elementary

Как только Elementary будет установлена и готова к работе на вашем MacBook, вы захотите с ней познакомиться. Она основана на Ubuntu 16.04, но использует собственный рабочий стол, разработанный для удобства пользователей Mac. Можете начать свое знакомство с <https://elementary.io/docs/learning-the-basics>, но структура рабочего стола весьма логичная, с доком, всплывающим меню Applications [Приложения] (подобным меню Пуск в Windows) и строкой меню наверху экрана с доступом к основным настройкам системы.

В окне Приложений нет традиционных меню — попробуйте щелкнуть левой (или правой) кнопкой по строке заглавия окна программы, чтобы посмотреть, не появятся ли какие-то дополнительные опции.

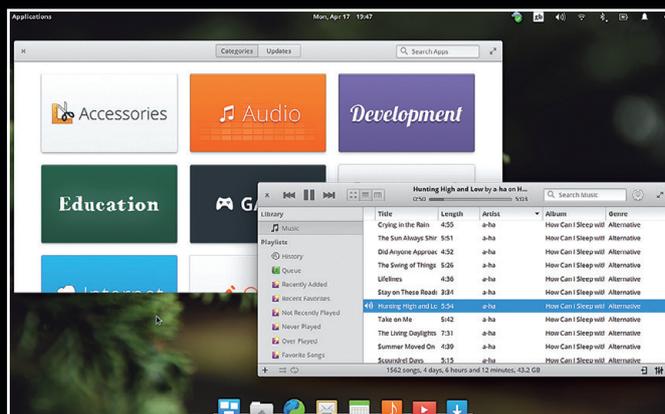
Есть и другие ограничения, которые более опытные пользователи, возможно, захотят исправить; во-первых, не установлен по умолчанию пакет `software-common-properties`, что не позволяет добавлять сторонние репозитории. К счастью, это весьма легко исправляется — просто откройте Терминал и введите следующую команду:

```
sudo apt-get update && sudo apt-get install software-common-properties
```

У нас также были некоторые проблемы с *App Center*, который работал не так, как надо. Чтобы это исправить, после установки Elementary обновитесь до самой свежей версии, опять же через Терминал:

```
$ sudo apt-get update && sudo apt-get upgrade
```

По завершении — а это довольно-таки долгий процесс — введите `sudo reboot` и нажмите Enter, чтобы перезагрузить Elementary. В результате этих действий *App Center* должен теперь работать корректно.



Интерфейс пользователя Elementary специально предназначен для удобства пользователей Mac, но у вас всё равно имеется полный доступ к Linux.

на разделы в более поздней версии OS X — раздел FAT32 в конце списка. Если он существует, выберите раздел FAT32 и нажмите на кнопку меню внизу, чтобы его удалить.

Выберите свободное место и нажмите на клавишу плюса. Установите размер в окне Create Partition [Создать Раздел] на 4096 МБ, чтобы создать 4 ГБ файл подкачки, выберите End of this Space [Конец этой области], нажмите на меню Use As [Использовать Как], настройте на Swap Area [Область Подкачки] и нажмите OK.

В конце выберите остающееся свободное место и снова нажмите +. Оставьте все настройки как есть, кроме Mount Point [Точки Монтирования] — нажмите на нее и установите в /. Нажмите на OK, затем на Install Now [Установить Сейчас]. Просмотрите изменения и нажмите Continue.

Далее проверьте правильность часового пояса, затем выберите клавиатуру Macintosh для своей страны. Когда будете устанавливать имя пользователя, рекомендуем выбрать то же имя пользователя, которое вы списали ранее с вашего Mac. Задайте подходящие имя компьютера и пароль, и затем оставьте процесс установки продолжаться до своего окончания.

## Восстановите rEFInd

Когда вам предлагают перезагрузиться, вы в первый раз загрузитесь прямо в Elementary. Первый шаг — вернуть rEFInd: откройте браузер Epiphany и скачайте тот же двоичный ZIP, что и раньше. Нажмите на ярлык Applications [Приложения] в верхнем левом углу рабочего стола Elementary и выберите Files [Файлы]. Перейдите в Downloads [Закачки], затем щелкните правой кнопкой по ZIP-файлу и выберите Extract Here [Извлечь Здесь]. Откройте вновь созданную папку **refind-bin**, затем щелкните правой кнопкой по окну Files и выберите



Open With > Terminal [Открыть с помощью >Терминал]. Наконец, введите

```
$ ./refind-install
```

Когда вам предложат, введите пароль своей учетной записи, и должен появиться целый ряд сообщений, подтверждающих, что rEFInd восстановлен в качестве менеджера загрузки по умолчанию. Перезапуститесь, и теперь вы сможете легко переключаться между OS X и Elementary.

## Доступ к домашней папке

В сложившейся ситуации вы можете видеть свой раздел данных, но любая попытка просмотреть домашнюю папку Mac приводит к сообщению, что вам не разрешено просматривать указанную папку. Это можно исправить, открыв Терминал и введя

```
$ sudo groupadd admin
```

```
$ sudo useradd -d /home/tempuser -m -s /bin/bash -G admin tempuser
```

```
$ sudo passwd tempuser
```

Введите пароль, когда вам предложат, затем перезагрузитесь и войдите как tempuser. Откройте Files и извлеките раздел данных, если он подключен (ищите кнопку eject). Затем откройте еще одно окно терминала и введите

```
$ sudo usermod --uid 501 yourusername
```

Менеджер загрузки rEFInd предлагает графическое меню, что облегчает переключение между операционными системами.

```
$ sudo chown -R 501:yourusername /home/yourusername /media/yourusername
```

Замените 501 на ID пользователя, который вы записали при перемещении домашней папки Mac на раздел данных, а yourusername — на свое имя пользователя. Последнее: вашего пользователя пока не будет видно на экране приглашения. Чтобы это исправить, введите команду

```
$ sudo nano /etc/login.defs
```

Нажмите Ctrl+W, введите UID\_MIN и нажмите Enter, чтобы найти следующую строку:

```
UID_MIN 1000
```

Замените 1000 на 501 (сделайте то же самое для строки ниже, которая начинается с GID\_MIN), сохраните файл и перезагрузитесь. Теперь вы должны быть в состоянии войти в систему под своим именем и иметь доступ к своей домашней папке Mac и всему ее содержимому. После этого удалите временного пользователя:

```
$ sudo userdel -r tempuser
```

Теперь вы готовы досконально исследовать Elementary, точно зная, что у вас есть доступ к вашим данным. Чтобы сделать ссылки на определенные папки данных, такие как Documents, с вашего Mac на вашу пользовательскую папку в Elementary, взгляните во врезку внизу. **LXF**

## Ссылки на папки пользователей

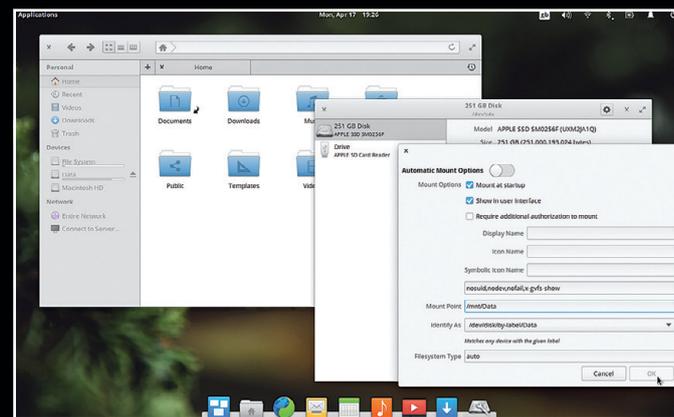
На данный момент вы получили доступ к вашей домашней папке Mac, но зачем останавливаться на достигнутом? Посредством создания символических ссылок вы можете перенаправить ваши пользовательские папки Elementary — Documents, Pictures, Music и т. д. — на соответствующие папки вашего раздела данных. Выберите, на какие папки делать ссылки — возможно, вы захотите, чтобы Downloads, например, была отдельной — но сначала убедитесь, что ваш раздел данных автоматически монтируется при запуске.

Для этого нажмите на Applications и выберите App Center. Найдите disks для установки инструмента Ubuntu Disks. После его установки откройте его из меню Applications. Выберите свой раздел данных, нажмите на кнопку Settings [Настройки] под ним и выберите Edit Mount Options [Редактировать Опции Монтирования]. Переключите Automatic Mount Options [Автоматические Опции Монтирования] на Off. Измените Identify As [Определять Как] на /dev/disk/by-Label/<Имя раздела> и проверьте, чтобы Mount Point [Точка Монтирования] была /mnt/<Имя раздела>. Нажмите OK, затем перезагрузитесь.

Теперь, получив автоматическую точку монтирования, сначала откройте свою папку пользователя Elementary и удалите те папки, на которые планируете создать ссылки. Затем откройте Терминал и введите следующее, заменив Data на имя своего раздела данных, а username — на свое имя пользователя:

```
$ ln -s /mnt/Data/username/Documents /home/username/
```

Вы должны увидеть ярлык папки в пользовательской папке Elementary. Теперь повторите это действие для всех папок, на которые хотите создать ссылки.



Перед созданием символических ссылок на свои пользовательские папки настройте Elementary на автоматическое монтирование вашего раздела данных.

# 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 меж сетевого 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 меж сетевого 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](http://www.linuxcenter.ru)

Linux  center

Наши эксперты помогут вам с любым приложением Linux!



**ЕВГЕНИЙ БАЛДИН**  
Подтвердивший  
свою квалификацию  
физик.

## Классика прекрасна и вечна

Музыка облагораживает нравы.  
*Аристотель*

Я довольно давно «положил глаз» на казуальную инди-игру *Frederic: Resurrection of Music Director's Cut*. Только что попробовал и пребываю с тех пор в некотором восторге.

Сюжет в меру бредовый, хоть и достаточно связный. Постмодернизм как он есть. Подача сюжета и рисовка довольно приятные, но ничего особо выдающегося. Игровой процесс представляет собой обычную QTE-аркаду, то есть нужно вовремя жать на клавиши виртуального пианино.

Следует понимать, что это игра для казуалов, и в ней нет ничего такого великого, кроме музыки. Великой музыки великого польского композитора Фредерика Шопена, хоть и несколько модифицированной современным исполнителем.

Почему классическая музыка — это почти всегда хорошо? Потому что а) до нас дожили действительно хорошие примеры композиций, а несущественное сгинуло во тьме веков, и б) у классики (именно музыки, а не ее исполнения) нет правообладателей.

Иными словами, можно взять за основу заведомо качественный результат человеческого гения и без оглядки на кого-либо сделать на его основе свой личный маленький шедевр.

**PS** Есть плохая и хорошая новость: W3C одобрила включение DRM-механизма в HTML5, Adobe официально прекращает развитие Flash в 2020 г. [E.m.Baldin@inp.nsk.su](mailto:E.m.Baldin@inp.nsk.su)



## В этом месяце вы научитесь...

- Делать проверки** ..... 54  
Джейсон Кэннон рассказывает, как писать скрипты оболочки, выполнение которых зависит от результатов проверки условия.
- Входить в Интернет** ..... 56  
Казалось бы, дело обычное? А вот Ник Пирс наловчился заниматься web-серфингом, не покидая командной строки.
- Защищать данные** ..... 58  
Видеокамерам Шона Конвэя не грозит перехват потока данных. Его список мер по обеспечению безопасности пригодится и вам.
- Уходить в тень** ..... 62  
Разочаровавшись в Интернете, Нейт Дрейк задумал создать свою собственную Всемирную паутину для избранных.
- Применять YubiKey** ..... 66  
Почитесь у Стюарта Бернса пользоваться удобным устройством, обеспечивающим двухфакторную аутентификацию.
- Самоидентифицироваться** ..... 70  
Ник Пирс не собирается прятать свою реальную личность: его главная задача — доказать, что он является именно самим собой.
- Управлять пакетами** ..... 74  
Вместе с Джонни Бидвеллом осваиваем азы администрирования операционной системы, под чутким руководством Linux Foundation.
- Наблюдать за домом** ..... 76  
Максим Черепанов затевает в своем Послушном Доме видеонаблюдение и изыскивает подходящее оборудование и программы.

## АКАДЕМИЯ КОДИНГА



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

- Программируем в Swift 3** ..... 80  
Подспела новая версия Swift, и Михалис Цукалос с энтузиазмом на нее набросился. Изменений, оказывается, немало...
- Пишем для Android** ..... 84  
Приложения следует штамповать в специально разработанной среде, утверждает Кент Ельчук. И такая среда отыскалась.

# Скрипты: О проверках и переменных

Учимся лаконично писать скрипты с проверками условий и переменными, под руководством **Джейсона Кэннона** из Udemy и курса Shell Scripting.



**Наш эксперт**

**Джейсон Кэннон** начал свою карьеру в 1999 г. как системный инженер Unix и Linux. С тех пор его опыт пригодился таким компаниям, как Xerox, UPS, Hewlett-Packard и Amazon.

**М**ы собираемся рассказать об основах создания скриптов оболочки. Скрипт — это программа командной строки, состоящая из ряда команд. Команды, содержащиеся в скрипте, выполняются интерпретатором — в случае скриптов оболочки, интерпретатором является оболочка (например, *Bash*), и она выполняет команды из скрипта последовательно, одну за другой. Всё, что запускается в командной строке, можно поместить в скрипт оболочки. Это отлично способствует автоматизации: если вы обнаружите, что выполняете для некой задачи серию команд и вам понадобится выполнять эту задачу в будущем, вы можете — и, пожалуй, должны — создать для этой задачи скрипт оболочки.

Рассмотрим простой скрипт оболочки, который мы назовем **script.sh**:

```
#!/bin/bash
echo "Скрипты - это здорово!"
```

Прежде чем его можно будет выполнить, следует установить для этого файла исполняемый бит — например, командой

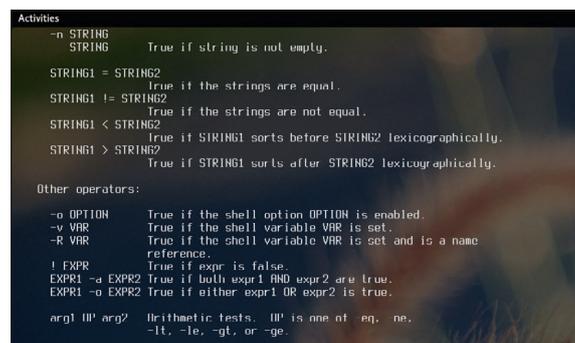
```
$ chmod 755 script.sh
```

Если скрипт находится на одном пути с вами, можно просто ввести имя скрипта, и скрипт выполнится. Если нет, надо задать скрипту относительный или полный путь. В этом случае мы делаем вид, что находимся в нашем домашнем каталоге и хотим выполнить скрипт, который мы только что там создали. Домашний каталог не в нашем пути, поэтому для выполнения скрипта надо определить путь с помощью точки и косой черты:

```
$. /script.sh
Скрипты - это здорово!
```

Первая строка скрипта начинается с «шалаша [shebang]» — двух символов `#!`, за которым следует имя программы оболочки. Мы использовали *Bash*, но существуют и другие оболочки, такие как *csh* и *ksh*. Если скрипт в первой строке содержит «шалаш», то интерпретатор и путь, используемые для вызова скрипта, передаются как аргумент интерпретатору; если же нет, команды в скрипте выполняются текущей оболочкой. Хотя во многих случаях это прекрасно работает, лучше явно и точно указать интерпретатор, который будет работать со скриптом. Например, некоторые функции и синтаксис отлично работают в оболочке *Bash*, но не работают с C-оболочкой. Также стоит отметить, что интерпретатором для ваших скриптов не обязана выступать именно оболочка — вы можете, например, запустить свой скрипт с помощью Python:

```
#!/usr/bin/python
```



» На странице справки отображается информация о доступных тестах в типично немногословном стиле документации Linux.

## Переменные

Переменные — это места хранения, снабженные именем. Можно их считать парами имя—значение. Для присвоения значения переменной используется такой синтаксис:

```
VARIABLE_NAME="Value"
```

Не ставьте пробелы до или после знака равенства.

Переменные чувствительны к регистру; по соглашению имена переменных пишутся в верхнем регистре. При использовании переменной перед ее именем ставят знак доллара:

```
#!/bin/bash
MY_SHELL="bash"
echo "Оболочка $MY_SHELL мне нравится."
```

Можно также заключать имя переменной в фигурные скобки: `$(MY_SHELL)`. Такой синтаксис не является обязательным, если нам не надо непосредственно перед или после переменной задавать еще какие-либо данные. Предположим, нам нужно добавить `ing` в конец нашей переменной:

```
#!/bin/bash
MY_SHELL="bash"
echo "I am $(MY_SHELL)ing on my keyboard."
I am bashing on my keyboard.
```

Если мы не заключим имя переменной в фигурные скобки — `$MY_SHELLing` — интерпретатор сочтет дополнительные буквы частью имени переменной. Поскольку переменной с таким именем не существует, на ее месте ничего не появится.

Можно также присвоить переменной вывод команды. Для этого заключите команду в круглые скобки и поставьте перед ними `$`:

## Скорая помощь

Термин «shebang» является приблизительной комбинацией слов «sharp» (потому что символ `#` выглядит как символ «диез» в музыке) и «bang» (иногда так называют символ «!»).



## Учитесь & экономьте с Udemу

Если вам понравилось это краткое знакомство с курсом Udemу *Shell Scripting*, вы можете получить неограниченный доступ к полному курсу на [www.udemy.com](http://www.udemy.com) с эксклюзивной скидкой для читателей *Linux Format*. Вы получите полное знание о применении стандартных инструментов терминала для администрирования, настройке тестовой машины, права доступа файлов в Linux, стандартных текстовых редакторов, работы с файлами, использовании передачи данных по сети, управлении процессами и многом другом.

### Получите скидку, начните учиться!

- » Посетите [bit.ly/SHELLSCRIPT10](http://bit.ly/SHELLSCRIPT10), чтобы записаться на курс по льготной цене £10\* (скидка 89%)!
- » Нажмите кнопку Купить сейчас и зарегистрируйте учетную запись в Udemу.
- » После регистрации учетной записи вас попросят подтвердить вашу покупку.
- » Цена £10 на курс со скидкой применяется автоматически по указанной выше ссылке. Введите данные своей кредитной карты и нажмите Оплатить сейчас.

» Вы успешно записались на курс! Наслаждайтесь пожизненным доступом.

Udemу был основан в 2010 г. с целью улучшения жизни посредством обучения. Udemу — это глобальный рынок обучения и преподавания онлайн, где более 15 миллионов студентов выбрали обучение из 45000 курсов, преподаваемых опытными инструкторами на 80 разных языках.

```
#!/bin/bash
SERVER_NAME=$(hostname)
echo "You are running this script on ${SERVER_NAME}."
You are running this script on linuxsvr
```

В этом примере вывод команды `hostname (linuxsvr)` сохраняется в переменной `SERVER_NAME`. Можно также заключить команду в обратные апострофы [backticks] — например, ``hostname``. Это старый синтаксис, его заменили только что показанным. Имена переменных могут содержать буквы, цифры и символы подчеркивания и начинаться с букв или символов подчеркивания, но не с цифры. Вот примеры допустимых имен переменных:

```
FIRST3LETTERS="ABC"
FIRST_THREE_LETTERS="ABC"
firstThreeLetters="ABC"
```

А вот примеры недопустимых имен:

```
3LETTERS="ABC"
first-three-letters="ABC"
first@Three@Letters="ABC"
```

Первое недопустимо, поскольку начинается с цифры, второе и третье — потому что они содержат знаки тире и @, а эти символы в именах переменных не разрешены.

## Проверки в скриптах

Скрипты предназначены для замены необходимости сидеть за клавиатурой и вводить ряд команд. А если у вас есть команда, которую вы хотите автоматизировать, но она требует действий, зависящих от разных обстоятельств? По мере ввода команд вы можете увидеть состояние и принять решение, исходя из этого. В сценарии оболочки то же достигается с помощью проверок условий. Чтобы создать проверку [test], поместите условное выражение в квадратные скобки. Можно проверять несколько типов ситуаций: одинаковы ли строки, больше ли одно число или равно другому, или существует ли файл. В следующем примере проверяется наличие файла `/etc/passwd`:

```
[ -e /etc/passwd ]
```

Если он существует, проверка возвращает `true` — иначе говоря, команда завершается со статусом 0. Если файл не существует, возвращается `false` — команда завершается со статусом 1. В оболочке *Bash* вы можете запустить команду `help test`, чтобы увидеть различные типы выполняемых проверок. Вы также можете прочитать ман-страницу проверки, набрав `man test`. Некоторые из наиболее распространенных проверок с оператором файлов —

- » `-d FILE True`, если файл — это каталог.
- » `-e FILE True`, если файл существует.
- » `-f FILE True`, если файл существует, и это обычный файл.
- » `-s FILE True`, если файл существует, и он не пустой.

Некоторые наиболее распространенные операторы строк:

- » `-z STRING True`, если строка пустая.
  - » `-n STRING True`, если строка не пустая.
  - » `STRING1 = STRING2 True`, если строки одинаковы.
  - » `STRING1 != STRING2 True`, если строки неодинаковы.
- Используя операторы, можно также сравнивать числа:
- » `arg1 -eq arg2 True`, если `arg1` равен `arg2`.
  - » `arg1 -ne arg2 True`, если `arg1` не равен `arg2`.
  - » `arg1 -lt arg2 True`, если `arg1` меньше `arg2`.
  - » `arg1 -le arg2 True`, если `arg1` меньше или равен `arg2`.
  - » `arg1 -gt arg2 True`, если `arg1` больше `arg2`.
  - » `arg1 -ge arg2 True`, если `arg1` больше или равен `arg2`.

Теперь, когда мы знаем, как выяснить, верно ли какое-либо условие или нет, мы можем объединить его с оператором `if`, чтобы принимать решения прямо в скриптах. Вот пример такого скрипта:

```
#!/bin/bash
MY_SHELL="bash"
if [ "MY_SHELL" = "bash" ]
then
echo "Похоже, вам нравится оболочка bash."
else
echo "Похоже, вам не нравится оболочка bash."
fi
```

Оператор `if` заканчивается символом `fi`, т.е. `if` наоборот (у программистов специфическое чувство юмора). Обратите внимание, что переменные лучше заключать в кавычки, чтобы при выполнении проверок условий предотвратить непредсказуемые побочные эффекты. Можно использовать конструкцию `else` для выполнения действия, если условие не является истинным, поэтому, если бы мы изменили значение `MY_SHELL` в приведенном выше скрипте, то он сказал бы, что нам, похоже, не нравится *Bash*.

Это всего лишь часть курса Udemу *Shell Scripting* — см. наше эксклюзивное предложение для читателей. **LXF**

```
GNU nano 2.8.1 File: show-network-port-info
#!/bin/bash
# This script displays the switch, port, and VLAN for each of the physical
# network interfaces on a Linux server that is connected to Cisco networking
# gear.
#
# Specific interfaces can be passed in on the command line. Other interfaces
# will be ignored.
#
# Set a timeout on the tcpdump command.
TIMEOUT_SECONDS=60
#
# Run as root
( "$UID" -eq 0 ) || {
echo "Please run with sudo or as root."
exit 1
}
#
# If arguments are supplied, they are a list of NICs.
NICs=$@
[ -z "$NICs" ] && {
# Get a list of physical NICs on the server that are UP; Also, ignore bonded.
# (onback, ush, virtual, tagged (0), and bridged interfaces.
NICs=$(ip link | grep '^0-9' | grep 'state UP' | cut -d: -f2 | egrep -v '^bond1|olus10')
}
#
# Loop through the NICs.
for NIC in $NICs
```

» Курс включает несколько реальных примеров, в том числе этот скрипт для отображения сложной конфигурации сети.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)

# Terminal. Входим в Интернет

Ник Пирс шарит в Интернете, собирает электронную почту, скачивает файлы и получает прогноз погоды, не покидая оболочки.



Наш эксперт

Ник Пирс работает с компьютерами уже более 30 лет, и почти десять из них балуется с Linux.

Скорая помощь

Вот аккуратный маленький трюк. Ищете прогноз погоды? Просто введите `curl http://wttr.in/~вашгород` и нажмите Enter для получения трехдневного прогноза, охватывающего утро, день, вечер и ночь. Вам потребуются изменить размер окна терминала, чтобы получить полный эффект.

Зачем вам вообще доступ к Интернету или чтение электронной почты в терминале, когда так много настольных инструментов? Ответ может быть неочевидным, когда вы находитесь в комфортной обстановке своего рабочего стола, но инструменты из этого проекта могут доказать свою ценность тогда, когда вы не сможете добраться дальше оболочки (или не захотите заходить на рабочий стол, только чтобы проверить электронную почту или просмотреть web-страницу).

## Просмотр с помощью Links

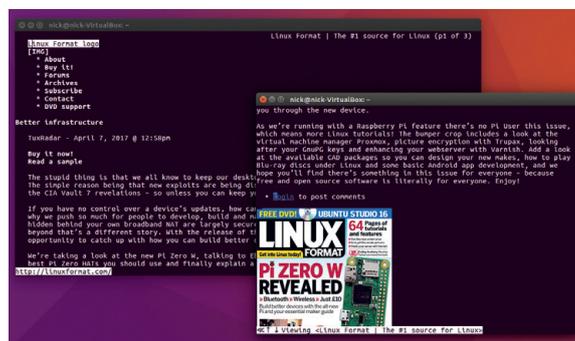
Начнем с получения доступа к Интернету. Один из вариантов — *Links*; мы выбрали вариант *Links2*, поскольку он поддерживает графику. Установите его с помощью

```
$ sudo apt install links2
```

После установки запустите его в текстовом (*links2*) или графическом режиме (*links2 -g*). Текстовый режим открывается в текущем окне терминала; графический режим открывается в отдельном окне, только если поддерживается графическая система, а именно ваш рабочий стол. В любом случае, появляется пустой экран с курсором — нажмите Esc, чтобы открыть панель меню, затем используйте мышь (только в графическом режиме) или клавиши курсора и Enter для перемещения и выбора элементов. Кроме того, ознакомьтесь с основными горячими клавишами, наиболее важной из которых является g. Нажмите ее, и вы увидите всплывающее окно Go to URL; а затем просто введите web-адрес, который хотите просмотреть — вводите `http://` не надо — и нажмите Enter.

Затем *Links* подключается к сайту — если в текстовом режиме, прокрутите вниз с помощью клавиш курсора и найдите ссылку для перехода, пропуская навигацию, чтобы сразу перейти к содержанию. Когда обнаружите ссылку, нажмите Enter, чтобы перейти по ней, и используйте клавиши z и x для перехода соответственно назад и вперед.

Вы можете нажать / для поиска определенного содержания на странице, нажать n, чтобы перейти к следующему найденному, или N, чтобы перейти к предыдущему. Картинки *Links*



» *w3m-img* имеет одно преимущество перед браузером *Links*: показ web-изображений внутри оболочки.

по умолчанию не отображает, но вы можете видеть, где они имеются на странице: в меню Вид > Настройки HTML отметьте Ссылки на изображения и Имена файлов изображений, чтобы найти их. Затем можете загрузить изображения на свой ПК для просмотра позже.

У *Links* очень подробная документация, поэтому выберите Справка > Руководство, чтобы узнать, на что он способен.

## Альтернативный браузер

Если *Links* вас не прельщает, попробуйте *w3m*. У него есть одно ключевое преимущество перед *Links*: возможность показывать web-изображения внутри оболочки. Чтобы получить эту функциональность, установите вариант *w3m-img* (`sudo apt install w3m-img`). Затем используйте его так:

```
$ w3m news.bbc.co.uk
```

Используйте курсор для навигации, нажмите Enter, чтобы перейти к гиперссылкам, и — если вы установили *w3m-img* — вы увидите встроенные изображения при прокрутке, а текст — с цветовой кодировкой, чтобы помочь навигации.

## Получайте файлы по FTP

Вы можете использовать *wget* (см. врезку сверху стр. 57) для получения файлов по FTP, но это только в одну сторону, а лучший вариант — с Linux.

Для начала введите `ftp` и нажмите Enter. Курсор изменится на `ftp>`, поэтому введите для подключения к удаленному серверу: `ftp> open ftp.example.com`.

При необходимости вам предложат ввести имя пользователя и пароль. Используйте те же команды, что и для навигации по вашей файловой системе — `cd` для смены каталога, `ls` для отображения содержимого каталога, и т.д. Введите `get filename`, чтобы скачать файл в каталог, из которого вы вызвали `ftp`, или `put filename` для его загрузки из того же каталога.

## Используйте BitTorrent в терминале

Нужно получить файл через BitTorrent? Начните с установки *rtorrent*:

```
$ sudo apt install rtorrent
```

После установки запустите его, введя `rtorrent` и дважды нажав Enter, чтобы появилась строка `load.normal>`. Просто введите или вставьте здесь URL торрента и нажмите Enter, чтобы загрузить его.

Теперь используйте клавиши курсора, чтобы выбрать торрент-файл из списка, и нажмите Ctrl+S, чтобы начать загрузку.

Вы можете остановить загрузку в любое время с помощью Ctrl+D или выйти из *rtorrent* с помощью Ctrl+Q. Примечание: до удаления торрента было бы правильно раздать хотя бы половину загрузки (50%).

## Скачиваем файлы

Хотя можно скачивать файлы и из *Links*, *wget* на самом деле является лучшим вариантом, если вы знаете URL (FTP, HTTP или HTTPS) файла, который хотите скачать. Чтобы скачать файл в текущую папку, введите следующее:

```
$ wget "http://url.com/file.zip"
```

Добавьте флаг `-O`, если хотите сохранить файл в другом каталоге:

```
$ wget -O /home/username/Desktop/ "http://url.com/file.tar"
```

Когда файл загружается, *wget* предоставляет вам индикатор выполнения, включая скорость загрузки и ETA, но если вы загружаете большой файл — ISO-образ, скажем — вы, возможно, захотите, чтобы терминал был свободен для других задач, пока загрузка не завершена. В этом случае нажмите `Ctrl+C`, чтобы отменить

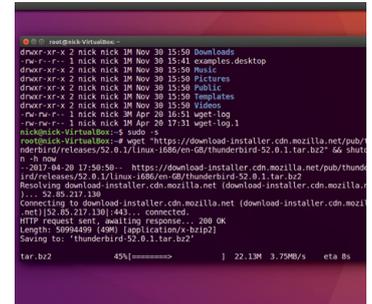
загрузку, а затем используйте флаг `-b` для загрузки в фоновом режиме:

```
$ wget -b -O /home/username/Downloads/ http://url.com/download.iso
```

Введите `ls -l -block-size=M` и нажимайте `Enter` каждый раз, когда хотите увидеть, далеко ли продвинулась загрузка, посмотрев размер ее файла. Если файл очень большой, вы можете позволить компьютеру выполнять задание всю ночь, и в этом случае он автоматически выключится, когда задача будет выполнена, чтобы сэкономить немного энергии:

```
$ sudo -s
```

```
$ wget -O /home/username/Downloads/ "http://url.com/download.iso" && shutdown -h now
```



► Можно скачивать файлы из командной строки, используя гибкую команду *wget*.

Введите `help` для получения списка поддерживаемых команд, и если *ftp* недостаточно мощности, попробуйте *ncftp* (`sudo apt install ncftp`).

## Пошлите электронное письмо

Электронная почта — не самая простая вещь, которую можно настроить через терминал, но и это возможно. Начните с установки *mailutils*:

```
$ sudo apt install mailutils
```

Во время установки вам предложат настроить *Postfix*: выберите в параметрах *Internet Site*. Оставьте имя системной почты как есть и нажмите `OK`, чтобы завершить установку.

Теперь вы можете использовать встроенные инструменты, такие как *mail*, но нужна дополнительная настройка. Если вам надо только отправлять электронную почту с терминала, проще установить *ssmtp*:

```
$ sudo apt install ssmtp
```

```
$ sudo nano /etc/ssmtp/ssmtp.conf
```

Полное руководство по настройке есть на <https://wiki.debian.org/ssmtp> — обратите внимание на последствия для безопасности хранения пароля электронной почты в текстовом файле — или, если вы пользователь *Gmail*, можете раскомментировать строку `FromLineOverride=YES`, удалив символ `#`, а ниже добавив следующее:

```
AuthUser=yourname@gmail.com
```

```
AuthPass=yourpassword
```

```
mailhub=smtp.gmail.com:587
```

```
UseSTARTTLS=YES
```

Если для *Gmail* включена двухэтапная аутентификация, вам надо сгенерировать пароль для приложения через `Account > Sign-in & Security > Signing in to Google > App Passwords`, а затем использовать его. Наконец, надо изменить строку `root=postmaster` на просто `root`.

После этого сохраните файл, закройте *nano* и скомандуйте *mail* для отправки настоящего электронного письма через *ssmtp*:

```
$ mail recipient@email.com
```

Вам предложат ввести адреса получателей копий, затем тему, и, наконец, содержание письма. Нажмите `Enter`, чтобы перейти к следующей строке, затем, наконец, нажмите `Ctrl+D`, чтобы отправить сообщение.

## Читайте электронные письма

Что делать, если вы хотите и читать электронную почту, и отправлять ее? Многие инструменты командной строки сложны для настройки — и, по сути, перебор для тех, кто не держит у себя сервер электронной почты. Но, к счастью, есть один автономный инструмент, который относительно легко настроить: *alpine* (`sudo apt`

`install alpine`). После установки запустите его, введя *alpine*, и нажмите `Enter`.

*Alpine* создаст для вас почтовую папку, а затем запустит экран приветствия. Нажмите `e` для выхода, затем `s`, а потом `l`. Выбрав *Mail*, нажмите `Enter`. Здесь вы можете настроить несколько почтовых учетных записей, поэтому используйте поле *nickname*, чтобы различать их.

Критическим полем является *Server* — здесь надо следовать следующему синтаксису (мы здесь основываемся на использовании *Gmail*, поэтому адаптируйте для своего конкретного провайдера электронной почты):

```
imap.gmail.com/ssl/user=yourname@gmail.com
```

По умолчанию почта хранится в главной папке `/home/mail`, но при желании вы можете переадресовать ее в подпапку. Оставьте поле *View* пустым, а затем нажмите `Ctrl+X` для сохранения. Вам предложат ввести пароль электронной почты, а затем нажмите `u`, если будет предложено создать почтовую подпапку и выйти.

Вернитесь к экрану установки, но теперь нажмите `s`, чтобы перейти к экрану настройки. Введите в путь SMTP следующее:

```
smtp.gmail.com/invalid-cert/user=yourname@gmail.com/ssl
```

Теперь введите следующее в путь *Inbox*:

```
imap.gmail.com/invalid-cert/ssl/user=yourname@gmail.com
```

При появлении запроса введите *INBOX* и нажмите `Enter`. Вернитесь на главный экран, затем выйдите из *Alpine*, нажав `q`, а затем `u`. Откройте *Alpine* снова и, после ввода пароля, вы увидите, что *Alpine* откроет ваш почтовый ящик и сообщит вам о его содержании. Нажмите `l` и нажмите `Enter`, чтобы просмотреть все ваши сообщения.

Посетите [www.washington.edu/alpine/](http://www.washington.edu/alpine/), чтобы больше узнать об *Alpine*, а также [www.sjwrc.com/guides/alpine-and-gmail/](http://www.sjwrc.com/guides/alpine-and-gmail/) — этот ресурс посвящен *Gmail*, но включает общий учебник и кое-какие полезные советы — например, способ получить возможность сохранить пароль в файле, чтобы не вводить его при каждом запуске *Alpine*. **LXF**



► После настройки вы сможете относительно просто собирать, читать и отвечать на вашу электронную почту из оболочки, благодаря *Alpine*.



IoT Обезопасьте свою камеру, скрыв передаваемые данные от любопытных глаз

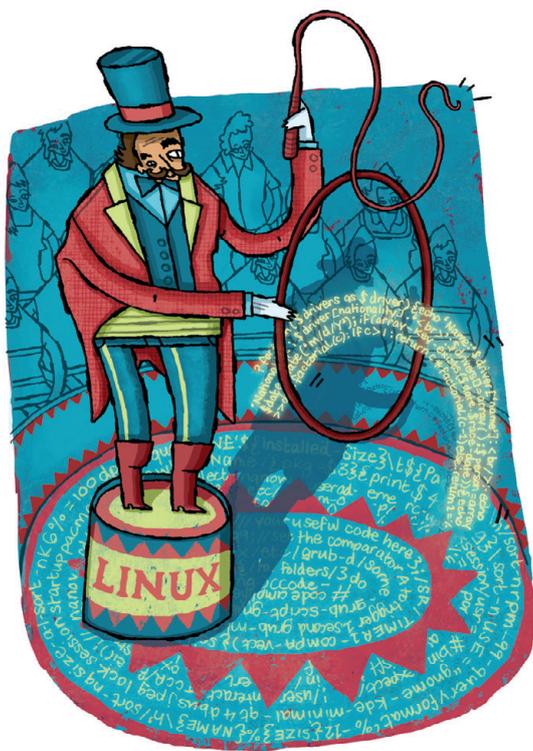
# IoT: Защита и безопасность данных

Личный опыт **Шона Конвэя** по безопасности данных устройств IoT при использовании и транспортировке помогает защитить камеры CCTV в сети.



**Наш эксперт**

**Шон Д. Конвэй** занимается безопасностью данных с тех пор, когда были разоблачены преступления Кевина Митника. Благодаря недавним подвижкам на работе сбылась его мечта, и теперь его должность — специалист по сетевой безопасности.



➤ Взаимоотношения между конфиденциальностью, целостностью и доступностью именуется триадой CIA.

of information] и доступность информации [Availability of information]. Мы взаимозаменяем слова «информация» и «данные»: в определенном контексте информация и есть данные. В нашем случае информация — это видеоданные (в данном контексте). CIA относится не только к данным; ее можно отнести и к самой камере. Взлом большого числа камер и превращение их в армию ботов для атаки типа «отказ в обслуживании» — это явная утрата доступности. Конфиденциальность информации защищает информацию от разглашения неуполномоченным сторонам. У компаний, организаций и людей обычно имеется информация, которую они не намерены сообщать кому попало. Одним из методов защиты конфиденциальности информации является шифрование, как в секретных посланиях, которыми вы обменивались с другом в школе. Только те, у кого есть ключ для дешифрации данных, могут получить доступ к информации.

Целостность информации — это защита информации от изменения неуполномоченными сторонами. Изменение информации может обернуться дорогим удовольствием. Просто подумайте, какие последствия может иметь сдвиг запятой в десятичной дроби влево в колонке с бухгалтерскими данными. Хэширование данных — метод криптографии, используемый для защиты информационной целостности. Хэш или профиль сообщения — это число, сгенерированное посредством обработки части текста специальным алгоритмом. Объем хэша меньше, чем сам текст,

**П**ару дней назад шеф вышел из кабинета в довольно расстроенных чувствах. «Всё бросай, рабочая пчелка, мне нужны от тебя тезисы для презентации про установку видеокamer в зданиях компании. Встреча с руководителями через 15 минут!».

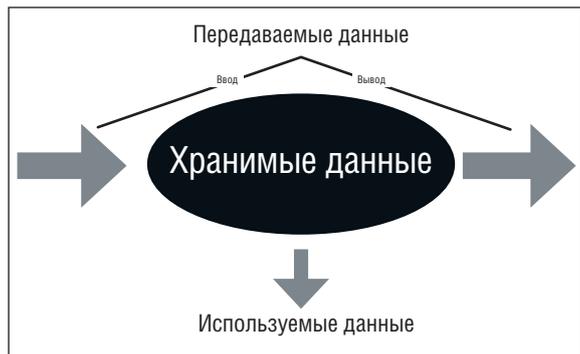
Отставим мультяшный тон Дилберта [серия комиксов об офисной жизни, названная по имени главного персонажа, — прим. пер.]: срочные задания — дело довольно обычное для отдела безопасности информационных технологий. Обычно таковые возникают в случае неотложной необходимости. Когда вы — единственный игрок на площадке, и тренер (ваш шеф) взывает о помощи, вы просто перехватываете мяч и направляетесь к воротам, даже если вы не бомбардир.

Так что же такое безопасность данных? Основной концепцией является CIA. Это не государственная организация США [CIA — ЦРУ]: эти буквы обозначают конфиденциальность информации [Confidentiality of information], целостность информации [Integrity

**Скорая помощь**



Переадресация (проброс) портов является простым способом обеспечить доступность устройства через Интернет. А также облегчает доступ к устройству плохим парням.



## Гип-гип, ура! Тройное ура данным!

и хэш уникален для этого текста. Алгоритмы создания хэш-суммы постоянно подвергаются тщательнейшему изучению, чтобы они не создавали коллизий (например, хэширование двух разных переменных текста и создание хэша с одинаковым значением).

Для обеспечения целостности хэш-сумма создается и отправляется вместе с текстовым сообщением. Хэш-сумма, созданная получателем, должна совпадать с присланной хэш-суммой, если данные не изменялись. И — нет, нельзя воссоздать текст, просто распотрошив хэш.

Доступность информации — это обеспечение возможности доступа к информации уполномоченным сторонам при необходимости. Доступность создается наличием внешних резервных копий информации. В этом случае данные можно восстановить после отказа жесткого диска. Резервирование как на главном сервере, так и на сервере резервного копирования обеспечивает доступность сервиса при отказе сервера.

Чтобы обеспечить безопасность информации с камеры (вот опять мы меняем слова информация и данные), надо определить положение данных. В любой системе могут быть передаваемые данные, хранимые данные и/или используемые данные. Передаваемые данные — это любые данные, входящие или исходящие из системы. Хранимые данные — это данные, находящиеся или хранящиеся в системе. Используемые данные — те, которые вызываются из системы, скажем, через браузер, в силу деловой необходимости. Определение положений всех данных в системе является первым шагом в процессе обеспечения безопасности данных.

Интересующие нас данные с камеры — это видео. Изображения с места наблюдения собираются (ввод), обрабатываются (хранимые данные), передаются потоком (используемые данные) и сохраняются (вывод) в сети через подключение Wi-Fi. Зная точки замеров, мы можем рассмотреть возможные шаги по управлению для обеспечения безопасности данных. Компенсаторные средства управления, как это называется, скрадывают слабости, мешающие безопасности данных.

## Анатомия камеры

Давайте рассмотрим модель камеры (рис. внизу справа). На диаграмме показаны три круга. Внутренний круг — оборудование. Это физические компоненты, которые создают камеру. Двигаемся наружу: следующий круг — это операционная система (ОС). ОС может быть проприетарным кодом, разработанным поставщиком или индивидуальной сборкой полноценной ОС, такой, как Linux, под определенную цель. Внешнее кольцо — это пользовательский интерфейс(ы). Считайте их такими сервисами, как web, интерфейсы GUI и/или приложения управления, работающие на камере для пользователя.

У всех трех колец камеры могут быть уязвимости (слабости в системе безопасности), которые оставляют данные открытыми для векторов угрозы, это термин, обозначающих точки атаки. Если до уязвимостей кто-то добрался (например, злоумышленники

## Список мер по безопасности

- Не обнародуйте того, чего не надо
- Исследуйте уязвимости и поддержку устройств
- Выполняйте частые обновления
- Управляйте учетными записями и паролями по умолчанию
- Используйте шифрование WPA2 для беспроводной сети
- HTTPS >= TLS 1.2
- Не допускайте никакой переадресации портов
- Разверните брандмауэр
- Размещайте устройства в отдельных сетях
- Как следует изучите свою сеть для регулирования скорости

## Скорая помощь

Помните, очень важно менять настройки по умолчанию. Правило брандмауэра 'any/any' означает, что любой IP-адрес брандмауэра имеет доступ к любому IP-адресу на другой стороне.

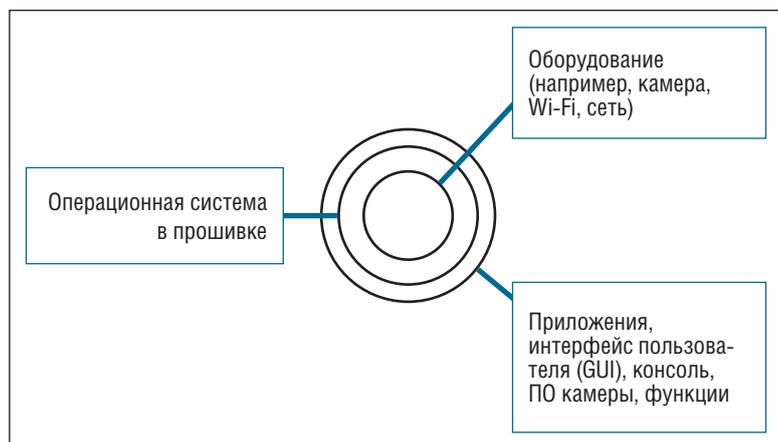
нашли способ их атаковать), то данные с камеры, которые вы хотели бы сохранить в безопасности, могут быть открыты публике. Когда данные становятся доступными тем, кому они не предназначены, они считаются скомпрометированными.

Поскольку камера для работы в сети является просто устройством IoT, то, при том же подходе с определением положения данных и их уязвимостей, вы получаете методологию обеспечения безопасности любого устройства IoT. Благодаря этому знанию вы можете проанализировать устройство IoT, чтобы определить, какие компенсаторные средства управления нужны для обеспечения его безопасности.

Чтобы применить наши знания по вопросам безопасности к обычной установке, мы поместим камеру в рядовую домашнюю сеть. Беспроводная камера передает данные на шлюз модема и роутера/резидентный шлюз провайдера сервиса (компания или интернет-провайдера, у которого вы приобретаете доступ к Интернету) через Wi-Fi. Сетевое устройство обеспечивает доступ в Интернет, а также поддержку локальной сети (LAN) для домашних ПК. Поскольку *Linux Format* распространяется в разных странах, рис. на стр. 60 является примером установки домашнего сервиса автора в Манитобе, Канада; конфигурация вашего доступа к Интернету может быть другой.

Если окончательное назначение данных камеры — точка мониторинга вне сети, то кольца безопасности, где исследуются уязвимости, должны быть расширены с камеры до других компонентов. Обеспечение безопасности данных камеры, чтобы они были доступны только уполномоченным пользователям в точке мониторинга, требует серии компенсаторных средств управления от исходной до конечной точки, для обеспечения безопасности данных от посторонних глаз.

Быстрый и легкий метод обеспечения доступности данных Интернету — настроить переадресацию (проброс) портов на сетевом



## Разрешите представить Властелина Колец камеры.

устройстве. Переадресация портов берет порт(ы) с устройства с закрытым IP-адресом на стороне LAN и выдает его в доступ из Интернета через присвоенный устройству открытый IP-адрес. В зависимости от того, как спроектирована камера или устройство IoT, для выполнения этой задачи может понадобиться переадресация одного или более портов.

Помните, что внутри колец камеры имеются уязвимости — переадресация портов может показать эти уязвимости Интернету.

Вот вам упражнение на понимание, на что способны плохие парни. Перейдите на [www.shodan.io](http://www.shodan.io) и задайте поиск брендов камер. Результат должен выявить известные уязвимости для этих брендов. Дабы не вызвать нареканий со стороны производителей, мы сознательно не упоминаем в этой статье никаких определенных камер, но бренд камеры, используемой в данном тесте, вернул пять попаданий между Европой и Северной Америкой, показавших явные уязвимости, которые так и ждут, когда их используют.

Теперь, когда мы отчасти разобрались с теорией безопасности устройства и его данных, давайте рассмотрим, как усложнить жизнь плохим парням.

Перед вводом устройства в действие неплохо проверить, не выпустили ли обновлений его производители. Обновление может включать операционную систему или приложения, поддерживаемые устройством. Использование самого нового ПО может исправить некоторые проблемы.

Производитель камеры может быть уже в курсе некоторых уязвимостей. Пользователям важно знать, что некоторые известные уязвимости не будут исправлены никогда. Предлагаемые производителем обновления не бесплатны — они стоят компании определенных затрат. Производитель устройств может решить игнорировать риск для потребителя во имя извлечения большей прибыли.

Производители редко приветствуют возможность обеспечения пользователям доступа к исходному коду. Одним из преимуществ камеры, подключенной к ОС, является то, что вы имеете полный контроль за обеспечением ее безопасности. Подумайте о камере на Raspberry Pi — вы можете настроить ОС по своему собственному желанию, и никакой производитель не будет ничего за вас решать. Остановитесь на минуту, приложите руку к сердцу и поблагодарите наших цифровых отцов-основателей за открытый код.

Итак, камера обновилась; а что у нас с настройками устройства по умолчанию? Выдаваемые имена пользователей и пароли для устройств хорошо известны. Если пароль по умолчанию superuser не изменялся и устройство доступно через сеть, то вы просто проделали в камере дверь, а ключ от нее положили под коврик. Поверьте, плохие парни туда заглянут. Если перед использованием не изменить настройки по умолчанию, шансы того, что камера окажется скомпрометированной, достаточно высоки.

Когда данные камеры в сети, они передаются. Если для передачи данных используется протокол HTTP, то становятся открытыми не только данные камеры, но и аутентификационная информация для камеры. Данные в протоколе HTTP передаются открытым текстом, включая имя пользователя и пароль для камеры, которые вы не забыли сменить с настроек по умолчанию. Если же для передачи данных камеры вместо HTTP используется протокол HTTPS, то передаваемые данные зашифрованы, и они в безопасности.

Вы засекли метод шифрования, активированный при настройке подключения к Wi-Fi? Не используйте WEP. Попробуйте задать в Интернете поиск уязвимостей WEP, и вы обнаружите технологии, которые менее чем за минуту позволяют взломать работающие на WEP устройства, выдав доступ злоумышленнику. Если камера беспроводная, включите шифрование WPA2. Желая обезопасить любое устройство в сети Wi-Fi, выбирайте шифрование WPA.

На сетевых устройствах от сервис-провайдеров вполне обычна программа брандмауэра. Брандмауэры — это устройства, отделяющие одну сеть от другой с помощью средств управления. Средствами управления могут быть IP-адреса или порты. Настройка брандмауэра со списком средств управления доступом, составленным с учетом требований к данным, ограничивает открытость камеры и прочих устройств LAN. Брандмауэр можно настроить с разрешениями только для определенных IP-адресов, и IP-адреса могут быть для тех устройств, которые будут наблюдать за камерами, подключенными к сети.

## Обезопасьте периметр

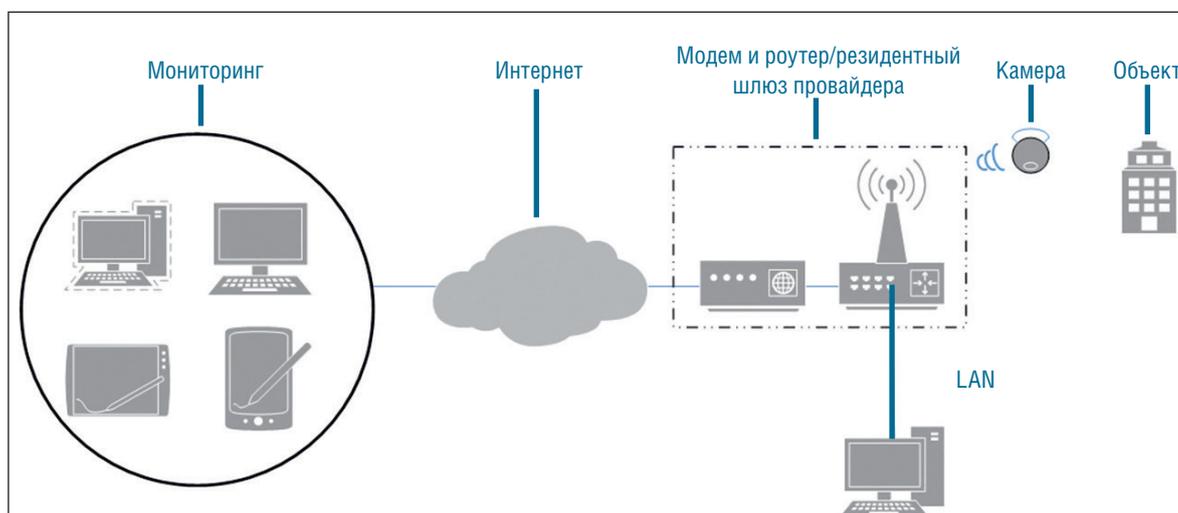
Создание ДМЗ с несколькими брандмауэрами создает многоуровневую архитектуру для сети. Плохим парням придется преодолеть больше уровней безопасности, чтобы добраться до ценного приза.

ДМЗ («демилитаризованная зона») отвечает за изоляцию ненадежной сети (Интернет) от надежной сети (LAN). Брандмауэр периметра настроен для контроля за доступом к ДМЗ, а другой брандмауэр ДМЗ настроен для контроля за доступом к локальной сети. Следует избегать установления правила брандмауэра, которое разрешает открытый IP-адрес Интернета через оба брандмауэра во внутреннюю сеть. Подобная конфигурация должна вызвать ярость у любого опытного сетевого администратора, потому что она создает слабое звено в цепи сетевой безопасности.

Внутри ДМЗ можно использовать обратный прокси-сервер для создания моста между внешней и внутренней сетью. Обратный прокси-сервер получает запросы от ненадежной сети и затем передает их на устройства в надежной сети. Ответ от конечно-го устройства передается через прокси. Обеспечиваемое обратным прокси сервером разграничение не позволяет ненадежной сети получить какую-либо информацию из надежной сети. Когда

### Скорая помощь

Подумайте о видео, которое вы делаете доступным через сеть. Наверное, не стоит настраивать потоковое вещание в Интернет с камер, которые находятся у вас в спальне.



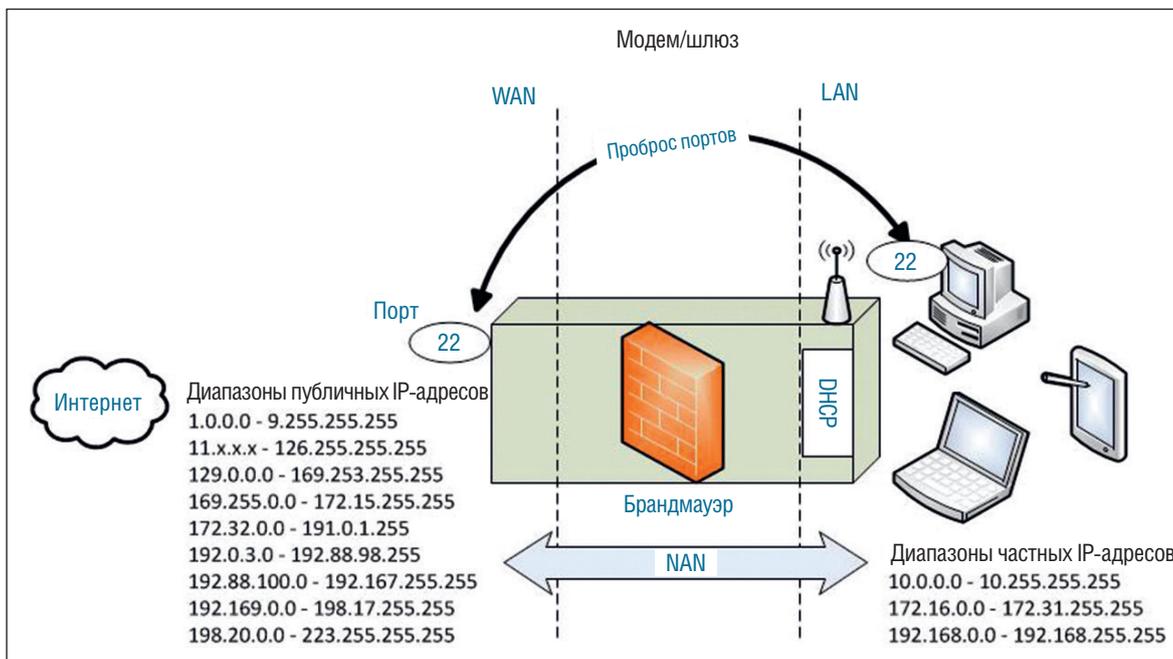
► Предоставление подключения к Интернету из-за пруда в Манитобе, Канада.

## Как убедиться, что 's' в https и вправду значит безопасность

HTTPS более надежен в зависимости от используемого стандарта. Не используйте для HTTPS стандарты SSL. SSL — устаревший стандарт, и в нем есть уязвимости. Используйте только стандарты

TLS и начните с чего-то большего, чем TLS 1.2. На момент написания даже безопасность TLS 1.2 была под вопросом из-за возможных уязвимостей. Возможно, вы помните бурю в Интернете из-за

уязвимости Heartbleed. Все web-серверы, использующие более низкие протоколы, были подвержены риску из-за TLS 1.0. Устройство, использующее >= TLS 1.2, находится в зоне более безопасного HTTPS.



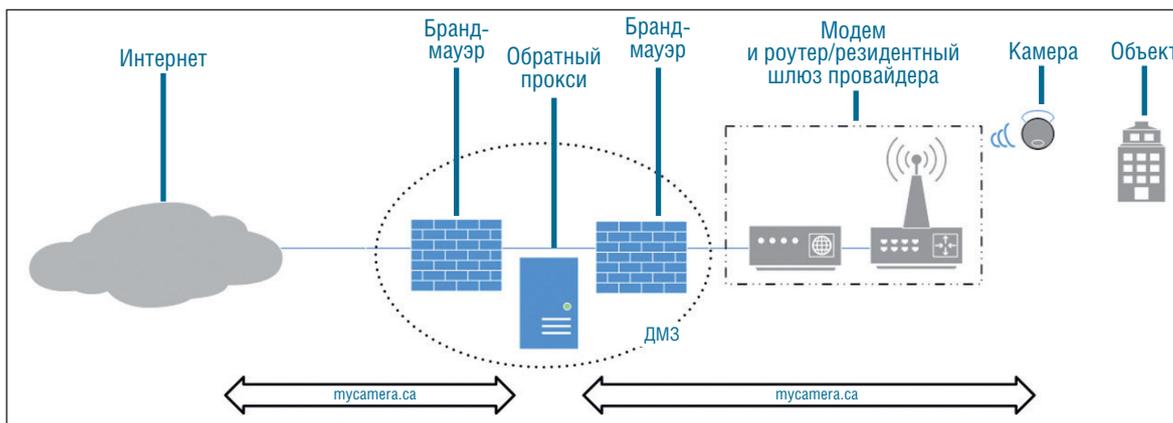
➤ Если это простейшие основы, то уж на расширенные подробности глаза бы не глядели.

на каком-либо устройстве в надежной сети имеется уязвимость, которая используется злоумышленниками и приводит ко взлому устройства, это подвергает риску все устройства в надежной сети. Если камеры в сети вызывают наибольший риск, то изоляция камер в собственную сеть с помощью брандмауэров обеспечивает большую надежность через разделение. Если камеры взломаны в изолированной сети, это не даст плохим парням доступа к остальной сети. Вы создаете один уровень безопасности за другим.

В зависимости от модели, устройства брандмауэра обеспечивают разные опции. У коммерческих брандмауэров имеется возможность регулировать пропускную способность сети. Можете считать регулятор брандмауэра чем-то вроде вентиля на шланге: вентиль регулирует подачу воды, так же и брандмауэр регулирует поток данных. Владелец сети должен осознавать структуру своей сети и характер ежедневного трафика. Если сброс больших

файлов данных по FTP не является нормой или отправка контента с камеры на неизвестный IP-адрес не является частью конфигурации, то владелец должен об этом знать. Настройка правил брандмауэра является одним из компенсаторных средств управления для обеспечения безопасности сети и устройств.

Итак, вот он: список основных моментов для выступления шефа по вопросу обеспечения безопасности камеры. Мы надеемся, что на чтение у вас ушло не более 15 минут, иначе он сильно огорчится, когда придет на собрание. Список может показаться длинным, но если вы знаете, какие данные вы хотите обезопасить, и векторы угрозы, от которых хотите защититься, вам просто нужно реализовать компенсаторные средства управления, чтобы уязвимости не стали всем видны. Теперь вы не только можете настроить безопасные камеры — ваши новые знания позволяют вам обезопасить любое устройство IoT. **LXF**



➤ Вы внутри или снаружи? Это брандмауэр.

# Tor. Создайте себе Darknet

Нейт Дрейк готов поделиться секретом: он поможет вам настроить и запустить свой сайт теневой Сети с доступом только для вас и ваших друзей.



Наш эксперт

Нейт Дрейк берет давно заслуженный отпуск от охоты за сокровищами и уворачивания от коварных наемников. Теперь он [Ред.: — Очень глупо] работает журналистом-фрилансером и пишет о технологиях.

Мы уже исследовали, как настроить «скрытый сервис» в Darknet [Теневой Интернет] с помощью Tor. Туда входит настройка сервера с помощью Apache2 и создание адреса `.onion` [onion — англ. луковица], чтобы могли подключиться другие пользователи Tor.

По умолчанию любой, у кого имеется адрес `www.facebookcore-wwwi.onion`, может подключиться к скрытому сервису Tor.

Для больших сайтов вроде Facebook в подобных действиях нет никакого вреда, поскольку все знают, кто владеет сайтом, и их бизнес-модель предполагает, что люди будут подключаться и делиться данными.

Для рядовых граждан Великобритании, которые теперь подлежат мониторингу всего web-трафика, сам факт скачивания вами серверного ПО должен быть зафиксирован вашим интернет-провайдером, как и тот факт, что вы вообще используете Tor.

Это означает, что анонимная настройка скрытого сервиса Tor — штука очень хитрая. Даже если вам удалось настроить и запустить его, подключение к нему других приводит к риску того, что один из узлов Tor будет скомпрометирован и запишет этот адрес `.onion`.

Вредоносные поисковые роботы вроде PunkSPIDER были способны составить карту всей сети Tor, проверяя скрытые сервисы на предмет уязвимостей. Само по себе это не является большой проблемой безопасности, потому что знание адреса сайта является лишь первым шагом на пути к взлому его сервера. Однако если вы просто хотите делиться информацией с ограниченной группой людей, есть способ не позволить вредоносным сканерам автоматически подключиться к вашему скрытому сервису и обнаружить реальный IP-адрес вашего сервера.

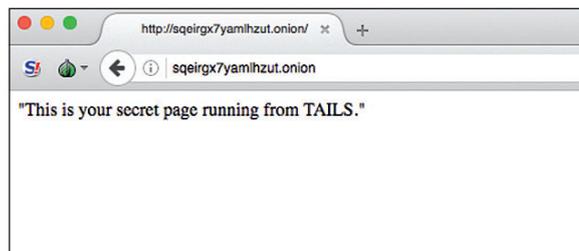
## Скрытные луковицы

Это делается с помощью луковичных сервисов 'stealth [скрытых]'. Вкратце суть их работы в том, что вы создаете уникальный адрес `.onion` для каждого отдельного пользователя вашего сайта, в дополнение к аутентификационному «куки [cookie]».

Скорая помощь



Если вы просто хотите поделиться файлами со своими друзьями через Tor, попробуйте OnionShare (<https://onionshare.org>). Она создаст уникальный адрес `.onion` для ваших контактов, чтобы скачать ваши файлы.



» Люди могут использовать свой уникальный адрес `.onion` и cookie аутентификации для доступа к вашему скрытому сервису только с вашего разрешения. Здесь Tor Browser Bundle работает в macOS.

Чтобы подключиться к вашему скрытому сервису Tor, пользователям нужен как этот уникальный адрес `.onion`, так и cookie. Любой другой человек или сервис просто видит сообщение о невозможности подключения.

Шаги данного руководства покрывают создание уникальных аутентификационных cookie и адресов `onion` для ваших контактов. Вам надо помочь друзьям настроить их Tor Browser Bundle на компьютере и/или мобильном устройстве для хранения cookie, чтобы у них был доступ к вашему сервису. Это делается посредством изменения файла настройки `torrc` внутри их программы Tor.

## Супер-TAILS

В этом проекте мы используем TAILS (The Amnesiac Incognito Live System). Эта ОС разработана с учетом защиты конфиденциальности. Все подключения направляются через сеть Tor, что снижает риск утечки вашего реального IP-адреса.

TAILS главным образом разработана для пользователей Интернета, которым нужен доступ к Darknet посредством запуска с USB или DVD и загрузки в ОЗУ. Преимущество здесь в том, что после перезапуска вашего компьютера не остается никаких следов вашей деятельности в Интернете.

Недостатком, естественно, является то, что есть данные, которые вы, возможно, захотите сохранить — например, пароли или адреса электронной почты. По этой причине TAILS предлагает специальный режим «постоянного хранения [persistent]» при установке на USB-брелке для хранения подобных данных.

Раздел **Persistent** шифруется, и вам надо задать пароль для его разблокировки при каждой загрузке TAILS. Следуйте инструкциям во врезке «Настройка Tails» внизу стр. 64.

Ваш сайт и все связанные с ним файлы безопасно хранятся в разделе **Persistent**. Это значительно повышает физическую надежность и безопасность вашего сервера, потому что диск шифруется и вы легко можете удалить или уничтожить его.

## Легковесный сервер

Tor умеет безболезненно работать с большим числом серверных программ. В этом проекте используется *Nginx-light*, который, как ясно по названию, является минимальной версией более многофункционального *Nginx*. Серверные программы работают на основе модулей HTTP, которые поддерживают базовые сайты, но не более продвинутые функции, как, например, Mail. Инструкции в руководстве также относятся к полнофункциональной версии *Nginx*, хотя увеличение поверхности атаки может сделать ваш сервер более уязвимым.

Если для вашего сервера вам нужны дополнительные функции, например, электронная почта, просто замените в руководстве и в скриптах резервного копирования 'nginx-light' на 'nginx-full'.

## Скрипты резервного копирования

Чтобы запустить сервер на TAILS, вам надо сохранить свои программы, web-страницы и клиентские ключи, чтобы при необходимости их можно было восстановить. Они хранятся на зашифрованном разделе **Persistent** на USB-брелке.

Откройте в TAILS терминал, затем создайте папку для хранения резервных копий с помощью `mkdir /home/amnesia/Persistent/tails-server`. Далее запустите `sudo nano /home/Amnesia/Persistent/backup.sh` и вставьте

```
cp /etc/nginx/sites-available/default /home/amnesia/Persistent/tails-server/
cp /etc/nginx/nginx.conf /home/amnesia/Persistent/tails-server/
cp /etc/tor/torrc /home/amnesia/Persistent/tails-server/
cp /var/lib/tor/hidden_service/hostname /home/amnesia/Persistent/tails-server/
cp /var/lib/tor/hidden_service/private_key /home/amnesia/Persistent/tails-server/
cp /var/lib/tor/hidden_service/client_keys /home/amnesia/Persistent/tails-server/
chmod -R 0777 /home/amnesia/Persistent/tails-server/
```

```
echo Done.
Сохраните и выйдите. Сделайте исполняемым:
sudo chmod a+x /home/amnesia/Persistent/backup.sh
Запустите резервное копирование:
cd /home/amnesia/Persistent
sudo ./backup.sh
Нажмите Ctrl+X, Y, затем Enter, чтобы сохранить и выйти.
Сделайте скрипт исполняемым с помощью sudo
chmod a+x /home/amnesia/Persistent/backup.sh.
Перейдите в директорию Persistent по cd /home/amnesia/Persistent, затем запустите резервное копирование — sudo ./backup.sh.
Чтобы создать скрипт восстановления, запустите команду sudo nano /home/amnesia/Persistent/restore.sh, затем вставьте следующее:
apt-get update
apt-get install -y nginx-light ufw
ufw enable
cp /home/amnesia/Persistent/tails-server/torrc /etc/tor/torrc
service tor restart
service tor stop
```

```
service nginx stop
cp /home/amnesia/Persistent/tails-server/default /etc/nginx/sites-available/
cp -f /home/amnesia/Persistent/tails-server/hostname /var/lib/tor/hidden_service/
cp -f /home/amnesia/Persistent/tails-server/private_key /var/lib/tor/hidden_service/
cp -f /home/amnesia/Persistent/tails-server/client_keys /var/lib/tor/hidden_service/
cp /home/amnesia/Persistent/tails-server/nginx.conf /etc/nginx/nginx.conf
chown -Rv www-data:www-data /var/www/
service tor start
service nginx start
echo Restored.
```

Сохраните и выйдите. Сделайте скрипт исполняемым с помощью

```
sudo chmod a+x /home/amnesia/Persistent/restore.sh
```

Чтобы протестировать, перезапустите TAILS и запустите скрипт восстановления. По `cd /home/amnesia/Persistent` перейдите в директорию **Persistent**, затем восстановите свой сервер командой `sudo ./restore.sh`.

## Приступаем

TAILS не предназначен для того, чтобы работать в качестве сервера, хотя есть планы ввести это в виде функции. Однако, поскольку он основан на Debian Linux, вы можете установить все необходимые программы, чтобы сделать его таковым.

Проект Debian также вполне подходит для размещения зеркала его серверов в Darknet. Учитывая решение правительства Великобритании о мониторинге всего трафика, это большое облегчение: благодаря тому, что вы будете делать установку через скрытые Tor, невозможно увидеть, что вы скачиваете серверное ПО вообще.

Следуйте инструкциям в руководстве, чтобы обеспечить подключение TAILS к Darknet с помощью «мостов [bridges]». Это усложнит вашему интернет-провайдеру возможность определить, что вы используете скрытый сервис.

На данном этапе TAILS не может работать без клавиатуры, мыши и монитора, поэтому выбирайте компьютер, который вам не понадобится, пока будет работать ваш скрытый сервис. Это также означает, что ваше присутствие будет необходимо при каждой загрузке TAILS, чтобы ввести пароли и восстановить ваши данные.

Сайт TAILS сообщает, что для запуска ОС нужен USB-брелок по крайней мере на 4 Гб. Обезопасьте себя и возьмите как минимум 8 Гб, ведь надо надеяться, что ваш сайт будет расти со временем.

Прежде чем начать выполнять инструкции руководства, выберите людей, у которых будет доступ к вашему сервису, и попросите их всех установить *Tor Browser Bundle* на свои компьютеры и/или *Orbot* на устройства Android. Вы всегда можете удалить или добавить пользователей по своему желанию.

## Настройте свой сайт

Руководство показывает, как создать базовую индексную страницу, чтобы сообщать вам о корректной работе сервера — однако когда вы во всё это ввязаетесь, вам, возможно, захочется большего.

Скрытые сервисы Tor могут размещать большую часть видов сайтов, хотя с учетом того, что вы запускаете свой с USB-брелка, чем легчевеснее, тем лучше. Если, например, вы хотите запустить

имиджборд, подумайте об использовании *vichan*, это ответвление ныне угасшей *tinyboard*. Зайдите на страницу каждой программы, чтобы посмотреть, не нужно ли дополнительное ПО, например, *MySQL*. Если надо, обязательно модифицируйте свой скрипт 'restore' (см. врезку вверху), чтобы оно автоматически устанавливалось каждый раз, когда вы перезапускаете TAILS.

## Защитите свой сервер

Как только ваш скрытый сервис будет готов к работе, займитесь установкой простейшего брандмауэра, открыв Терминал и запустив `sudo apt-get install ufw`. По завершении установки введите команду `sudo ufw enable`, чтобы запустить брандмауэр. По умолчанию все входящие соединения блокируются, а исходящие разрешаются.

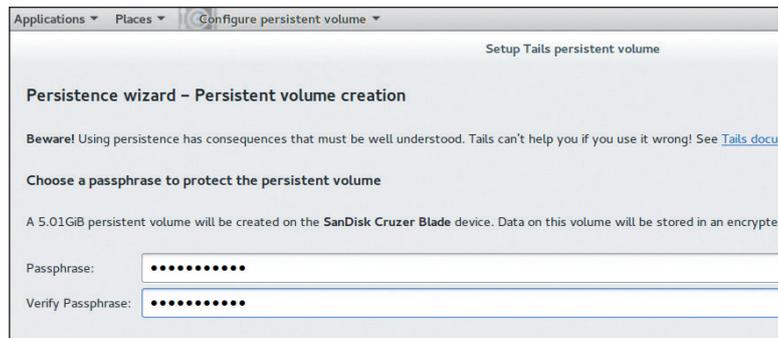
Желая включить входящие подключения на каком-то порте, используйте синтаксис `sudo ufw allow` — например, `sudo ufw allow 23`.

Когда все ваши клиенты смогут подключиться к вашему сайту, уделите время должной защите *Nginx*, запустив `sudo nano /etc/nginx/nginx.conf`.

Удалите `#` в начале строки, начинающейся с `server_tokens off`. Это означает, что если доступ попытается получить неавторизованная



В Linux **torrc** находится в `~/[path_to_tor_browser]/Browser/TorBrowser/Data/Tor/torrc`.  
В Windows перейдите в `C:\Users\[user]\Desktop\Tor Browser\Browser\TorBrowser\Data\Tor\torrc`.  
В Mac это `~/Library/ApplicationSupport/TorBrowser-Data/Tor/torrc`.



» Включите persistence, чтобы настроить зашифрованный раздел на USB-брелке. Это сохранит данные вашего сайта в безопасности.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)

личность, тот факт, что вы используете *Nginx-light*, не будет автоматически отображаться. Далее, прокрутите до раздела Basic Settings [Базовые настройки] и найдите Logging Settings [Настройки входа в систему]. Поставьте # перед строками, начинающимися с access и error.

Эта функция отключает большую часть функций входа в систему, значительно затрудняя определение того, кто подключился к серверу, хотя сообщения об ошибках всё равно фиксируются.

Теперь перезапустите *Nginx*, чтобы применить изменения, с помощью `sudo service nginx restart`.

«Облегченная» версия *Nginx*, которую вы используете, содержит голый минимум модулей HTTP для настройки базового сайта, хотя при желании вы можете добавить другие. Помните, что чем больше модулей вы установите, тем больше вероятность, что злоумышленник отыщет тот, который сумеет взломать.

## Горести TAILS

По своей природе TAILS страдает встроенной амнезией. Если вы извлечете USB-брелок или перезапустите сам компьютер, все файлы и настройки будут потеряны.

По этой причине вам обязательно надо включить persistence, следуя инструкциям во врезке внизу «Настройка TAILS», а уж потом следовать руководству.

Когда ваш сайт Darknet успешно заработает, обязательно просмотрите раздел «Скрипты резервного копирования» на стр. 63, чтобы вы могли сохранить и перезагрузить свои настройки.

Сайт Riseup (<https://riseup.net/en/security/network-security/tor/onionservices-best-practices>) содержит ряд отличных советов по работе со скрытыми сервисами Tor. Обратите особое внимание на раздел по обходу локального хоста. И обязательно всегда

```

amnesia@amnesia: ~
File Edit View Search Terminal Help
amnesia@amnesia:~$ sudo cat /var/lib/tor/hidden_service/hostname
[sudo] password for amnesia:
scnlmzdg2gyxfhfp.onion ZsWpXT9a0iPGjF7p3bg3RR # client: dick
osb7xvs3gct3rk75.onion Kd9K8cEAV0Yd5Cr0TunvDR # client: harry
y7tcduzojqejoqax.onion WlJYeX7IbLUov0w+FGGnxh # client: tom
amnesia@amnesia:~$
    
```

➤ Запустите команду `sudo cat /var/lib/tor/hidden_service/hostname` для отображения всех onion-адресов вашего контакта и их жетонов аутентификации.

обновляйте свое ПО, потому что в TAILS и скрытых сервисах Tor постоянно обнаруживаются новые уязвимости.

При всем при том, помните, что Tor поддерживает ряд скрытых сервисов, так что не бойтесь экспериментировать с запуском приватного почтового сервера или сервиса чата, не ограничиваясь простой web-страницей. Хороший подход к безопасности заключается в том, чтобы считать ключ аутентификации только одним уровнем защиты, и вы можете попросить пользователей входить на ваш сайт через индивидуальные пароли.

Помните также, что если кто бы то ни было получит доступ к вашему компьютеру, когда на нем работает TAILS, он может получить доступ также и к аутентификационной информации ваших клиентов.

Единственный безопасный способ обеспечить доступ к вашему скрытому сервису только выбранным вами людям — передать им их ключи лично. Помните, что любой, имеющий доступ к их устройству, может использовать скрытый сервис или сделать копию жетона аутентификации. Попросите их хранить *Tor Browser* на зашифрованном диске. Пользователи мобильных устройств также могут включить шифрование на устройствах Android на случай кражи или конфискации их устройства.

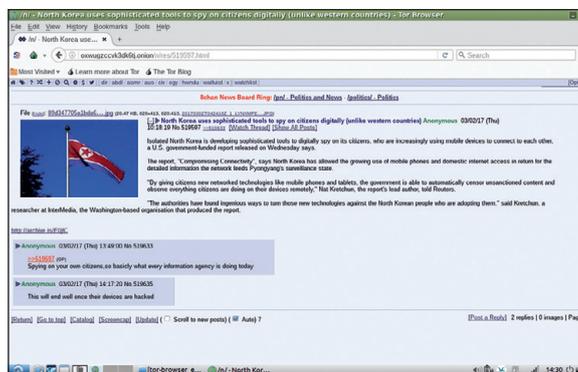
Если вас беспокоит, что ваши ключи скомпрометированы, запустите команду `sudo shred -v /var/lib/tor/hidden_service/client_keys` и перезапустите Tor с помощью `sudo service tor restart`. Система сгенерирует новые ключи для ваших пользователей.

И, наконец, помните, что пока вредоносные сканеры не смогут получить автоматический доступ к вашему скрытому сервису, они не смогут получать те же сообщения, которые они получали бы для недействительного onion-адреса — иными словами, они будут знать, что этот адрес существует, но им будет казаться, что сервер не работает. Это не самое правдоподобное отрицание, но Tor Project на него полагается. LXF

### Скорая помощь

Для укрепления безопасности измените порт, используемый вашим сервером, с порта по умолчанию (80), запустив `sudo nano/etc/nginx/sites-enabled/default` и изменив `listen 80`. Обновите также `HiddenServicePort` в `z/etc/tor/torrc`.

➤ Скрытый сервис имиджборда 8chan, показанный здесь, использует программу *vchan*. Для работы с некоторыми сайтами TAILS может понадобиться дополнительное ПО.



## Настройка TAILS

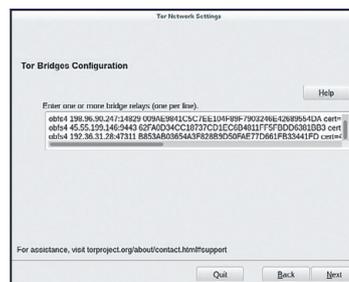
Этот проект разработан для загрузки из ОС Tails, установленной на USB-брелке. Следуйте инструкциям сайта TAILS — <https://tails.boum.org/install/index.en.html>. Емкость устройства USB должна быть не менее 4 Гб.

После установки перезапустите компьютер и загрузитесь с USB. Возможно, для этого понадобится изменить настройки BIOS. Нажмите Yes в More Options и введите пароль администратора.

По умолчанию любой, кто проводит мониторинг исходящих подключений TAILS, сможет увидеть, что вы используете Tor. Вы можете сильно усложнить обнаружение трафика Tor, посетив <https://bridges.torproject.org/options> и выбрав 'obfs4' в Advanced Options для получения адресов bridge. В TAILS нажмите на This Computer's Connection is Censored [Подключение Компьютера проверяется], затем Configure > Yes > Next [Настройка > Да > Далее]. Вставьте адреса bridge.

В TAILS по умолчанию после перезапуска компьютера все данные теряются. Однако вы можете создать зашифрованный раздел для хранения файлов и настроек. После загрузки рабочего стола нажмите Applications > Tails > Configure Persistent Volume. Выберите прочный пароль и нажмите Create. Мастер спросит вас, какие данные сохранять — обязательно отметьте Apt Packages, Apt Lists и Dotfiles, затем нажмите Save.

Перезапустите TAILS для полного включения persistence [долговременного хранения]. На экране загрузки в ответ на Use Persistence? [Использовать Долгосрочное хранение?] нажмите Yes и введите свой новый пароль. Нажмите снова Yes для других опций и Forward для выбора пароля администратора, как раньше. Следуйте инструкциям во врезке «Скрипты резервного копирования» вверху стр. 63 для сохранения и восстановления настроек вашего сервера.



➤ Используйте obfs4 bridges, чтобы скрыть сам факт использования сети Tor.

# Настройте свой сайт Darknet

```

amnesia@amnesia: ~
File Edit View Search Terminal Help
GNU nano 2.2.6 File: ...istence/TailsData_unlocked/persistence.conf Modified

/home/amnesia/Persistent          source=Persistent
/etc/NetworkManager/system-connections source=nm-system-connections
/var/cache/apt/archives           source=apt/cache
/var/lib/apt/lists                source=apt/lists
/home/amnesia                    source=dotfiles,link
/var/www                          source=www
    
```

```

GNU nano 2.2.6 File: /etc/nginx/sites-available/default Modified

server {
    listen 127.0.0.1:8080 default_server;
    server_name localhost;
    server_tokens off;
    root /var/www;
    index index.html index.htm;
    location / {
        allow 127.0.0.1;
        deny all;
    }
}
    
```

## 1 Настройте папку сервера

Следуйте «Настройке TAILS», чтобы загрузиться с USB-брелка и настроить долгосрочное хранение. Откройте Applications > Terminal, запустите `sudo mkdir /var/www`, затем `sudo mkdir /live/persistence/TailsData_unlocked/www`. Отредактируйте файл настройки долгосрочного хранения с помощью `sudo nano /live/persistence/TailsData_unlocked/persistence.conf` и добавьте в конце строку `/var/www source=www`. Нажмите Ctrl+X, затем Y, затем Enter, чтобы сохранить и выйти. Перегрузите TAILS, откройте терминал и запустите `sudo apt-get update`, затем `sudo apt-get install nginx-light`. Запустите `sudo nano /etc/nginx/sitesavailable/default`, приставьте в конце текст из шага 2, сохраните и выйдите.

```

amnesia@amnesia: ~
File Edit View Search Terminal Help
GNU nano 2.2.6 File: /etc/tor/torrc Modified

##### This section is just for location-hidden services #####

## Once you have configured a hidden service, you can look at the
## contents of the file ".../hidden_service/hostname" for the address
## to tell people.
##
## HiddenServicePort x y:z says to redirect requests on port x to the
## address y:z.
##
HiddenServiceDir /var/lib/tor/hidden_service/
HiddenServicePort 80 127.0.0.1:8080

HiddenServiceDir /var/lib/tor/other_hidden_service/
HiddenServicePort 80 127.0.0.1:80
HiddenServicePort 22 127.0.0.1:22
    
```

## 2 Настройка web-сервера

```

server {
    listen 127.0.0.1:8080 default_server;
    server_name localhost;
    server_tokens off;
    root /var/www;
    index index.html index.htm;
    location / {
        allow 127.0.0.1;
        deny all;
    }
}
    
```

```

amnesia@amnesia: ~
File Edit View Search Terminal Help
GNU nano 2.2.6 File: /var/www/index.htm

"This is your secret page running from TAILS."
    
```

## 3 Настройка сервисов stealth

Запустите `sudo nano /etc/tor/torrc`. Прокрутите до this section is just for location-hidden services [этот раздел только для сервисов со скрытым местоположением]. Именно здесь вы будете настраивать скрытые сервисы для каждого пользователя. Измените имена в последней строке перед тем, как делать вставку:

```

HiddenServiceDir /var/lib/tor/hidden_service/
HiddenServicePort 80 127.0.0.1:8080
HiddenServiceAuthorizeClient stealth tom,dick,harry
    
```

Сохраните и выйдите.

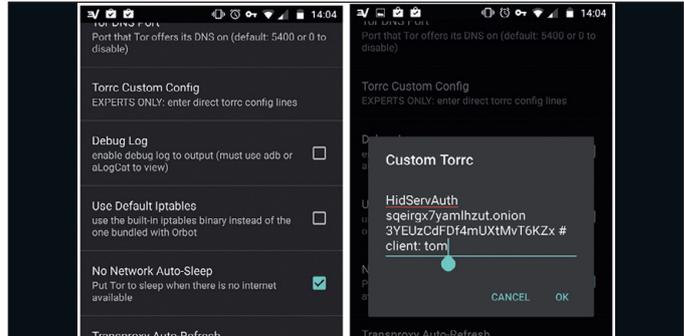
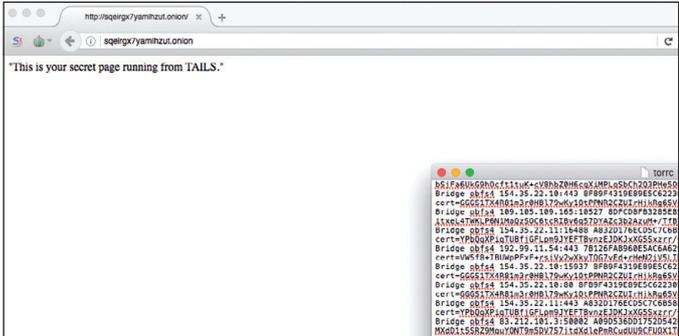
## 4 Создайте индексную страницу с разрешениями

Создайте простую целевую страницу для своего сайта с помощью `sudo nano /var/www/index.htm`. Введите какой-нибудь текст, например: «Это ваша секретная страница, работающая из TAILS», затем сохраните и выйдите. Командой `sudo chown -Rv wwwdata:www-data /var/www` предоставьте пользователю www право владения папкой /var/www.

Перезапустите Tor и Nginx с помощью

```

sudo service tor restart
sudo service nginx restart
    
```



## 5 Настройте клиентский доступ (Tor Browser)

Запустите `sudo cat /var/lib/tor/hidden_service/hostname` для перечисления жетонов доступа для всех пользователей. Попросите своего пользователя открыть папку *Tor Browser Bundle* и используйте текстовый редактор, чтобы открыть файл `torrc`. В Linux он находится в `~/(path_to_tor_browser)/Browser/TorBrowser/Data/Tor/torrc`. Введите `HidServAuth`, затем вставьте строку для адреса и cookie Аутентификации Скрытого Сервиса [Hidden Service Authentication] в конце.

## 6 Настройте клиентский доступ (Orbot)

Если ваши контакты предпочитают использовать мобильное устройство, попросите их установить на своих устройствах Android приложения *Orbot* и *Orweb*. Далее запустите *Orbot* и перейдите в Menu > Settings > Torrc Custom Config. Введите `HidServAuth`, затем оставьте место и вставьте их информацию из `hostname`.

Отныне у ваших контактов появится доступ к вашему скрытому сервису из браузера *Orfox*.

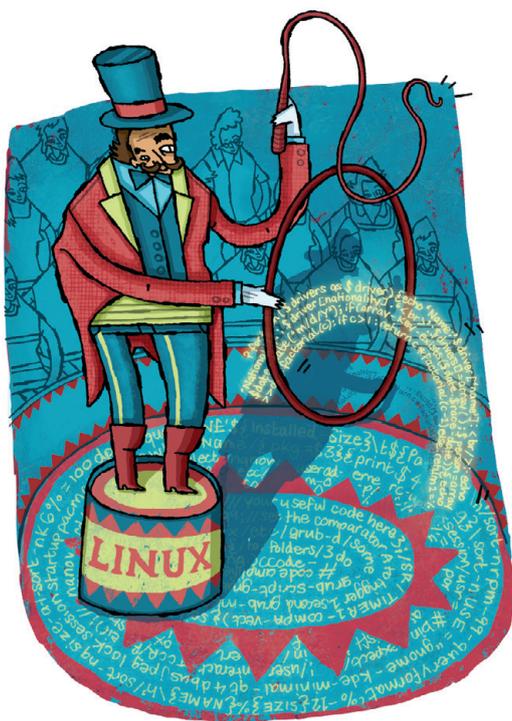
# YubiKey: Способ хранить секреты

Благоразумный **Стюарт Бернс** учит, как серьезно повысить безопасность системы Linux благодаря YubiKey и двухфакторной аутентификации.



## Наш эксперт

**Стюарт Бернс** работает системным администратором в компании Fortune 500. Когда он не занят решением сложных проблем, его можно найти безумствующим в Linux и других интересных штуках.



ряд особых случаев использования YubiKey: надежное шифрование всего диска; обеспечение безопасности ключей SSH; и, наконец, это безопасный способ входа на другой компьютер Linux через SSH.

Для нашего урока мне придется сделать ряд допущений. Первое — что у читателя есть YubiKey 4 и что это устройство не используется в данный момент для иных целей. Во-вторых, я подхожу к предмету с той точки зрения, что LUKS уже настроен и работает, как положено. Предполагается также наличие хорошей, проверенной резервной копии и что она проверена на другом компьютере, не используемом для работы. Эти действия, если они выполнены неправильно, могут привести к тому, что пользователь временно лишится доступа к своим зашифрованным данным.

## Установка

Для корректной работы в Linux YubiKey требует дополнительных пакетов. На нашем уроке используется Ubuntu 16.04. Из командного терминала установите эти пакеты:

```
sudo su -
apt-get update -y
apt-get install -y YubiKey-luks YubiKey-personalization sdaemon
exit
```

После установки пользователь сможет настроить аппаратный ключ. Важно понять, как работает YubiKey. На устройстве YubiKey есть несколько разъемов-слотов. Каждый слот может содержать определенный ключ/пароль шифрования. Технически слотов несколько, но только два первых официально поддерживают использование паролей. Это отдельно от местоположения хранения SSH.

YubiKey надо настроить для работы с шифрованием LUKS. Сначала подключите YubiKey. Чтобы инициализировать использование второго слота с LUKS, запустите следующую команду:

```
sudo ykpersonalize -2 -ochal-resp -ochal-hmac -ohmac-1t64 -oserial-api-visible
```

Примите изменения, нажав Y. Прежде чем запускать нижеприведенную команду для настройки YubiKey и LUKS, стоит проверить, что используется должный зашифрованный раздел. Это делается обычным просмотром файла `/etc/crypttab`. Используйте следующую команду для подтверждения диска:

```
cat /etc/crypttab
```

Зашифрованный раздел находится в начале первой строки. В моем случае им является `/dev/sda5`. Если во время установки были выбраны настройки Ubuntu по умолчанию, то раздел `sda5` — это всегда зашифрованный диск. Последнее, что осталось — обновить систему LUKS для использования слота 7 устройства YubiKey. Добавится второй ключ LUKS, который не будет работать без аппаратного ключа и пароля:

```
sudo YubiKey-luks-enroll -d /dev/sda5 -s 7
```

**П**оследнее время многое говорится о безопасности системы и эффективном управлении системой. Реализация на практике устройств с двухфакторной аутентификацией (2FA) помогает повысить эту безопасность.

Использование устройств вроде YubiKey имеет ряд преимуществ, включая содействие в шифровании локальных дисков, безопасность аутентификационной информации и безопасность ключей SSH от хакеров. Эти устройства предназначены для безопасного хранения данных и показывают информацию с диска только когда это требуется.

YubiKey можно использовать как отдельное устройство или зарегистрировать его с сервисом пароля YubiKey и компании для обеспечения безопасности управления паролями и автоматического управления сетевыми паролями. YubiKey также можно использовать с Facebook, Google и рядом других провайдеров для обеспечения надежных логинов 2FA ради предотвращения взлома учетной записи. На разных сайтах это реализуется по-разному, но есть масса руководств, объясняющих, как это делается.

Более того, если нажать на емкостное устройство, оно сгенерирует исключительно длинный и надежный пароль, именуемый OTP (one time password — разовый пароль) вместо запоминаемого пароля, что значительно затруднит возможность для кого бы то ни было похитить вашу информацию. На нашем уроке рассмотрен

## Скорая помощь

После настройки YubiKey убедитесь, что исходный пароль LUKS достаточно длинный и надежный. Чтобы его изменить, используйте команду `cryptsetup luksChangeKey <целевое устройство> -S <номер слота целевого устройства>`. Если ваш пароль слишком длинный, сделайте копию и храните его в надежном месте.

В этот момент пользователю надо вставить YubiKey и ввести новый добавочный пароль для использования совместно с LUKS. Введите новый, уникальный пароль. Вам предложат подтвердить его. Наконец, вас попросят ввести исходный пароль для диска LUKS. Вводите его правильно, иначе диск не станет нормально работать. Теперь остается только перезагрузить компьютер и убедиться, что новая настройка шифрования применена правильно.

После перезагрузки пользователь должен вставить YubiKey и ввести вновь созданный пароль LUKS. Если всё нормально, он должен нормально загрузиться. Если по каким-то причинам это не сработало, можно использовать исходный пароль LUKS. Исходный пароль позднее можно усилить (в противном случае он сводит на нет безопасность, обеспечиваемую YubiKey).

## Обезопасьте свои ключи

Пора рассмотреть использование YubiKey для обеспечения безопасности частных ключей пользователей. Если у пользователя несколько компьютеров с Linux, на которые он заходит по паролю, то он неправ. А вот вход с помощью ключей SSH затрудняет незаконный доступ к процессу входа в систему, который становится более сложным.

Вход в систему через ключи SSH осуществляется посредством создания двух ключей: частного (закрытого, приватного) ключа и открытого (публичного) ключа. Открытый ключ загружается на компьютеры для администрирования, а частный ключ надежно хранится на YubiKey и защищается паролем. Пытаясь войти в систему, пользователь применяет свой частный ключ, который есть только у него, совместно с паролем. Когда система получает обе части ключа, она может удостоверить личность пользователя. Без обеих частей ключа ничего не произойдет. Пароль нужен лишь для того, чтобы разблокировать частный ключ на момент его использования.

YubiKey сам по себе не реализует создания частного/открытого ключа. Чтобы настроить YubiKey, надо использовать для создания ключей пакет *openpgp2*, и затем переместить частные ключи на YubiKey. Обратите внимание, что по умолчанию установлены две версии *gpg*: *gpg* и *gpg2*. В этой статье используется *gpg2*, так как там солидно исправлен код и намного лучше работа с картами на *gpg* вроде YubiKey. Попытки использовать на нашем уроке исходный *gpg* приведут к некорректной работе некоторых действий или к полному отказу от работы.

Процесс создания ключей и настройки может показаться вам громоздким, но на самом деле он довольно прост. Следующие несколько шагов включают:

- » Создание мастер-ключа
- » Создание ключа отмены
- » Добавление подключа для входа в систему, подписания [signing] и шифрования
- » Удаление мастер-ключа из связки, чтобы его нельзя было использовать для подписания новых ключей

- » Размещение частных сертификатов на YubiKey
- » Экспорт открытых ключей SSH
- » Копирование их на серверы назначения

Первый шаг — создание мастер-ключа. В терминологии управления ключами этот ключ обычно является эквивалентом пользователя *root*. В производных Debian север создает два ключа: мастер-ключ, способный служить для удостоверения подключения; и подключ с возможностями шифрования и подписания. Это может показаться странным, зато отделяет ваш мастер-ключ (который является исключительным подтверждением вашей личности) от повседневных ключей. И за этим решением стоит солидная причина. Если подключ будет утерян или скомпрометирован, пользователю не придется начинать процесс его создания с нуля. По завершении процесса создания ключа, описанного ниже, у вас будет четыре ключа: мастер-ключ, ключ для шифрования и для удостоверения и ключ авторизации. Ключ авторизации используется для подключения к устройствам SSH.

Мастер-ключ создается командой

```
gpg2 --full-gen-key
```

— вместе с ним создадутся файлы и папки связки ключей, если их не существовало. Поскольку наш урок использует это исключительно ради примера настройки, мы можем использовать то, что предложено по умолчанию. Но в среде *live*, по очевидным причинам, такого делать не рекомендуется. Для данного ключа подойдет опция 4: Только подпись RSA.

Когда будут приняты первые вопросы о длине ключа и сроке его действия (по умолчанию), приложение спросит у вас имя и адрес электронной почты для уникальной идентификации ключа. Для целей нашего урока введите свое имя и адрес электронной почты. Комментарии опциональны. Когда вы подтвердите правильность информации, нажмите **O** для создания ключа. Система

## Скорая помощь

Если система продолжает световить на недостаток энтропии (случайности), можете сделать следующее. Установите пакет *rngtools* (`sudo apt-get install rngtools`) и запустите команду для создания случайности: `sudo rngd -r /dev/urandom`.

```
stuart@testmachine: ~
"Stuart Burns <stuart@newlymintedmedia.com>"
Change (N)ame, (C)omment, (E)-mail or (O)kay/(Q)uit? o
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilise the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
gpg: /home/stuart/.gnupg/trustdb.gpg: trustdb created
gpg: key BF4E2BD0 marked as ultimately trusted
gpg: directory '/home/stuart/.gnupg/openpgp-revocs.d' created
gpg: revocation certificate stored as '/home/stuart/.gnupg/openpgp-revocs.d/DED0772A8A7D0A99C8B39CF91B62AD6CBF4E2BD0.rev'
public and secret key created and signed.

gpg: checking the trustdb
gpg: marginals needed: 3 completes needed: 1 trust model: PGP
gpg: depth: 0 valid: 1 signed: 0 trust: 0-, 0q, 0n, 0m, 0f, 1u
Note that this key cannot be used for encryption. You may want to use
the command "--edit-key" to generate a subkey for this purpose.
pub  rsa2048/BF4E2BD0 2017-05-01 [S]
     Key fingerprint = DED0 772A 8A7D 0A99 C8B3 9CF9 1B62 AD6C BF4E 2BD0
uid          [ultimate] Stuart Burns <stuart@newlymintedmedia.com>

stuart@testmachine: ~$
```

» Вот готовая настройка ключа, где показан вывод от ключа.

## Интернет и вы

Заставить YubiKey работать с интернет-паролями в отдельности от инфраструктуры YubiKey может сначала показаться вам сложным. На самом деле это очень просто.

Надо использовать OTP YubiKey вместе с локальной и надежной системой управления паролями. Например, на Mac вы можете использовать *MacPass* (есть альтернативы для Linux. Одним из достойных примеров является *KeepassXC*, <https://keepassxc.org>), который основан на локальном устройстве

и не общается с Интернетом. Затем используйте этот OTP в качестве надежного пароля для входа на устройство.

Это означает, что перед вами встанет ужасающая задача изменить все интернет-пароли, которые защищались неким онлайн-продуктом, оказавшимся скомпрометированным. Прелесть *MacPass* и его Linux-сородичей в том, что в приложении можно создавать отличные, длинные, надежные пароли и использовать их в качестве уникальных.

Есть и подводные камни. Во-первых, обязательно делайте резервную копию файла паролей. Запишите ее на CD, который храните в сейфе. Во-вторых, надо делать резервные копии OTP. OTP — это всего лишь длинная строка. Чтобы получить доступ к этой строке, просто откройте редактор и нажмите YubiKey. Вы получите пароль OTP. Распечатайте его и храните в надежном месте. Тогда у вас будет резервная копия на случай какой-нибудь неприятности™ или если вы потеряете YubiKey.

также попросит вас ввести пароль — он предотвратит применение ключа, если ключ похитят. Система выдаст немало текста, но нас интересует открытый ключ. ID ключа находится справа от `rsa2046/`.

Как вы можете видеть, у созданного ключа имеется надпись `rsa2046/xxxxxx`, где `x` — длинное число. Это идентификатор ключа. Поскольку наш урок будет использовать этот ключ несколько раз, ваша жизнь упростится, если вы дадите ему имя. Это уменьшит операции по копированию и вставке и количество потенциальных опечаток в цифрах. В терминале введите

```
export KEYID=ID вашего ключа
```



▶ Вот YubiKey, использованный для создания этого учебника.

## Скорая помощь

Создание сертификата отмены крайне важно, если вы планируете использовать его для чего-то важного. Он позволяет отозвать все потерянные/украденные ключи. Как это сделать, вы сможете найти онлайн. Ключ отмены надо хранить отдельно.

## Держите в тайне, спрячьте надежно

[Наказ Гэндальфа Бильбо Бэггину о Кольце Всевластия во «Властелине Колец», — *прим. пер.*] Следующий важный шаг — создание ключа отмены [revocation key]. С ним можно отозвать любой ключ, который был потерян, украден или скомпрометирован. Экспортированный ключ не стоит хранить на локальном устройстве, лучше на флэшке в надежном месте. Чтобы создать файл отмены, сделайте следующее, изменяя местоположения для пользователей USB-флэшки:

```
gpg2 --gen-revoke $KEYID > /path/to/usb/revoke.txt
```

Как и при создании оригинала ключа, процесс задаст вам несколько вопросов. На вопрос о причине отмены можно указать "0". Помимо всего прочего, вас спросят пароль для мастер-ключа и новый пароль для ключа отмены. Следующий шаг — экспортировать "secretkeys [секретные ключи]". Пусть вас не вводит в заблуждение терминология. «Секретные ключи» означает «частный ключ». Это у него название такое! Чтобы экспортировать свой ключ

на USB-брелок, используйте приведенную ниже команду, при необходимости заменив местоположение:

```
gpg2 --armor --export-secret-keys $KEYID > /path/to/usb/master.key
```

После настройки основных ключей надо создать подключи. Здесь мы создаем подключ для подписания. Обратите внимание на параметр `--expert`. Он включает несколько нужных нам опций:

```
gpg2 --expert --edit-key $KEYID
```

Для добавления ключа используйте команду `addkey`. Выберите опцию 4 (RSA Sign only). Опять примите настройки по умолчанию и новый пароль. Обратите внимание, что внизу появилась новая запись, и ее использование помечено "S". Это означает, что ключ служит только для подписания.

Процесс повторяется для создания ключа шифрования. Предполагая, что пользователь всё еще находится в приложении `gpg2`, введите `addkey`. Выберите опцию для RSA Encrypt only [Только шифрование]. Примените настройки по умолчанию, как в предыдущих итерациях. В конечном итоге у вас должен быть новый ключ, отмеченный E (от Encryption).

Последний ключ, который нужно создать — ключ авторизации. Это ключ, который используется для входа в систему, а не для подписания, отмены и т. д. Снова запустите команду `addkey`. Однако на сей раз нам надо выбрать RSA (Set your own capabilities) [задайте свои полномочия]. Причина в том, что нам надо изменить способ использования, чтобы это был ключ авторизации. Если это не появилось, значит, вы не включили режим эксперта!

В ключе авторизации полномочия ключа нужно установить на `Authorize` [Авторизацию]. Разрешенные действия можно включать/отключать клавишами S, E или A. Переключайтесь по необходимости, пока не появится `Authenticate`. Нажмите G, чтобы завершить настройку. Примите настройки по умолчанию. Будем надеяться, что появятся четыре ключа: SC, S, E и A. Сохраните всю работу, введя `save` [сохранить]. Пользователь будет переведен

```
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Secret key is available.

sec rsa2048/BF4E2BD0
   created: 2017-05-01  expires: never       usage: SC
   trust: ultimate    validity: ultimate
[ultimate] (1). Stuart Burns <stuart@newlymintedmedia.com>

gpg> addkey
Please select what kind of key you want:
(3) DSA (sign only)
(4) RSA (sign only)
(5) Elgamal (encrypt only)
(6) RSA (encrypt only)
(7) DSA (set your own capabilities)
(8) RSA (set your own capabilities)
(10) ECC (sign only)
(11) ECC (set your own capabilities)
(12) ECC (encrypt only)
(13) Existing key
Your selection? █
```

▶ Вот детали процесса создания подключей, с расширенными опциями.

## Послужной список YubiKey

YubiKey — это устройство аутентификации оборудования, которое поддерживает разовые пароли, шифрование и аутентификацию с открытым ключом и протокол Universal 2nd Factor (U2F), являющийся международным стандартом для обеспечения беспрепятственной аутентификации (см. <https://fidoalliance.org>). Он позволяет безопасно входить в учетные записи пользователей, предоставляя одноразовые пароли или используя основанную на FIDO пару открытого/частного ключей, созданную устройством.

YubiKey также позволяет хранить статические пароли для использования на сайтах без поддержки

одноразовых паролей. Среди самых крупных сайтов, имеющих такую поддержку, можно назвать Facebook, который использует YubiKey для аутентификационной информации своих работников, и Google, который поддерживает его как для сотрудников, так и для пользователей. Некоторые менеджеры паролей тоже поддерживают YubiKey.

YubiKey реализует основанный на HMAC алгоритм одноразовых паролей (HOTP) и алгоритм одноразовых паролей с временным критерием (TOTP), и определяет себя как клавиатуру, которая предоставляет одноразовые пароли через протокол USB HID. YubiKey NEO и YubiKey 4 включают протоколы

вроде OpenPGP card, использующего 2048-битный алгоритм RSA. Он позволяет подписывать, шифровать и дешифровать сообщения, не выдавая частных ключей. Четвертое поколение YubiKey поддерживает OpenPGP с 4096-битными RSA-ключами и поддержкой PKCS#11 смарт-карт PIV — эта функция позволяет подписывать код образов Docker.

Yubico заменили в YubiKey 4 закрытым кодом все компоненты с открытым кодом. Yubico утверждает, что их код подвергся внутреннему и внешнему пересмотру, и их руководитель разместил обобщение этого шага на [www.yubico.com/2016/05/secure-hardware-vs-open-source](http://www.yubico.com/2016/05/secure-hardware-vs-open-source).

в командную строку. После завершения должен появиться еще один подключ с пометкой “usage A”. Теперь надо сохранить всю эту настройку. Сохраните ее на диске, введя `save`. Пользователь снова окажется в командной строке Linux.

Теперь экспортируйте ключи и подключите частные ключи:

```
gpg2 --armor --export-secret-keys $KEYID > /path/to/usb/master_and_sub.key
```

```
gpg2 --armor --export-secret-subkeys $KEYID > /path/to/usb/sub.key
```

Далее надо экспортировать ключи SSH. В `gpg2` есть инструмент для экспорта частного и открытого ключей SSH из ключей OpenPGP:

```
gpg2 --armor --export $KEYID > /path/to/usb/public_gpg.key
```

```
gpg2 --export-ssh-key $KEYID > /path/to/usb/public_ssh.key
```

В этот момент можно установить YubiKey. Чтобы проверить, распознается ли карта, используйте команду

```
gpg2 --card-edit
```

Если всё прошло нормально, выведется информация, относящаяся к рассматриваемому YubiKey. В нашем контексте мы можем проигнорировать большую часть записей. Единственные пункты, которые я бы предложил вам изменить — это вывод пользователя и администратора. Для этого надо перевести ключ в режим администратора. С этой целью введите `admin` и затем `passwd`. Вывод для пользователя по умолчанию ‘123456’, а для администратора — ‘12345678’. Переделайте их, выбрав какое-нибудь подходящее число и изменив контакты. Не забудьте сохранить информацию, прежде чем выйдете.

## Загрузка на YubiKey

Следующая задача — загрузить ранее созданные ключи авторизации, шифрования и подписания на YubiKey. Это делается командой `keytocard`. Этот шаг придется повторить несколько раз, для каждого ключа, пока не будут загружены ключ подписания, ключ шифрования и ключ аутентификации:

```
gpg2 --edit-key $KEYID
```

В этот момент пользователю надо выбрать соответствующий ключ. Введите `key 1`. Будет выбран ключ 1 в YubiKey. Для перемещения частного ключа на карту ключей введите `keytocard`, чтобы активировать перенос локальных ключей на YubiKey. У вас спросят, куда сохранять ключ на слотах YubiKeys для частных ключей. Выберите `Signature key` [Ключ подписания]. Снова введите `key 1`, и это отменит выбор ключа. Выбранный ключ обозначается знаком ‘\*’ около подключа.

Повторите `keytocard`, используя `key 2`, и в ответ на `keytocard` выберите 2. Снова повторите с `key 3` и выбором 3 (он должен быть единственным доступным). Теперь все ключи скопированы на карту. Но они по-прежнему имеются на компьютере! Лучше всего удалить ключи подписания `root` (здесь позвольте мне еще раз повторить, что удалить эти ключи можно только по завершении резервного копирования, когда у вас будет надежная копия):



➤ А вот новый вид LUKS, настроенного на использование YubiKey.

```
stuart@prodesk:~$ sudo ykpersonalize -2 -ochal-resp -ochal-hmac -ohmac-1t64 -ose
rial-api-visible
Firmware version 4.3.4 Touch level 770 Program sequence 2
Unsupported firmware revision - some features may not be available
Please see
https://developers.yubico.com/yubikey-personalization/doc/Compatibility.html
for more information.

Configuration data to be written to key configuration 2:

fixed: m:
uid: n/a
key: h:
acc_code: h:000000000000
OATH IMF: h:0
ticket_flags: CHAL_RESP
config_flags: CHAL_HMAC|HMAC_LT64
extended_flags: SERIAL_API_VISIBLE

Commit? (y/n) [n]:
```

```
gpg2 --delete-secret-keys $KEYID
```

Программа `gpg2` несколько раз попросит вас подтвердить, что вы на самом деле, действительно хотите удалить мастер-ключ. По завершении проверьте статус ключей YubiKey:

```
gpg2 --list-secret-keys
```

Теперь рядом с ключом безопасности должен быть знак ‘#’. Он дает пользователю знать, что этого ключа не существует на устройстве. А ‘>’ на подключах означает, что ключ находится на карте. Теперь, поскольку мы удалили мастер-ключ, можно импортировать оставшиеся части ключей. На отдельном компьютере, чтобы настроить устройство, запустите команду `gpg2`, и следом `Ctrl+C`. Это делается для создания некоторых важных файлов и папок.

По завершении импортируйте открытые ключи:

```
gpg2 --import < /path/to/usb/public_gpg.key
```

Дважды проверьте доступность этих ключей с помощью `gpg2 --list-secret-keys`. Теперь надо выдать доверие новому ключу, поскольку мы лишили его родителя и, следовательно, цепь доверия разорвалась. Чтобы облегчить себе жизнь, используйте `export KEYID=the sec# key`. Для задания опция доверия `[trust]` отредактируйте ключ:

```
gpg2 --edit-key $KEYID
```

```
trust
```

Когда вам предложат выбрать уровень доверия, выберите 5 (Доверяю в высшей степени — I trust ultimately). Согласитесь на доверие, нажав `Y`. Затем выйдите с помощью `quit`.

Теперь вся настройка для вашего устройства YubiKey выполнена. Последнее, что нам следует сделать — при необходимости включить автоматическую функцию “pass-through [сквозной доступ]” частного ключа. Это делается с помощью двух команд:

```
echo "enable-ssh-support" > ~/.gnupg/gpg-agent.configuration
```

```
echo "export SSH_AUTH_SOCK=~/.gnupg/S.gpg-agent.ssh" >>
~/.bashrc
```

Перезагрузите компьютер, чтобы сервисы обновили конфигурацию. После завершения откройте новое окно терминала и скопандуйте `ssh-add -L`. Если сквозное прохождение SSH работает, должно показаться несколько записей. Сейчас вы можете использовать команду `ssh-copy-id user@remotehost`. Войдите со своим обычным паролем. Ваш открытый ключ скопируется на сервер. На это уйдет буквально секунда.

Теперь, если пользователь попытается зайти по SSH на хост, он введет информацию частного ключа и у него спросят пароль частного ключа. Чтобы узнать об этом подробнее, скопандуйте

```
ssh -v user@remotehost
```

Будем надеяться, что она вам покажет: открытый ключ в настоящий момент используется. Если вы попытаетесь еще раз без YubiKey, ничего не выйдет.

Подводя итоги: YubiKey и другие аппаратные ключи предлагают большой выбор полезных функций. И хотя многие из них выглядят весьма сложными, всё же их определенно стоит использовать хотя бы ради душевного покоя. LXF

➤ Настройка LUKS для использования YubiKey при загрузке.

# Keybase: Азы криптографии

Ник Пирс открывает более дружелюбный к пользователю способ безопасно обмениваться файлами и сообщениями с шифрованием открытого ключа.



## Наш эксперт

Ник Пирс проявляет все больший интерес к конфиденциальности и безопасности, учитывая мир, в котором он оказался. Кто-нибудь знает, как вернуться назад в 2000 г. из оболочки?

**Ш**ифрование с открытым (публичным) ключом уже давно слывет одним из самых надежных способов идентификации и обеспечения конфиденциальности общения и данных, но при этом не самым дружелюбным к пользователю. *Keybase* предназначен изменить эту ситуацию посредством предоставления удобного инструмента для облегчения вашей идентификации другим, чата и совместного использования файлов через безопасный канал, использующий сквозное шифрование.

*Keybase* не подойдет тем, кто хочет скрыть свою реальную личность — как показывает врезка *Спрятаться на виду* (вверху стр. 71), его основной смысл заключается в самопровозглашении, обеспечивающем доказательство, что вы — и правда тот, кем себя объявили. Это делается через привязывание учетных записей ключей к вашему профилю *Keybase* таким образом, чтобы обеспечить солидное подтверждение вашей личности. После этого вы сможете использовать *Keybase* для идентификации с высокой степенью надежности друзей и контактов, а затем общения и совместного использования файлов надежным и безопасным способом.

*Keybase* пока что находится на стадии альфа, но уже создает ощущение готового продукта — его можно использовать в браузере, и есть также клиенты для Linux, Mac и Windows (так что ни у кого нет оправдания не зарегистрироваться). Приложение является самым надежным и удобным способом использования *Keybase*, но пока что могут быть ситуации, когда вы решите использовать браузер — например, чтобы отправить PGP-шифрованные сообщения, не прибегая к командной строке. Мы расскажем об этом на нашем уроке.

## Приступаем

Сначала хорошие новости: раньше *Keybase* работал только по приглашению, но теперь зарегистрироваться может каждый. Перейдите на <https://keybase.io> и нажмите Join. Введите свой адрес электронной почты, а затем имя пользователя *Keybase*, под которым вы будете известны другим. Вам предложат создать пароль — минимум 12 символов, так что сейчас самое время попробовать надежный генератор паролей, например, *KeePass*, если вы еще этого не сделали. После этого нажмите Activate!

Вы попадете на страницу профиля, где увидите кнопку, сообщающую вам о необходимости дальнейших действий; но сначала проверьте почту на наличие сообщения об удостоверении информации. Нажмите на него, чтобы завершить настройку, и ваш адрес электронной почты готов. *Keybase* не раскрывает вашего адреса или номера телефона; всё, что удостоверяет вашу личность, идет из других онлайн-источников, которые вы сообщаете.

Перед этим начните заполнять личную информацию — чем больше вы сообщите, тем лучше. Введите свое полное имя, добавьте краткую биографию и местонахождение — все эти мелкие

подробности помогут другим проверить, кто вы, и отображаются при попадании на <https://keybase.io/username> (где 'username' — это имя, которое вы заявили при регистрации).

## Установка приложения

Прежде чем двинуться дальше, давайте перейдем на использование приложения *Keybase* вместо доступа к нему через браузер. Следующие инструкции предназначены для 64-битных версий Ubuntu и прочих разновидностей Debian; чтобы познакомиться с инструкциями для других дистрибутивов, загляните на <https://keybase.io/download> и нажмите Linux.

Откройте окно Терминала и введите следующее:

```
$ curl -O https://prerelease.keybase.io/keybase_amd64.deb
```

```
$ sudo dpkg -i keybase_amd64.deb
```

Сообщения об ошибках из-за отсутствия 'libappindicator1' проигнорируйте, и введите

```
$ sudo apt-get install -f
```

```
$ run_keybase
```

Вы увидите, как в окне терминала запустится сервис *Keybase* (после этого можно спокойно его закрыть) и затем, впервые, сам *Keybase* — потом вы увидите его значок в строке меню. Откроется web-страница с кратким описанием; по прочтении закройте ее, затем нажмите на значок *Keybase* в строке меню и выберите Show *Keybase*, затем нажмите Log In, чтобы открыть главное окно.

Нажмите на ссылку Log In внизу нового окна, затем введите свое имя пользователя или адрес электронной почты и нажмите Continue. Введите свой пароль и снова нажмите Continue. Вам предложат задать открытое имя для вашего компьютера — вполне подойдет нечто вроде Desktop или Laptop, или, если вы планируете запускать *Keybase* на нескольких платформах, выберите нечто более описательное, например, Ubuntu Desktop.

Нажмите Continue, и вам покажут «бумажный ключ [paper key]», который описан как позволяющий вам «выполнять важные задачи в *Keybase* в будущем» (одна из этих задач позволит вам установить *Keybase* на другом компьютере и привязать к нему свою учетную запись — см. врезку *Скорая помощь* слева). Согласно инструкциям, запишите его в надежное место или скопируйте в надежное приложение заметок, после чего нажмите на Yes, I Wrote This Down [Да, я записал] и на Done [Готово].

## Удостоверьте свою личность

Теперь вы входите в приложение, и вам надо обустроить свою личность. Начните с нажатия на Edit Profile [Редактировать профиль], чтобы добавить свое имя, местоположение и личные данные, если вы этого еще не сделали, затем нажмите Save. О фото пока что не заботьтесь — оно появится из вашей учетной записи Twitter или GitHub, когда вы подтвердите одну из них.

## Скорая помощь

При установке *Keybase* на другой компьютер вам предложат подключиться к существующему устройству или ввести свой бумажный ключ. Следуйте инструкциям, чтобы привязать свою учетную запись к этому компьютеру.

## Спрятаться на виду

Хотя *Keybase* создан для безопасности и конфиденциальности, его нельзя назвать секретным. Ваш профиль максимально открыт и доступен, и на то есть важная причина. В основном клиент *Keybase* — будь то приложение или ваш браузер — создан именно для того, чтобы не доверять никакому серверу, поэтому он должен подтвердить личность человека, не полагаясь на сервер для доказательства.

Это означает, что если сервер говорит вам, что является контактом, то клиент удостоверяет каждое из доказательств его личности независимо, и далее исключительно от вас зависит просмотр

этих доказательств, с целью убедиться, что это лицо — действительно тот, за кого себя выдает (и кого вы ищете). Побочным преимуществом подобного подхода является то, что открытая часть криптографической пары ключей *Keybase* не требует никаких усилий или неофициального канала для передачи, упрощая процесс общения с другими людьми.

Такой трехшаговый подход является основой всех действий *Keybase*, от идентификации и общения с людьми для чата и совместного использования файлов: он повторяется каждый раз, когда вы изучаете доказательство, общаетесь с кем-то

или делитесь с ними файлами, и если что-то изменится, вы узнаете об этом первым.

Всё это происходит невероятно быстро, и когда вы приписываетесь [follow] к кому-то, вы по сути предоставляете «подпись», подтверждающую личность этого человека, которая подписана вашим частным ключом. Это не просто помогает укрепить заявку вашего контакта на подлинность его личности, но и обеспечивает вас подписанным подтверждением, которое может быть передано на другой компьютер при установке клиента *Keybase* на ваш ноутбук.

Следующий шаг — создать ряд доказательств, которые подтвердят, что вы — это вы. Сюда входят ссылки на разные учетные записи: Twitter, Reddit, Facebook, GitHub, Hacker News и один или более сайтов. Для начала нажмите на один из них — возьмем в качестве примера Twitter; вам предложат ввести свое имя пользователя, затем скопировать и вставить твит в свою учетную запись. Затем *Keybase* поищет его в вашей временной шкале, и — когда найдет — соответственно подтвердит вашу личность.

Попутно отметим, что когда вы подключите свою учетную запись Twitter, вы будете получать уведомления по электронной почте каждый раз, когда кто-то, кого вы читаете в Twitter, присоединится к *Keybase* и подключит свою учетную запись — если это вам досаждают, перейдите в Settings > Notifications [Настройки > Уведомления], где вы сможете отключить это и другие уведомления.

Другие сервисы работают аналогичным образом; в случае Facebook просто нажмите на ссылку Make a Facebook Post, введя свое имя пользователя (если вы его не знаете, перейдите в свой профиль Facebook и загляните в адресную строку на [www.facebook.com/ваше\\_имя.123](http://www.facebook.com/ваше_имя.123)). Обязательно сделайте его Public.

Желая подтвердить свой сайт, вы можете или разместить в Сети текстовый файл, или поместить подтверждение *Keybase* в своих записях DNS. Если вы выбираете последний способ, зайдите к своему провайдеру доменных имен и загляните в раздел расширенных настроек своего DNS домена для пункта TXT, куда и введите подтверждение. Не паникуйте, если у вас появится ошибка 'unreachable' — продолжайте проверять, и в течение 24 часов вы должны получить подтверждение.

Перезагрузите свой профиль, когда вам предложат это сделать, и вы увидите, что ваша личность подтверждена. Теперь вы можете просматривать свое подтверждение (или отозвать его при необходимости), просто нажав на галочку рядом с данной личностью. Чем больше вы создадите подтверждений, тем солиднее ваша личность для других — и все, кто вас читают, получат уведомление по электронной почте для каждого подтверждения, которое вы предоставляете по мере их добавления; вы, в свою очередь, получаете уведомления по мере того, как те, кого вы читаете, добавляют подтверждения в свою учетную запись.

Для максимальной безопасности обязательно нажмите на ссылку Add a PGP Key [Добавить ключ PGP] — вы можете импортировать существующий ключ, например, тот, который вы используете для электронной почты, или позволить *Keybase* создать его для вас. При создании с нуля включите все адреса электронной почты, которые планируете использовать для передачи зашифрованных PGP сообщений электронной почты.

## Создание контактов

Теперь можете приступать к поиску друзей и контактов через *Keybase*. Вы можете разослать другим приглашения, или они могут первыми найти вас и появиться в вашем списке Followers. Если они нашли вас первыми, нажмите на их имя и просмотрите их подтверждения, затем нажмите Follow, чтобы тоже их читать. Вы также можете просматривать их список читателей (и тех, кого они читают), чтобы определить их контакты; или можете воспользоваться значком Search.

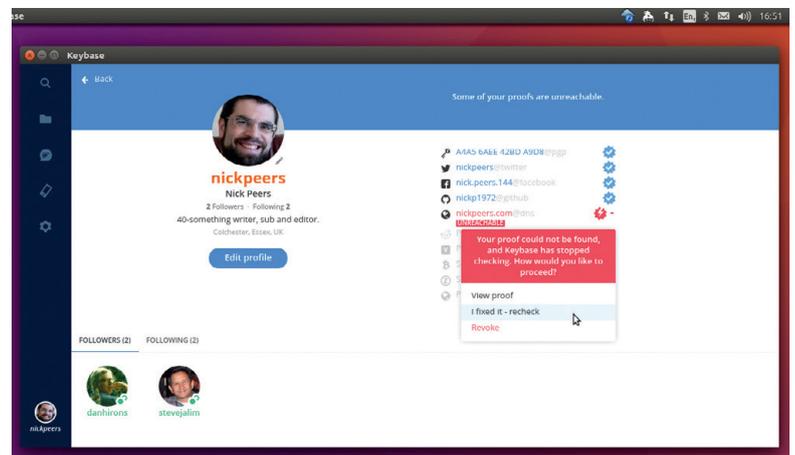
Кстати, читать кого-то — это то же самое, что и подтвердить их личность, добавляя доказательств к их заявлению, что они именно те, кем себя объявляют.

Выбор Search позволяет искать контакты с помощью не только их имени *Keybase*, но также в Twitter, Facebook, GitHub, Reddit или Hacker News. Если у них есть профиль *Keybase*, он выделится; если нет, вы можете начать беседу или настроить папку общего доступа с ними и затем отправить им ссылку с приглашением. Как только они зарегистрируются в *Keybase* и подключат эту учетную запись, беседа и папка станут им видны.

Вы также увидите опции открыть папки или начать чат — через минуту мы расскажем подробнее. Инструмент Search также удобен, если вы хотите настроить групповой чат или создать папку группового использования — вы просто ищете тех, с кем хотите связаться, затем щелкаете по ним, чтобы добавить в список имен. Далее общаться с ними в чате будет просто — пошаговое руководство расскажет вам всё, что вы можете сделать.

### Скорая помощь

Если вы блокируете несколько бесед или забыли, которую беседу заблокировали, введите `keybase chat hide -i` в окне Терминала, чтобы показать все заблокированные беседы.



➤ Нажмите на кнопку с галочкой рядом с подключенной учетной записью, чтобы удостовериться в ней онлайн или — при необходимости — отозвать.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)

## Скорая помощь

Если вы когда-либо решите, что *Keybase* не для вас, перейдите в *Settings* [Настройки] в приложении *Keybase* и выберите *Delete Me* [Удалить Меня]. Нажмите *Delete My Account Forever* [Навсегда Удалить Мою учетную запись] для удаления и учетной записи, и имени пользователя — чтобы ни вы, ни кто-либо другой не использовали их в будущем.

## Обобществление файлов

Если вы хотите поделиться файлами с другими, *Keybase* предлагает криптографически надежное монтирование файлов. Предлагается две опции: частная, для монтирования только с избранными людьми, и публичная, папка с распределенным доступом для всех.

Вы можете настроить частные папки с распределенным доступом одним из двух способов: через инструмент *Search*, для чего вам надо выделить свой контакт и нажать *Open Private Folder*. Создастся новая папка в `./keybase/private` с помощью следующего синтаксиса: `yourname.theirname`. Перетащите в нее файлы, и они будут зашифрованы и переданы в соответствующую папку, которая создается на компьютере вашего контакта, где они дешифруются. Всё это делается невидимо, за сценой.

Вы также можете получить доступ к вашим частным папкам, нажав на значок с папками в главном окне или заново открыв окно состояния *Keybase* из панели меню — отсюда вы можете создавать с нуля новую частную папку, вручную вводя имя контакта, с которым хотите поделиться.

Опция публичной папки весьма интересна — она очевидно не предназначена для обеспечения безопасных средств передачи файлов между людьми, но зато дает пользователям *Keybase* гарантию, что файлы, которыми вы решили поделиться, никоим образом не были подделаны — здесь нет серверной стороны и возможности взлома методом «человек-посредник».

Публичные папки не синхронизируются автоматически с компьютерами других; вместо этого файлы передаются потоком по запросу. Вы можете получить доступ к вашей папке совместного пользования через значок *Folders*, перейдя на вкладку *Public/*. Нажмите на свое имя, чтобы открыть собственную публичную папку *Public* (она расположена в `./keybase/public/yourname`). Все скопированные туда файлы становятся доступными для всех, кто решит подключиться к вашей публичной папке.

Доступ к публичным папкам других можно получить тем же способом, которым вы подключаетесь к частным папкам — через

вкладку *Search* (нажмите *Open Public Folder*) или открыв значок строки состояния, перейдя на вкладку *Public/* и введя их имя. При выборе последней опции вы сможете также создавать групповые ресурсы совместного использования, с помощью синтаксиса `yourname.theirname`.

## Обмен сообщениями PGP

Добавив ключ PGP, вы сможете отправлять PGP-зашифрованные сообщения через свой браузер, через командную строку (`keybase pgp encrypt`) или поддерживаемое приложение — например, такое, как *GPG*.

Чтобы сделать это через браузер, войдите в *Keybase* на [www.keybase.io](http://www.keybase.io) и затем выберите человека из своего списка *Following* [Читаемые] или *Followers* [Читатели] или используйте инструмент поиска, чтобы найти их в *Keybase*. Если есть опция *PGP Encrypt*, нажмите на нее.

Отсюда просто введите сообщение, которое хотите отправить, затем нажмите на кнопку *Encrypt*, и оно будет отправлено в зашифрованном виде. Вы также увидите кнопки для дешифрации, подписания и удостоверения сообщений с помощью PGP, но не все эти опции работают в вашем браузере, если только вы не решили разместить свой частный ключ в зашифрованном магазине *Keybase* при его создании или импорте. Если вы этого не сделали, вам придется дешифровать и подписывать сообщения из командной строки:

```
$ keybase pgp decrypt -m "message text"
```

```
$ keybase pgp sign -m "message text"
```

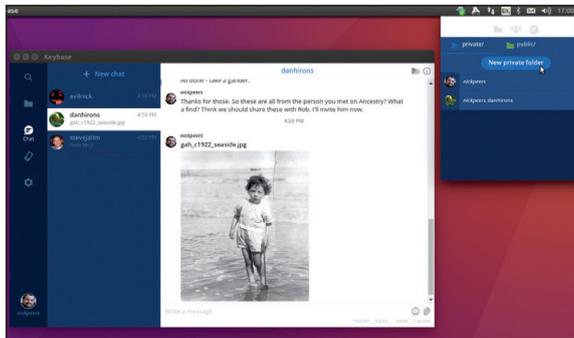
Столь же легко импортировать ключи *Keybase* в *GPG* (или любые другие приложения PGP) для получения разрешения читать и отправлять сообщения оттуда. Если вы используете *GPG*, это можно сделать с помощью приложения *Keybase*:

```
$ keybase follow username
```

```
$ keybase pgp pull username
```

## И наконец...

Хотя *Keybase* относительно прост в использовании, может настать момент, когда вам захочется просто нажать на кнопку перезагрузки и снова начать с нуля, или вдруг вы потеряете доступ ко всем своим частным ключам. Чтобы бы вы ни делали, не удаляйте свою учетную запись; вместо этого перейдите на <https://keybase.io/username>, где 'username' — это, естественно, ваше имя пользователя *Keybase*, и нажмите на кнопку *Settings* [Настройки]. Выберите *Reset Your Keys & Start From Scratch* [Перезапустить Ключи и Начать с Нуля], введите пароль для своей учетной записи и нажмите *Reset Account* [Перезапустить учетную запись]. Это удалит ваши ключи — вместе со всеми вашими доказательствами и устройствами — позволяя вам начать всё с самого начала. LXF



➤ Можно делиться файлами частным порядком или публично, с отдельными людьми или группами. Альтернатива — делиться файлами внутри зашифрованного чата.

## Работа из командной строки

Презираете GUI? Хорошая новость: у *Keybase* имеется свой полнофункциональный инструмент командной строки. Вот вполне очевидная стартовая точка:

```
$ keybase help
```

Если вы — новичок в *Keybase*, начните с `keybase signup`; либо войдите в готовую учетную запись по `keybase login`. Используйте команду `prove` для добавления доказательств — например, `keybase prove twitter`. Команда `search` поможет вам найти других по совпадениям параметров, а команда `id` (`keybase`

`id` nick) позволяет удостоверить определенную учетную запись.

Команда `follow` вполне понятна и особенно ценна, поскольку подключение к людям через командную строку проще, если вы их читаете. Команда `encrypt` обеспечивает серию инструментов для отправки зашифрованных файлов и сообщений. Следующие примеры с сайта *Keybase* ([https://keybase.io/docs/command\\_line](https://keybase.io/docs/command_line)) показывают, как это работает:

```
$ keybase encrypt max -m "это секрет"
```

```
$ echo "это секрет" | keybase encrypt max
```

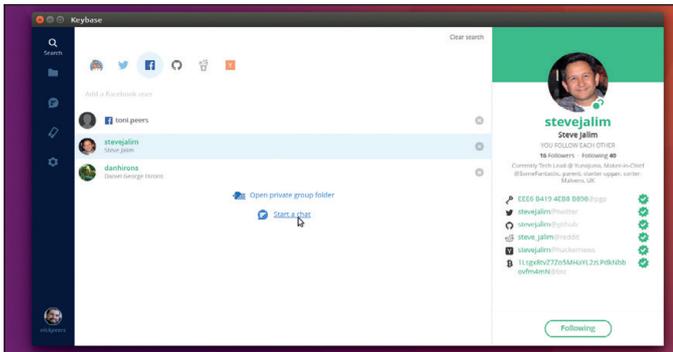
```
$ keybase encrypt max -i secret.txt
```

```
$ keybase encrypt max -i secret.mp3 -b -o secret.mp3.encrypted
```

Среди других команд, которые вам пригодятся — `decrypt` (для доступа к зашифрованному контенту, отправленному вам) и `pgp`, которая позволяет зашифровать и расшифровать с помощью вашего ключа PGP в дополнение к подписи и удостоверению сообщений.

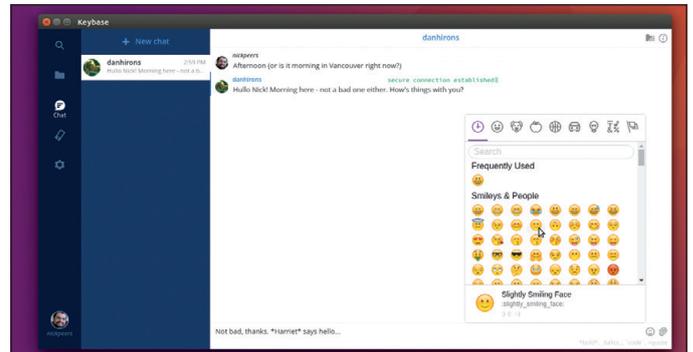
➤ Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

## Давайте поболтаем



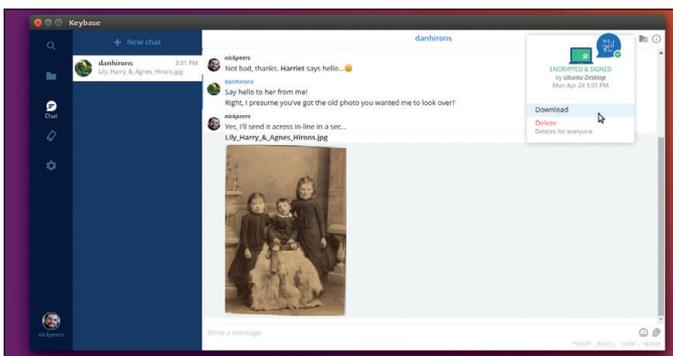
### 1 Выберите, с кем общаться в чате

Нажмите на значок Search в верхнем левом углу и ищите людей по их имени в Keybase или по иному способу идентификации. Если обнаружится несколько совпадений, прокрутите список в поисках своего контакта и затем нажмите на его имя. Проверьте его личность, затем повторите, чтобы добавить другие контакты. Чтобы удалить контакт, нажмите X рядом с ним. Закончив, нажмите Start a Chat [Начать беседу].



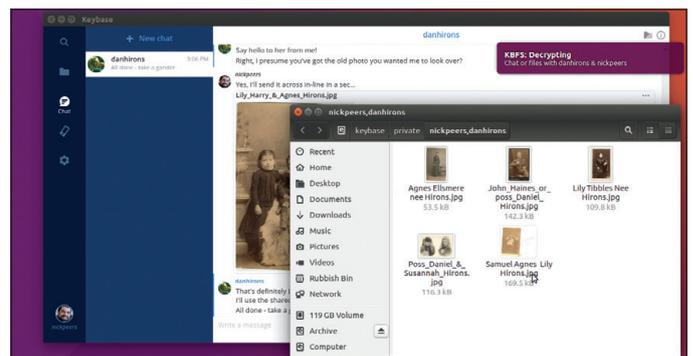
### 2 Начните чат

Keybase работает, как и большинство других инструментов чата — введите сообщение и нажмите Enter для отправки. Поддерживается ограниченное форматирование с помощью хорошо известных текстовых кодов — например, заключение жирного шрифта в значки \* (например, \*это жирный шрифт\*). Нажмите на кнопку со смайликом, чтобы добавить какой-нибудь эмодзи. Когда сообщения отправлены, люди получают уведомление через Keybase и Ubuntu.



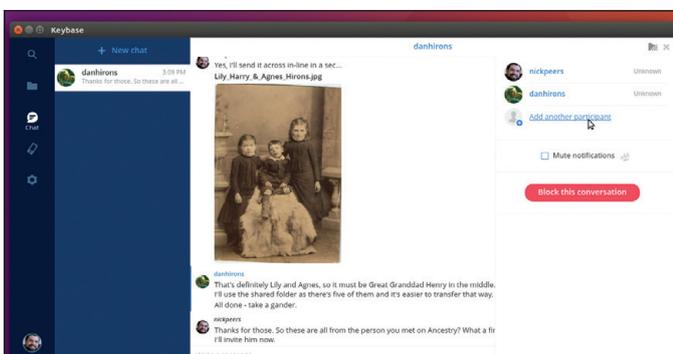
### 3 Поделитесь файлами

Вы можете делиться файлами через чат, нажав на значок со скрепкой — выберите файл, затем нажмите Send. Файлы будут появляться как часть беседы (показаны эскизы изображений). Прокрутите мышью по файлу и нажмите на ... справа от него, для удостоверения, что он подписан, и выберите, скачать себе копию файла или удалить его для всех.



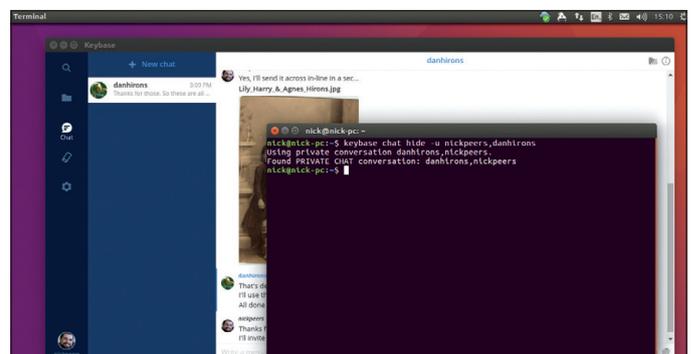
### 4 Откройте частную папку

Вы также можете делиться файлами с партнерами по чату, создавая или открывая папку частного доступа — нажмите на кнопку с папкой наверху экрана, чтобы открыть папку на вашем рабочем столе, затем скопируйте туда любые файлы. Сообщите вашим партнерам по чату, что вы сделали, и они смогут получить файлы — и добавить собственные — точно таким же способом.



### 5 Другие опции беседы

Нажмите на кнопку  в верхней части окна чата, чтобы просмотреть список людей в беседе — нажмите Add Another Participant [Добавить Участника] для добавления людей к чату, и при необходимости отметьте Mute Notifications [Отключить звуковые уведомления] или даже Block this Conversation [Заблокировать эту беседу]. Это не блокирует людей; это просто скрывает беседу из окна чата и отключает все звуковые уведомления.



### 6 Отмена блокировки

Ваши партнеры по чату не знают, что вы их заблокировали, и вы не можете начать новый чат с этим человеком или группой. Если вы передумали, откройте окно Терминала и введите

```
$ keybase chat hide -u yourname,theirname
```

Если вы не уверены в том, как называлась беседа, загляните во врезку *Скорая помощь* на стр. 71.

# APT: Управляем пакетами

Попробуйте на зуб курс Администрирования системы от Linux Foundation вместе с **Джонни Бидвеллом** — он бегло обрисует стиль управления Linux.



**Наш эксперт**

**Джонни Бидвелл** — кочующий хакер Linux, бывший эксперт нумерологии и почитатель Лавкрафта. Его попытки установить Kтулху с помощью apt-get install cthulhu на разные части инфраструктуры Башен LXF пока что закончились неудачей, однако он не сдаётся...

**К**урс *Основы Системного Администрирования в Linux* от Linux Foundation, LFS201, весьма информативен. Он состоит из 42 частей с краткими проверочными тестами и более длинными лабораторными работами. На нашем уроке мы рассмотрим тему систем управления пакетами (часть 27) перед тем, как переходить к курсу по Debian *APT* (часть 32).

Необходимо разобраться с тем, как устанавливается, обновляется и удаляется ПО в системах Linux. Системы управления пакетами предлагают понятный способ достижения этой цели и не позволяют системам с течением времени скатиться в хаос, застой или отказ. Более того, они обеспечивают методику проверки программ в системе на предмет повреждения, злоумышленного или случайного. Основная функция распространителей Linux — разработки и поддержка пакетов и обеспечение должной поддержки зависимостей со временем.

Системы управления пакетами обеспечивают инструменты, позволяющие системным администраторам автоматизировать установку, обновление, настройку и удаление программных пакетов известным, предсказуемым и уместным способом. Эти системы позволяют добиться следующего:

- » собрать и заархивировать соответствующие программные файлы в один пакет (архив), который может требовать сперва установить один или более других пакетов;
- » способствуют простой установке и удалению программ;
- » могут подтвердить целостность файла через внутреннюю базу данных;
- » могут удостоверить происхождение пакетов;
- » упрощают обновления;

- » группируют пакеты по логическим функциям; и
- » управляют зависимостями между пакетами.

Определённый пакет может содержать исполняемые файлы, файлы данных, документацию, скрипты установки и файлы настройки. Также включаются атрибуты метаданных, такие как номера версий, проверочные суммы, информация о производителе, зависимости, описания и т. д. После установки вся эта информация хранится локально во внутренней базе данных, куда можно отправить запрос о состоянии версии или информации об обновлении.

Системы управления программными пакетами считаются одним из главных улучшений, внесенных Linux в корпоративную ИТ-среду. Отслеживая файлы и метаданные автоматическим, предсказуемым и надежным способом, системные администраторы могут использовать системы управления пакетами для масштабирования своих процессов установки на тысячи систем без необходимости ручной работы в каждой отдельной системе. Функции включают:

- » автоматизацию — нет нужды в ручной установке и обновлении;
- » масштабируемость — установка пакетов на одной или 10 000 систем;
- » повторяемость и надежность; и
- » безопасность и аудит;

Пакеты предлагаются разных типов:

- » Двоичные пакеты содержат файлы, готовые к использованию, включая исполняемые файлы и библиотеки. Они зависят от архитектуры и компилируются для каждого типа компьютеров.
- » Исходные пакеты используются для создания двоичных пакетов. Следует всегда иметь возможность пересобрать двоичный пакет (например, в системах на основе RPM — с помощью `rpmbuild --rebuild`) из исходного. Один исходный пакет пригоден для многих архитектур.
- » Независимые от архитектуры пакеты содержат файлы и скрипты, которые работают в интерпретаторах скриптов, а также документацию и файлы настройки.
- » Мета-пакеты являются группами соответствующих пакетов, которые собирают всё необходимое для установки относительно крупной подсистемы, например, среды рабочего стола или офисного пакета.

Именно с двоичными пакетами системным администраторам приходится работать большую часть времени.

## Удвоим удовольствие

64-битные системы умеют запускать и 32-битные программы, поэтому можно установить на определенную программу два двоичных пакета, возможно, один с `x86_64` или `amd64` в названии, и другой — с `i386` или `i686` в названии. Исходные пакеты помогут

```
Activities
johnnie@debian:~$ apt-get help
apt 1.8.9.8.4 for amd64 compiled on Dec 11 2016 09:48:19
Usage: apt-get [options] command
       apt-get [options] install|remove pkg1 [pkg2 ...]
       apt-get [options] source pkg1 [pkg2 ...]

apt-get is a simple command line interface for downloading and
installing packages. The most frequently used commands are update
and install.

Commands:
  update - Retrieve new lists of packages
  upgrade - Perform an upgrade
  install - Install new packages (pkg is lib64 not lib64.deb)
  remove - Remove packages
  autoremove - Remove automatically all unused packages
  purge - Remove packages and config files
  source - Download source archives
  build-dep - Configure build-dependencies for source packages
  dist-upgrade - Distribution upgrade, see apt-get(8)
  dselect-upgrade - Follow dselect selections
  clean - Erase downloaded archive files
  autoclean - Erase old downloaded archive files
  check - Verify that there are no broken dependencies
  changelog - Download and display the changelog for the given package
  download - Download the binary package into the current directory

Options:
  -h This help text.
  -q Loggable output - no progress indicator

johnnie@debian:~$
```

» Как и для всех команд Linux, справка для apt-get находится всего в нескольких символах рядом.



Подпишитесь сейчас и сэкономьте **10%**

\* Действует до 30.12.2017

## Учитесь с Linux Foundation

Подпишитесь на любой обучающий курс Linux Foundation сегодня и получите 10-процентную скидку!\* Эксперты Linux Foundation разработали курсы по всем аспектам Linux, от классических курсов системных администраторов до новейших технологий, таких как OpenStack и Open Daylight.

Если вам понравился этот небольшой экскурс и вы хотите узнать больше, мы взяли эту часть из очень популярного курса *LFS201 Основы*

*Системного Администрирования*, который идет с опциональной программой сертификации системных администраторов от Linux Foundation Certified Sysadmin. Курс исключительно онлайн, на основе самоподготовки, и содержит более 80 лабораторных работ для отработки ваших новых навыков.

Чтобы получить 10-процентную скидку на *любой* курс, следуйте инструкциям: Перейдите на <https://training.linuxfoundation.org/linux-courses> и про-

смотрите полный список курсов.

Выберите интересующий вас курс, например, тот самый популярный курс *LFS201 Основы Системного Администрирования*.

Нажмите Register и создайте учетную запись, введя код купона\* **TECH10** для получения 10-процентной скидки. Вот и всё!

в отслеживании изменений и исходного кода с двоичными пакетами. Обычно они не устанавливаются в систему по умолчанию, но их можно скачать у производителя.

У системы пакетов есть два уровня:

» Низкоуровневая утилита — она просто устанавливает или удаляет один пакет или список пакетов, каждый из которых имеет индивидуальное и определенное название. Зависимости не обрабатываются в полном объеме, о них только предупреждают. Поэтому, если сначала нужно установить другой пакет, установки не получится, а если пакет нужен другому пакету, то не получится его удалить. Утилиты *rpm* и *dpkg* играют эту роль для использующих их систем пакетов.

» Высокоуровневая утилита — она решает проблемы зависимостей. Если нужно установить другой пакет или группу пакетов перед установкой программы, это будет проделано. Если удаление пакета влияет на другой установленный пакет, администратору будет предложено на выбор прервать или удалить задействованную программу.

Утилиты *yum*, *dnf* и *zypper* заботятся о разрешении зависимостей на системах RPM, а *apt-get*, *apt-cache* и другие утилиты заботятся об этом в системах *dpkg*.

## Этот жуткий APT

Для использования в системах на Debian набор программ *APT* (advanced packaging tool) обеспечивает более высокий уровень разумного сервиса для использования лежащей в основе программы *dpkg* и играет ту же роль, что и *yum* в системах на Red Hat. Основные утилиты — *apt-get* и *apt-cache*. *APT* может разрешать зависимости при установке, обновлении и удалении пакетов и имеет доступ к внешним программным репозиториям, синхронизируется с ними и находит и устанавливает программы по необходимости.

Запросы делаются с помощью утилиты *apt-cache*. Чтобы найти в репозитории пакет с именем *apache2*, используйте

```
$ apt-cache search apache2
```

Чтобы отобразить базовую информацию о пакете *apache2*, запустите

```
$ apt-cache show apache2
```

Чтобы отобразить более подробную информацию о пакете *apache2*, введите

```
$ apt-cache showpkg apache2
```

Для получения списка всех зависимых пакетов для *apache2*:

```
$ apt-cache depends apache2
```

Поиск по репозиторию файла под названием *apache2.conf*:

```
$ apt-file search apache2.conf
```

Список всех файлов в пакете *apache2*:

```
$ apt-file list apache2
```

Программа *apt-get* — это рабочая лошадка установки, удаления и обновления пакетов.

Индексы имеющихся пакетов берутся из местоположений, указанных в */etc/apt/sources.list*. Для синхронизации файлов индексов пакетов с исходниками их репозиториях используйте команду

```
$ sudo apt-get update
```

Если вы хотите установить новые пакеты или обновить уже установленные, введите

```
$ sudo apt-get install [package]
```

Удалите пакет из системы, не удаляя его файлов настройки:

```
$ sudo apt-get remove [package]
```

Удалите пакет из системы вместе с его файлами настройки:

```
$ sudo apt-get --purge remove [package]
```

Примените все имеющиеся обновления к уже установленным пакетам:

```
$ sudo apt-get upgrade
```

Выполните умное обновление с более тщательным разрешением зависимостей и удалите устаревшие пакеты и установите новые зависимости. Это не приведет, как часто неправильно считают, к обновлению до новой версии дистрибутива Linux:

```
$ sudo apt-get dist-upgrade
```

Обратите внимание, что вы должны провести обновление до апгрейда, в отличие от ситуации с *yum* (или *dnf*), когда аргумент обновления выполняет оба шага, обновляя репозитории и затем выполняя апгрейд пакетов. Это может запутать привыкших к *yum* пользователей на системах на базе Debian.

Чтобы избавиться от пакетов, которые больше не нужны, например, от старых версий ядра Linux:

```
$ sudo apt-get autoremove
```

Очистите файлы кэша и все архивные файлы пакетов, которые были установлены:

```
$ sudo apt-get clean
```

Это сэкономит вам много места.

В описанном курсе есть много чего еще (около 40 частей), и всем, кто желает привести в порядок свои навыки с перспективной работы системным администратором, мы настоятельно рекомендуем пройти его. **LXF**

```
Activities
joni@Debian:~$
joni@Debian:~$ apt-cache rdepends apache2
apache2
Reverse Depends:
  loundcloud-config-apache
  loundcloud-deps-php7.0
  loundcloud-deps-php5
  lzoneminder
  lzabbix-frontend-php
  lxapian-omega
  lwordpress
  ltrae
  lstacks-web
  lrouting-www
  lroundcube-core
  lr14-apache2
  lrauid-8ui
  lpython-certbot-apache
  lpuppet-master-passenger
  lpp4nagios-web-config-nagios3
  lpp4nagios-web-config-icinga
  lpiparts-master
  loar-web-status
  loar-restful-api
  loar-restful-api
  lmediawiki
  ledgersmb
  ljava-server
  ljscmunicator-web-phone
  licinga2-classicui
joni@Debian:~$
```

» Web-сервер Apache имеет много обратных зависимостей. Apt-get перечислит все, а не только установленные.

ЧАСТЬ 8

# IoT: Дом, который послушен нам

Максим Черепанов заделывается Большим Братом и учиняет в своем доме видеонаблюдение. Результатом оказалось уловление собаки за совершением противоправных действий...



Наш эксперт

Максим Черепанов — заместитель директора в небольшой коммерческой фирме. По образованию железнодорожник-управленец. Linux занимается с 2008 г. Чем дольше это делает, тем дальше от Windows. За эти годы разучился чистить реестр и забыл слово «антивирус».



Прошло совсем немного времени (полгода), и мы имеем достаточно продуманную и понятную схему нашего Послушного Дома, изображенную на рис. 1.

Разумеется, не все службы управления привязаны так, чтобы составлять единое целое. Причина проста: ПД — это живой учебный проект, разрабатывается как наглядное пособие для изучения систем управления и контроля. И в конфигурации, существующей в данный момент, для того, чтобы реально управлять нагрузкой на исполнительном контроллере с помощью, допустим, электронной почты, надо скорректировать файл `procmail` (так же, как и отсылку контрольных данных с помощью SMTP). Я не буду при внесении принципиальных изменений в конфигурацию ПД изменять конфигурационные файлы тех служб, которые были рассмотрены ранее. Это всё вы сможете сделать сами по приведенным шаблонам, основываясь на природной смекалке и опыте программирования ПД.

А сегодня мы займемся другой, не менее интересной темой.

## «Шеф, а я вас вижу!»

Ну конечно, какой же Послушный (или Умный) Дом без видеонаблюдения? Этот сервис в настоящее время предоставляется всеми и повсеместно, устройства (камеры) для его реализации

из дорогостоящих превратились в доступные. А в смартфоны их встраивают по две или даже три штуки.

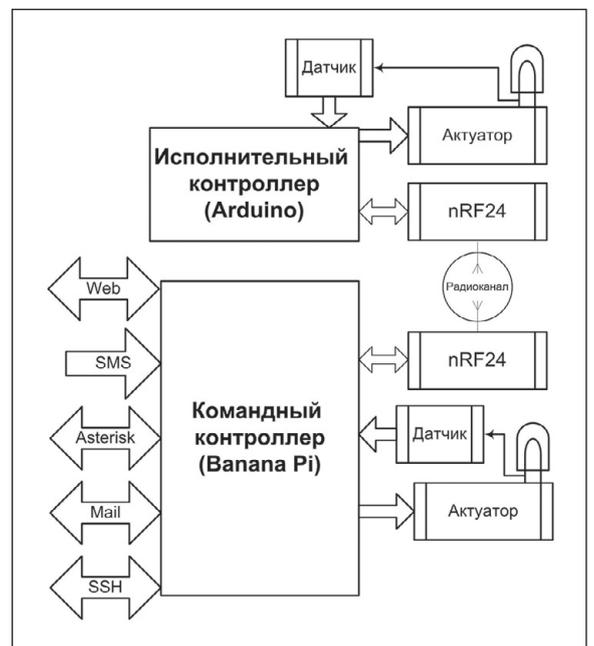
Не исключение и BananaPi, на основе которого мы строим ПД. Существует целый ряд камер, которые основаны на чипе OV5640. Они совместимы с другими одноплатными компьютерами, подключаются к разъему CON1 (40 Pin FPC to the Camera Sensor Interface) ленточным кабелем. Характеристики таких камер вы без труда найдете в Интернете; из них надо отметить матрицу в 5 Мп и автофокус. Наиболее популярны 2 варианта: один производится фирмой LeMaker (производитель Banana, рис. 2), а другой — без наименования, продается на AliExpress (рис. 3).

Для них существует встроенный в ОС модуль ядра, который включается в терминале так:

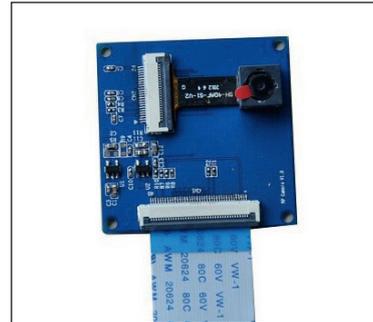
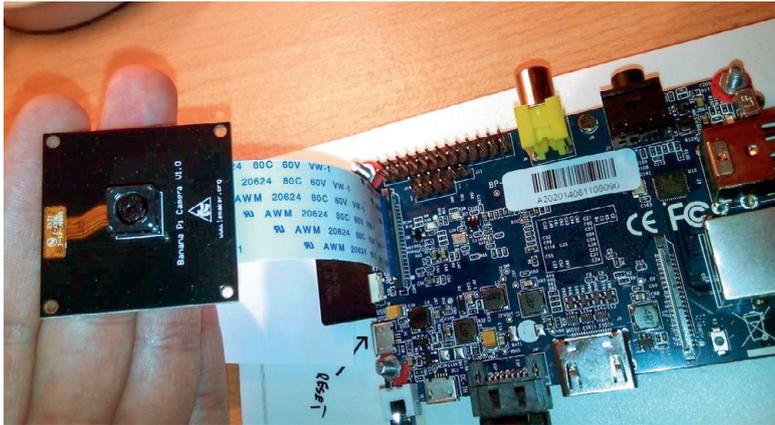
```
$ sudo modprobe ov5640
```

После этой манипуляции камера должна увидаться как `/dev/video0`.

Но стоят эти камеры недешево (от 700 рублей без доставки), приклеплены к контроллеру хрупким ленточным кабелем, что ограничивает нас в размещении камеры относительно командного



» Рис. 1. Блок-схема ПД.



► Рис. 3. Безымянная камера с Али.

► Рис. 2. Камера от LeMaker.

контроллера. Да и зачем нам 5 мегапикселей для наблюдения за домом?

Поэтому «мы пойдем другим путем!»! До того момента, когда видеоконференции были не так популярны, как сейчас, многие ноутбуки выпускались без встроенных камер. Поэтому видеоконференции с USB-интерфейсом в свое время разлетались, как горячие пирожки у студенческого общежития. Сейчас спрос резко упал (смартфоны и ноутбуки с интегрированными камерами сделали свое дело), и почти у каждого в доме есть старая рабочая, но уже не нужная USB-камера. Если ее у вас нет, почти наверняка она есть у соседей или друзей.

Такая оказалась и у меня. В свое время она мне помогла работать со *Skype* в Linux, потому что не все камеры определялись как `/dev/video*` — иногда надо было поплясать с бубном. Называется она L-PRO 1181, имеет аж 0,3 Мп и максимальное разрешение картинки 640×480 пикселей (рис. 4). В век фильмов 4K такое должно казаться смешным, но для ПД это именно то, что надо. Слишком большое разрешение заставило бы программно уменьшать разрешение или потребовало бы большого объема дискового пространства.

Данная камера подключается по USB для видеопотока и имеет стандартный 3,5-мм разъем «папа» для передачи звука. Микрофон, конечно, не HiFi, но голос различается хорошо, а в данном случае это вообще не важно. Преимуществом является также крепление в виде широкой прозрачной прищепки, которая может служить и подставкой.

Подключается камера просто в разъем USB напрямую или через хаб, и после этого сразу появляется `/dev/video0` — ничего придумывать не надо.

Проверьте наличие этого блочного устройства, и далее приступим к установке ПО.

## Знакомьтесь — motion

Утилита *motion* доступна для всех известных мне дистрибутивов Linux. Название говорит само за себя; кроме передачи движения, *motion* может определять движение в определенной области кадра, отправлять сообщение при обнаружении активности в этой области, делать снимки, транслировать изображение по сети. *Motion* исповедует в настройке Linux-way, конфигурационный файл хорошо комментирован, и если у вас нет проблем с английским (достаточно даже начального уровня), то понять смысл опций для вас не составит труда.

Итак:

```
$ sudo apt-get install motion
```

Вам предложено будет вместе с основной утилитой установить несколько зависимостей. Соглашайтесь. После скачивания

и установки система попытается запустить *motion*, но вы увидите сообщение:

```
[warn] Not starting motion daemon, disabled via /etc/default/motion ... (warning).
```

То есть режим демона отключен. Включаем:

```
nano /etc/default/motion
```

Меняем в конце единственной строки “no” на “yes”.

Но это еще не всё, теперь тот же режим надо включить в другом файле:

```
nano /etc/motion/motion.conf
```

Ищем строки

```
# Start in daemon (background) mode and release terminal
(default: off)
daemon off
```

Меняем снова на “on” и запускаем *motion* с конфигурационным файлом по умолчанию:

```
service motion start
```

Проверьте, стартовал ли сервис:

```
ps aux | grep motion
motion 19971 2.0 0.9 51316 8380 ? S 13:56 0:00 /usr/bin/motion
```

Всё в порядке. Есть желание посмотреть на сырые данные, которые камера отправляет на микро-HTTP-сервер? Тогда наберите из сессии SSH:

```
telnet localhost 8081
```

»



► Рис. 4. USB-камера L-PRO 1181.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)

8081 — порт HTTP-микросервера по умолчанию, изначально доступен только локально (только с localhost), это мы потом поправим. Будьте готовы, что в окно терминала повалятся кракозябры, и корректно закрыть терминал не получится. Откройте другой терминал и снимите процесс, который вы обнаружили на предыдущем шаге:

```
# kill 19971
```

Это было маленькое отступление, почти шутка. Зато вы теперь знаете, что с помощью telnet можно проверять активность сервисов на внешних портах localhost'a.

## Настроим motion «под себя»

Итак, где всё это веселое хозяйство, вы уже знаете: в `/etc/motion/motion.conf`. Параметров много, но не очень. Файл являет собой шеддер конфигурационного файла — комментарии исчерпывающие, краткие и понятные.

Пройдемся по основным параметрам, чтобы интегрировать motion в наш ПД. Они будут не в порядке следования в файле, а в порядке важности (по моему мнению), и расположены в разных секциях:

```
# Videodevice to be used for capturing (default /dev/video0)
```

```
# for FreeBSD default is /dev/bktr0
```

```
videodevice /dev/video0
```

— видеоустройство. Благо, в 99% случаев это оно и есть, так что оставляем как есть.

```
# The mini-http server listens to this port for requests (default: 0 = disabled)
```

```
webcam_port 8081
```

— порт встроенного web-сервера. Можно менять по желанию, только не попадите на какой-нибудь существующий порт, занятый нужной службой (типа 22, 23, 80, 143 и подобных).

```
# Restrict webcam connections to localhost only (default: on)
```

```
webcam_localhost off
```

— доступ к камере только с localhost; если стоит "on", камера не будет видна из сети.

```
# Image width (pixels). Valid range: Camera dependent, default: 352
width 320
```

— ширина картинка; для случая нашей камеры, максимум — 640.

```
# Image height (pixels). Valid range: Camera dependent, default:
288
height 240
```

— аналогично — высота, максимум — 480.

```
# Maximum number of frames to be captured per second.
```

```
# Valid range: 2-100. Default: 100 (almost no limit).
```

```
framerate 2
```

— количество кадров в секунду. Сами решайте; для нашего случая 2 в секунду самое то.

```
# Rotate image this number of degrees. The rotation affects all
saved images as
```

```
# well as mpeg movies. Valid values: 0 (default = no rotation), 90,
180 and 270.
```

```
rotate 0
```

— можно повернуть изображение на заданное количество градусов (если камера прикреплена под углом).

```
# Threshold for number of changed pixels in an image that
```

```
# triggers motion detection (default: 1500)
```

```
threshold 1500
```

— этот параметр можно охарактеризовать как чувствительность при определении движения. Чем меньше значение, тем чувствительнее детектор.

```
# Noise threshold for the motion detection (default: 32)
```

```
noise_level 32
```

— уровень шума. Позволяет игнорировать шум, создаваемый камерой. Установите оптимальное значение для своей камеры; меня устроило по умолчанию.

```
# Detect motion in predefined areas (1 - 9). Areas are numbered
like that: 1 2 3
```

```
# A script (on_area_detected) is started immediately when motion
is 4 5 6
```

```
# detected in one of the given areas, but only once during an event.
7 8 9
```

```
# One or more areas can be specified with this option. (Default: not
defined)
```

```
; area_detect value
```

— позволяет следить только за определенной областью. 1 — левый верхний угол, 2 — верхняя средняя часть, ..., 5 — центр, ..., 9 — правый нижний угол. Если область не определена, то слежение происходит по всему экрану.

```
# The quality (in percent) to be used by the jpeg compression
(default: 75)
```

```
quality 75
```

— качество jpeg-сжатия.

```
# Make automated snapshot every N seconds (default: 0 = disabled)
snapshot_interval 0
```

— интервал в секундах, между которыми будут автоматически делаться снимки при наличии движения.

```
# Locate and draw a box around the moving object.
```

```
# Valid values: on, off and preview (default: off)
```

```
# Set to 'preview' will only draw a box in preview_shot pictures.
```

```
locate off
```

— позволяет рисовать рамку вокруг движущегося объекта. Интересная опция, попробуйте.

```
# Draws the timestamp using same options as C function
strftime(3)
```

```
# Default: %Y-%m-%d\n%T = date in ISO format and time in 24
hour clock
```

```
# Text is placed in lower right corner
```

```
text_right %Y-%m-%d\n%T-%q
```

— рисуем ли текст на кадре; если да — то где и в каком формате.

```
# Target base directory for pictures and films
```

```
# Recommended to use absolute path. (Default: current working
directory)
```

```
target_dir /tmp/motion
```

— каталог, в котором мы будем складывать файлы видео и картинки. далее подробно расписываются имена для разных вариантов записи.

Далее для краткости опишу сразу всю секцию:

```
# External Commands, Warnings and Logging:
```

```
on_event_start value
```

```
on_event_end value
```

```
on_picture_save value
```

```
on_motion_detected value
```

```
on_area_detected value
```

```
on_movie_start value
```

```
on_movie_end value
```

```
on_camera_lost value
```

— выполняет определенное действие по наступлении определенных событий. В качестве value можно указать скрипт, например, управляющий вам сообщением. Или включать музыку при обнаружении кого-либо. Всё ограничивается вашей фантазией и зависит от вашей цели.

Прочитайте внимательно — создатели предусмотрели практически всё, что нужно для эффективного наблюдения и регистрации событий.

➤ **Пропустили номер?** Узнайте на с. 108, как получить его прямо сейчас.

Дальше идут параметры для связи с базами данных *MySQL* и *PostgreSQL* и настройки конфигов для нескольких камер. Не описываю: кто пользуется этими службами, настроит это сам.

Закрываем, перезапускаем *motion*:

```
service motion restart
[ ok ] Restarting motion...[...] Stopping motion detection daemon:
motion.
[ ok ] Starting motion detection daemon: motion.
done.
```

Заходим из локальной сети через браузер на IP-адрес сервера (на порт 8081, если вы его не изменяли) — например, так: <http://192.168.0.30:8081/>.

И наслаждаемся картинкой. Обратите внимание на нижний правый угол картинки — там стоят дата и время, ровно так, как мы задали в файле */etc/motion/motion.conf*.

Но наслаждаемся недолго — вы же не будете открывать одно окно браузера для панели управления, а второе — для камеры? Сейчас очень модно стало слово «дашборд [dashboard]», означающее «приборная панель». Так вот, сведем на один дашборд кнопки и картинку.

## Первый дашборд

Откройте файл, который выводит кнопки и индикаторы. У меня их несколько; обкатываю на *index-test.php*, потом переносу удавшиеся изменения в *index.php*.

В конец текста, после закрывающей скобки *?>* (конец *php*) и перед тегом *</body>*, добавьте строку

```

```

Перезагрузите web-страницу — внизу, после кнопок, должна появиться область видеокamеры. Место *width="25%"* делает окно видеопотока в 25% от общей ширины окна браузера; можете менять это число так, как вам нравится.

Ну что, всё? Можно праздновать победу? Ан нет. Давайте поразмышляем, что будет, если вы захотите «пробросить» порт web-сервера наружу, в Интернет. Сейчас я не про безопасность, а про то, что отобразится на дашборде. Ваш браузер будет искать адрес *192.168.0.30* в своей локальной сети. И точно не найдет открытый порт 8081 с видеопотоком.

Поэтому надо найти способ сделать так, чтобы браузер знал точный адрес хоста, в котором расположен ваш ПД. Положим, вы сделаете выход в Интернет через хост *111.222.333.444*; тогда надо пробросить порт 8081 на этом хосте на внутренний IP-адрес *192.168.0.30* на тот же порт. И, убрав предыдущее изменение, добавить строку

```

```

Успех? Попробуйте зайти из локальной сети. В случае удачи вы всё же увидите окно камеры. Но если вы используете динамический DNS, или неправильно прописали правила *iptables* на сервере, то вы не увидите ничего.

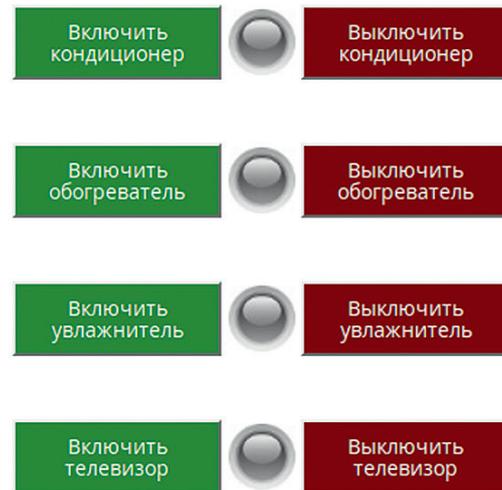
А нет ли способа узнать, на какой адрес мы стучимся? Пусть хост сообщит браузеру, какой его настоящий IP-адрес в данный момент. Тогда при доступе из локальной сети это будет *192.168.0.30*, при доступе извне — тот IP-адрес, через который вы входите на свой дашборд.

Такой способ есть, для этого в *php* есть массив переменных *\$\_SERVER*. Он содержит информацию о заголовках, путях и местоположении скриптов. Записи в этом массиве создаются web-сервером. Для начала мы воспользуемся только одной переменной.

Определим функцию, которая будет извлекать полный путь к IP-адресу камеры и помещать ее в переменную. Внутри тегов *<?php ...?>* ближе к началу страницы расположите строки:

```
function getUrl() {
    $url = @( $_SERVER["HTTPS"] != 'on' ) ?
    'http://'.$_SERVER["SERVER_NAME"] : 'https://'.
    $_SERVER["SERVER_NAME"];
```

## Здравствуйте, я ваш Послушный дом



```
$url .= ":8081";
return $url;
}
```

А в том месте, где вам надо вывести область камеры (опять-таки внутри кода *php*), расположите две строки:

```
$cam = getUrl();
echo "<img src=$cam width='25%'>";
```

Нужно ли комментировать этот великолепный прием, позволяющий независимо от протокола (HTTP или HTTPS) иметь верный IP-адрес сервера? Думаю, что не стоит.

В результате вы должны получить нечто, похожее на рис. 5.

Моя собака тоже решила принять участие в написании статьи, а заодно показать себя. Если честно, она просто пытается достать печенье.

Небольшое замечание: с этого момента учитывайте загрузку командного контроллера. Одновременные запись информации на медленную карту памяти и вещание в дашборд могут перегрузить процессор. Если это станет критичным, отключите запись на диск (что там писать?). Достаточно будет закомментировать строку *# target\_dir /tmp/motion* в конфигурационном файле, и *motion* перестанет записывать, тем самым вполностью разгрузив процессор.

Ну что, теперь вполне можно показать первое приближение ПД кому-нибудь из домашних? Особенно если у вас наличествует смартфон с большим экраном, который можно использовать как пульт.

А в следующей серии вас снова ждет интересное продолжение, пока! **LXF**

Рис. 5. Дашборд, вариант 1.

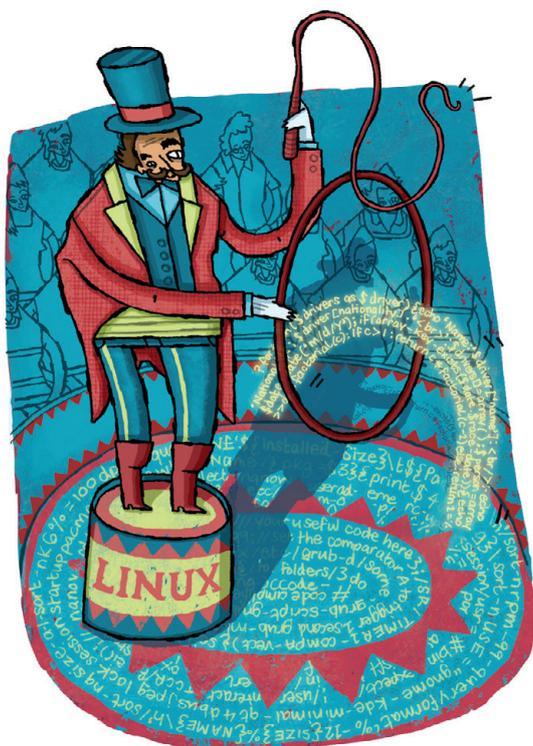
# Swift 3: Глянем, что новенького

Михалис Цукалос любит новизну. И после существенных изменений в языке от Apple пора ему установить и обновить Swift — и за программирование!



**Наш эксперт**

**Михалис Цукалос** — администратор, программист, администратор баз данных и математик, который любит писать статьи и узнавать что-то новое. С ним можно связаться через сайт [www.mtsoukalos.eu](http://www.mtsoukalos.eu) или в Твиттере — @mactsouk.



```

2. mtsouk@LTTng: ~/code/swift (ssh)
mtsouk@LTTng:~/code/swift$ ll
total 12
drwxrwxr-x  2 mtsouk mtsouk 4096 Mar 22 20:46 ./
drwxrwxr-x 11 mtsouk mtsouk 4096 Mar 21 11:45 ../
-rw-rw-r--  1 mtsouk mtsouk 173 Mar 22 20:44 systems.swift
mtsouk@LTTng:~/code/swift$ swiftc systems.swift
mtsouk@LTTng:~/code/swift$ file systems
systems: ELF 64-bit LSB executable, x86_64, version 1 (SYSV), dynamic
 /lib64/ld-linux-x86-64.so.2, for GNU/Linux 2.6.32, BuildID[sha1]
2236092e977f2dc9d, not stripped
mtsouk@LTTng:~/code/swift$ ./systems 1 2 3
./systems Is not a number!
Sum: 6
mtsouk@LTTng:~/code/swift$ ./systems -1 2 3
./systems Is not a number!
Sum: 4
mtsouk@LTTng:~/code/swift$ ./systems a b 1 2 c
./systems Is not a number!
a Is not a number!
b Is not a number!
c Is not a number!
Sum: 3
mtsouk@LTTng:~/code/swift$ ./systems a b
./systems Is not a number!
a Is not a number!
b Is not a number!

```

➤ Результат запуска файла `systems.swift`, который обрабатывает аргументы командной строки.

в переменную `PATH`, а последняя команда необходима для чтения файла `.profile` и обновления текущего пути `PATH`.

Чтобы узнать используемую версию Swift и проверить, что установка прошла успешно, выполните следующую команду:

```

$ swiftc -version
Swift version 3.0.2 (swift-3.0.2-RELEASE)
Target: x86_64-unknown-linux-gnu

```

После успешного завершения установки можно удалить файл `swift-3.0.2-RELEASE-ubuntu16.04.tar.gz` для экономии места на диске. И приступать к разработке на Swift 3!

## Привет, я Swift 3!

История языка Swift коротка, так как он появился только в 2014 г., но его будущее обещает многое. Swift быстро меняется. На момент написания этой статьи Swift 3.1 находится в состоянии бета-тестирования, а функции Swift 4 обсуждаются и окончательно согласовываются разработчиками. Поэтому хотя вскоре официально выйдет Swift 3.1, уже в июне 2017 г. на WWDC 2017 будет анонсирован Swift 4.0. Поэтому новые изменения ожидаются в самом ближайшем будущем. Тем не менее, Swift 3 является относительно зрелым, открытым, безопасным, быстрым и современным языком программирования, который становится всё лучше и лучше.

Дополнительная информация об этих изменениях доступна на сайте [www.github.com/apple/swift](http://www.github.com/apple/swift), а страница сообщества Swift с руководствами и рекомендациями — на [www.swift.org/contributing](http://www.swift.org/contributing).

**Н**а этом уроке мы рассмотрим последнюю версию языка программирования Swift. Сначала объясним, как установить Swift 3 в Ubuntu, и рассмотрим основные различия между Swift 2 и Swift 3. Потом поговорим о некоторых удобных функциях Swift, а далее разработаем две простых, но полезных программы на Swift 3.

Помните, что единственный способ получить максимальную отдачу от нашего урока — самостоятельно писать код на Swift 3! Так как Swift 3 не является стандартным пакетом в Linux, необходимо установить его вручную, выполнив действия, описанные ниже. Итак, в Ubuntu Linux надо выполнить следующие действия:

```

$ sudo apt-get install clang
$ wget https://swift.org/builds/swift-3.0.2-release/ubuntu1604/swift-3.0.2-RELEASE/swift-3.0.2-RELEASEubuntu16.04.tar.gz
$ tar xzf swift-3.0.2-RELEASE-ubuntu16.04.tar.gz > /dev/null
$ sudo mv swift-3.0.2-RELEASE-ubuntu16.04 /usr/local
$ echo 'export PATH=/usr/local/swift-3.0.2-RELEASEubuntu16.04/usr/bin:$PATH' >> ~/.profile
$ . ~/.profile

```

Объем файла `.tar.gz` составляет около 112 МБ, поэтому на выполнение команды `wget` может потребоваться некоторое время. Команда `echo` используется для добавления каталога Swift

### Скорая помощь

Дополнительная информация о Swift доступна на сайтах [www.swift.org](http://www.swift.org) и <https://developer.apple.com/swift>. Вся подоготную Swift 4 и процесса перехода на Swift 4 со Swift 3 можно узнать на странице [www.swift.org/blog/swift-4-0-release-process](http://www.swift.org/blog/swift-4-0-release-process).

## Чем привлекательны универсальные типы?

Идея универсальных типов состоит в том, что они позволяют разрабатывать типы и функции, которые работают с любыми типами данных, даже с теми, которых еще не существует! Многие встроенные типы, такие как массивы, реализуются с использованием универсальных типов. Это позволяет им работать с любыми типами данных, которые вы им передаете. Тем не менее, поскольку вы не знаете

точного типа переменной, которую собираетесь обработать, вы не можете выдвигать никаких предположений о типе и значении переменной. Поэтому не всегда возможно реализовать общие функции для таких операций, как сортировка, сложение или проверка равенства.

С другой стороны, реализовывать функцию сложения для всех существующих типов данных

не только долго, но и довольно глупо! Таким образом, универсальные типы позволяют не создавать отдельные функции для каждого типа данных, которые существуют или найдут свое существование в будущем, но могут сделаться дополнительным источником трудностей для ваших программ.

Оправданы ли эти сложности — решать вам самим.

## Сравнение Swift 2 и Swift 3

В чем заключаются различия в синтаксисе между Swift 3 и Swift 2? Хороший вопрос!

Следующий код Swift больше не является корректным в версии Swift 3:

```
print(add(1, b:2))
```

Попробовав скомпилировать его, вы получите следующее сообщение об ошибке:

```
add.swift:5:10: error: missing argument label 'a:' in call print(add(1,
^
b:2))
a:
```

Как видите, Swift 3 больше не позволяет опускать метку для первого аргумента функции. Для вызова предыдущей функции нужен следующий код:

```
print(add(a:1, b:2))
```

Это изменение синтаксиса, скорее всего, сделает весь ваш код на Swift 2 неработоспособным!

Как вы увидите далее на нашем уроке, для доступа к аргументам командной строки Swift 3 использует **CommandLine.arguments** вместо **Process.arguments**. Это приведет к тому, что большинство ваших утилит командной строки тоже перестанут работать.

Наконец, операторы **++** и **--** в Swift 3 больше не поддерживаются, вместо них следует использовать **+= 1** и **-= 1**.

Между версиями 2 и 3 есть и другие различия, но большинство из них связаны с фреймворком Cocoa, который используется в iOS и macOS.

## Обработка ошибок

Чтобы проиллюстрировать обработку ошибок в Swift, возьмем небольшую функцию Swift под названием `division()`, которая реализует деление целых чисел. Существует два способа обработки ошибок: внутри функции или вне функции. Оба способа приемлемы, но большую часть времени ошибки обрабатываются после возврата функции, поскольку это позволяет более точно решить, что делать с ошибкой.

Ошибки в Swift 3 соответствуют протоколу `Error`, и вы можете создать новое сообщение об ошибке следующим образом:

```
enum DivisionError: Error {
    case hasRemainder(remainder: Int)
    case divisionByZero
}
```

Как вы помните из уроков математики в школе, при делении целых чисел возникает два типа ошибок: во-первых, когда делитель равен нулю, и во-вторых, когда деление не является целым и имеет остаток. Первая ошибка — серьезная, вторая — нет. Однако сообщение об ошибке `DivisionError` должно поддерживать обе ошибки. Кроме того, `hasRemainder` возвращает остаток от деления в качестве переменной.

Функция `division()`, которая использует это сообщение об ошибке, включает следующий код:

```
func division(a: Int, b: Int) throws -> Int {
    if b == 0 {
        throw DivisionError.divisionByZero
    }
    if a % b != 0 {
        throw DivisionError.hasRemainder(remainder: a%b)
    }
    return a / b
}
```

Ключевое слово `throws` указывает, что метод может выдавать ошибки через утверждение `throw`, и помещается перед оператором `return`. В остальном в реализации функции нет ничего особенного. Ошибки могут быть перехвачены и за пределами функции. Для этого используются несколько операторов `catch`, как показано в файле `errorH.swift`. При запуске файла `errorH.swift` мы получим следующий результат:

```
$. ./errorH
Not enough arguments [Не хватает аргументов]
$. ./errorH 0 0
Division by zero! [Деление на ноль!]
$. ./errorH 10 2
Result: 5
$. ./errorH 10 3
Not a perfect division. Remainder: 1 [Не целочисленное деление. Остаток: 1]
$. ./errorH 10 -2
```

### Скорая помощь

Swift также поддерживает функциональное программирование, расширения, расширения протоколов и многое другое. Но вы не обязаны использовать все функции Swift в своей программе. Вместо этого оцените плюсы и минусы каждой функции и используйте ее в тех случаях, когда это облегчит вашу жизнь как разработчика или упростит структуру вашей программы.

```
sets.swift — code
1 var aSet = Set<Int>()
2
3 print(aSet)
4 let number = 1
5 if(aSet.contains(number)) {
6     print(number, "is in the set")
7 } else {
8     print(number, "is not in the set")
9 }
10
11 var count = 0
12 aSet.insert(1)
13 aSet.insert(1)
14 for element in aSet {
15     count += 1
16 }
17 print("Number of elements in aSet:", count)
18
19 var setA : Set<Int> = [0, 1, 2, 3, 4]
20 var setB : Set<Int> = [3, 4, 5, 6]
```

» В коде файла `sets.swift` вы видите, как выполнять итерацию по множеству с помощью цикла `for` и как вставить элемент во множество методом `insert()`.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)



Result: -5

Главный урок здесь в том, что действия, которые надо выполнить при появлении ошибки, определяются разработчиком.

## Системное программирование

Основная причина использования Swift 3 на компьютере Linux — разработка системных программ. Прочтя этот раздел, вы сможете создать простую утилиту командной строки, которая считывает аргументы командной строки (они должны быть целочисленными), складывает их и выводит результат. Приятно, что код файла **systems.swift** умеет отличать настоящие целые числа от аргументов других типов. Кроме того, программа выведет список проигнорированных аргументов пользователю.

Код файла **systems.swift** таков:

```
var sum = 0
for argument in CommandLine.arguments {
    if let number = Int(argument) {
        sum += number
    } else {
        print(argument, "Это не число!")
    }
}
print("Sum:", sum)
```

Обратите внимание, что предыдущий код работает только в Swift 3, поскольку перечисление `enum` в **CommandLine** в Swift 2 называется "Process". При запуске файла **systems.swift** будет получен результат, показанный на рис. на стр. 80.

## Природа множеств

Хотя множества (они же — наборы [set]) — функция в Swift не новая, это очень интересная концепция, которая встречается во многих языках программирования. Давайте поговорим о множествах (соответствующий код можно найти в файле **sets.swift**). Множество — это набор уникальных значений, которые имеют один и тот же тип и не упорядочены. В файле **sets.swift** наиболее важен следующий код:

```
var aSet = Set<Int>()
print(aSet)
let number = 1
if(aSet.contains(number)) {
    print(number, "is in the set")
}
```

```
1 import Glibc
2
3 func readInteger() -> Int {
4     print("Guess my number > ", terminator:"")
5     while true {
6         if let myInput = readLine() {
7             if let number = Int(myInput) {
8                 return number
9             } else {
10                print(myInput, "Is not an integer!")
11                print("", terminator:"")
12            }
13        }
14    }
15 }
16
17 func getRandom(min:Int, max:Int) -> Int {
18     srandom(UInt32(time(nil)))
19     var guessMe:Int = Int(random())
20     guessMe = guessMe % (max - min) + 1 + min
21     return guessMe
22 }
23
24 var min:Int = 0
25 var max:Int = 0
26 if CommandLine.argc < 3 {
27     print("Not enough arguments")
28     exit(0)
29 }
30 if let n1 = Int(CommandLine.arguments[1]) {
31     min = n1
32 } else {
33     print("Not enough arguments!")
34     exit(0)
35 }
36 if let n2 = Int(CommandLine.arguments[2]) {
37     max = n2
```



Хотя Swift 3 — мощный язык программирования, существуют и другие языки, которые могут вас заинтересовать, в том числе Go и Rust. Однако если вам также интересна разработка для iOS или macOS, то Swift 3 — на данный момент единственный вариант, который у вас есть.

➤ На рисунке показан код в файле **guess.swift**, который включает функции получения ввода от пользователя и генерации случайных чисел в заданном диапазоне.

```
2. mtsouk@LTTng: ~/code/swift (ssh)
mtsouk@LTTng:~/code/swift$ swiftc guess.swift
mtsouk@LTTng:~/code/swift$ ls -l guess
-rwxrwxr-x 1 mtsouk mtsouk 19480 Mar 23 10:23 guess
mtsouk@LTTng:~/code/swift$ file guess
guess: ELF 64-bit LSB executable, x86-64, version 1 (SYSV),
reter /lib64/ld-linux-x86-64.so.2, for GNU/Linux 2.6.32, Bu
e9aea192a0c170388982ae9e2, not stripped
mtsouk@LTTng:~/code/swift$ ./guess
Not enough arguments
mtsouk@LTTng:~/code/swift$ ./guess 1
Not enough arguments
mtsouk@LTTng:~/code/swift$ ./guess 1 a
Not enough arguments!
mtsouk@LTTng:~/code/swift$ ./guess 1 4
4
Guess my number > 1
1
Try something bigger
Guess my number > 3
3
Try something bigger
Guess my number > 5
5
Try something smaller
Guess my number > 4
4
Well done!
```

➤ Так работает игра в файле **guess.swift**, и вот какие результаты от нее можно ожидать.

```
} else {
    print(number, "is not in the set")
}
```

В первом утверждении из предыдущего кода определяется переменная множества (Set) с целочисленными значениями. Вторая инструкция выводит все элементы переменной множества. Кроме того, функция `aSet.contains()` проверяет, принадлежит ли указанный элемент множеству, что очень удобно.

Код файла **sets.swift** приведен на рис. стр. 81. Из этого кода вы можете увидеть, как найти пересечение, разность и объединение для двух множеств. После компиляции и запуска файла **sets.swift** мы получим следующий результат:

```
$. /sets
[]
1 is not in the set [Элемента 1 во множестве нет]
Number of elements in [Число элементов в] aSet: 1
Union [Объединение]: [2, 4, 5, 6, 0, 1, 3]
Intersect [Пересечение]: [4, 3]
Subtracting [Разность]: [2, 0, 1]
```

## Приближаемся к замыканиям

Замыкания [closure] — компактные блоки кода, похожие на функции, но без имени и полного объявления функции. Поэтому замыкания лучше всего подходят для задач, реализуемых в небольшом объеме кода Swift. Чтобы воспользоваться замыканием, его надо назначить переменной — по крайней мере, в большинстве случаев.

Незатейливый код Swift этого раздела можно найти в файле **closure.swift** — ниже приведена его важная часть, в которой два одинаковых замыкания определяются двумя различными способами:

```
var aTimesB: (Int, Int) -> Int
aTimesB = { (a: Int, b: Int) -> Int in
    return a * b
}
var otherWay: (Int, Int) -> Int
otherWay = {
    $0 * $1
}
```

Как вы видите из двух определений замыканий, замыкания заключаются в фигурные скобки. Стрелка `->` внутри фигурных

➤ Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

## Получение пользовательского ввода

Наряду с получением параметров в командной строке иногда надо запросить данные у пользователя, пока программа еще запущена. Пример кода приведен ниже, и его можно найти в файле `getInput.swift`:

```
var valid:Bool = false
print("> ", terminator:"")
while valid != true {
    if let myInput = readLine() {
        if let number = Int(myInput) {
            print("Чтение: ",myInput, separator:"")
            valid = true
        } else {
            print(myInput, "Это не целое число!")
        }
    }
}
```

```
print("> ", terminator:"")
}
}
```

Функция `readLine()` получает ввод пользователя, который завершается символом `newline` [перевод строки]. Использование свойства `terminator` в функции `print()` позволяет печатать без перевода строки, а свойство `separator` позволяет вывести несколько строк без пробелов между ними. Использование свойств `terminator` и `separator` не влияет на работу функции. При запуске файла `getInput.swift` мы получим следующий результат:

```
$. ./getInput
> a
a Это не целое число!
> -4a
-4a Это не целое число!
> -1
Чтение: -1
```

`getInput.swift` продолжает запрашивать ввод пользователя, пока не будет введено действительное целое число.

Код файла `game.swift` также используется в файле `game.swift` в функции под названием `readInteger()`.

скобок отделяет аргументы от типа возвращаемого значения. Оба замыкания получают два целых числа как входные параметры и возвращают целочисленные значения. Ключевое слово `in`, которое можно опустить, если отсутствует заголовок (например, при определении `otherWay`), отделяет заголовок замыкания от тела замыкания. Кроме того, вы не обязаны указывать имена для аргументов замыкания — в этом случае достаточно обращаться к аргументам по их позиции: `$0`, `$1` и т.д. Наконец, у замыкания `otherWay` отсутствует утверждение `return`, так как Swift знает, что нужно возвращать! Хотя для небольших замыканий можно не указывать возвращаемое значение и сделать так, чтобы Swift сам определил его, лучше написать соответствующий код, чтобы избежать игр в догадки и неприятных ошибок!

При запуске файла `closure.swift` мы получим следующий результат:

```
$. ./closure
-3
-3
```

Наряду с простым использованием замыканий, как показано в файле `closure.swift`, замыкания можно передавать в качестве параметров функций. В этом случае присваивать замыкание переменной не нужно. Вместо этого достаточно указать код замыкания в качестве аргумента при вызове функции.

Обратите внимание, что если замыкание ничего не возвращает, то в качестве типа возвращаемого значения необходимо указать `Void`.

## Сыграем

Давайте попробуем разработать простую игру, в которой компьютер вычисляет случайное число, а пользователь пытается угадать его, получая подсказки от компьютера.

В следующем коде Swift описана функция, которая генерирует случайное число в указанном диапазоне:

```
func returnRandom(min:Int, max:Int) -> Int {
    srand(UInt32(time(nil)))
    var guessMe:Int = Int(random())
    guessMe = guessMe % (max - min) + 1 + min
    return guessMe
}
```

Большая часть кода игры связана с обработкой ошибок, ведь когда вы запрашиваете что-то у пользователя, не стоит ожидать, что он обязательно введет то, что нужно, или укажет правильное число аргументов:

```
if CommandLine.argc < 3 {
    print("Не хватает аргументов")
    exit(0)
}
```

Таким образом, приведенный код проверяет, достаточно ли указанное количество аргументов для командной строки. На рис. внизу стр. 82 приведен код Swift 3 файла `guess.swift`, а на рис. вверху стр. 82 показан запуск файла `guess.swift`. Несмотря на простоту этой игры, код в файле `guess.swift` довольно сложен, поскольку включает обработку ошибок и две очень удобных функции. Код со строки 24 по 41 работает с аргументами командной строки программы. Помните, что доверять пользователю нельзя никогда, и обработка ошибок обычно занимает львиную долю любой программы!

Логика игры реализована в строках с 43 по 55 благодаря функциям `readInteger()` и `returnRandom()`, которые реализованы в начале программы. Удобство функций состоит в том, что их легко применить в других программах, в других модулях или в других классах.

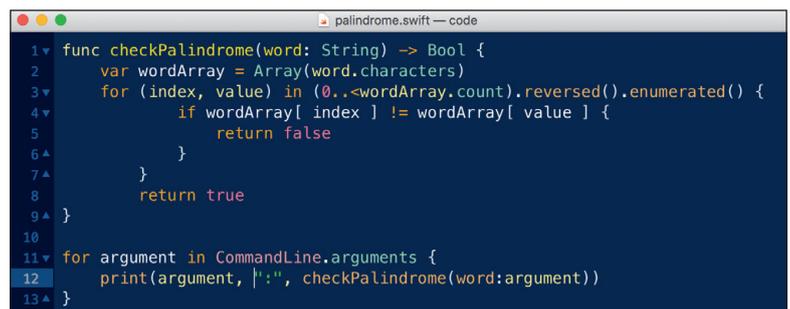
## Поиск палиндромов

Палиндром — это слово или фраза, читаемое одинаково слева направо и справа налево. В этом разделе мы представим программу `palindrome.swift`, которая проверяет, является ли слово палиндромом, и работает со строками.

При запуске файла `palindrome.swift` мы получим следующий результат:

```
$. ./palindrome 5 aba bb bc
./palindrome : false
5 : true
aba : true
bb : true
bc : false
```

На рис. внизу справа вы видите полный код файла `palindrome.swift`, который иллюстрирует работу со строками в Swift. Решив попробовать сделать нечто посерьезнее, вы можете написать программу, проверяющую, является ли слово анаграммой другого слова. **LXF**



```
1 func checkPalindrome(word: String) -> Bool {
2     var wordArray = Array(word.characters)
3     for (index, value) in (0..

```

➤ Код на Swift 3 в файле `palindrome.swift`. Функция `checkPalindrome()` проверяет, является ли строка палиндромом.



# Cordova: Ваяем для Android

Если у вас руки чешутся наштамповать приложений для Android с помощью Cordova, Netbeans и Android Studio, Кент Ельчук спешит помочь.



Наш эксперт

**Кент Ельчук** — опытный web-разработчик, который создал множество web-приложений для средних школ, колледжей и web-сайтов, настроил для них защиту и поддерживает их.

Еще в LXF223 мы рассказали о разработке приложений для Android и их публикации в Google Play. На этом уроке по разработке приложений мы воспользуемся собственными функциями и web-представлениями Android. На сей раз мы расскажем о еще двух методах разработки приложений для Android: Cordova и Ionic Framework.

Создание приложений в средах Cordova и Ionic обладает массой преимуществ, особенно для тех, у кого есть какой-либо опыт в web-дизайне или web-разработке. Преимущество Cordova и Ionic Framework заключается в том, что с ними можно создавать приложения на знакомых языках, таких как HTML, CSS и JavaScript.

В дополнение к удобному перечню языков, в Cordova и Ionic Framework есть плагины, которые можно добавлять и использовать в новых приложениях. Плагины позволяют настроить базу данных, получить данные местоположения GPS и выполнить другие действия.

Разобравшись с основами, перейдем к простым приложениям и запустим их на вашем тестовом устройстве. Хотя приложения Cordova и Ionic можно создавать и тестировать из командной строки, на нашем уроке мы воспользуемся Android Studio, Netbeans и параметрами командной строки для создания и изменения приложений для Android.

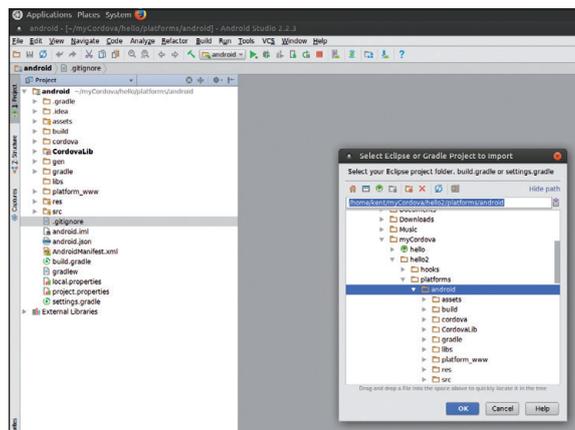
## Войдите в Android Studio

При создании приложений Android с помощью Cordova или Ionic вам могут понравиться простота и удобный графический интерфейс Android Studio. Однако поскольку все эти действия можно выполнить в другом редакторе — даже таком простом, как Gedit или Pluma — приложения удобно создавать и из командной строки. Если у вас нет Android Studio, ее можно загрузить на странице <https://developer.android.com/studio/index.html>.

Вернувшись к статье о приложениях Android в LXF223, вы найдете подробную информацию об установке Android Studio в Ubuntu 16.04 и настройке виртуализации в Linux. Это потребуется, если вы решите протестировать свои приложения на виртуальных устройствах. Если вы собираетесь установить Android Studio в первый раз, то это очень просто. В самом деле, всего через минуту вы сможете создавать и импортировать приложения.

Даже если вы будете создавать приложения без Android Studio, установить ее всё равно надо, поскольку вам понадобится Android SDK. Android SDK — это комплект инструментов для разработки программного обеспечения, используемый для создания приложений Android. Как я уже отметил, даже при сборке из командной строки всё равно будет использоваться Android SDK.

Вот несколько советов по использованию Android SDK. Вам нужно создать переменную среды ANDROID\_HOME. Если для сборки приложений Cordova вы планируете использовать Netbeans, установите путь ANDROID\_HOME в файле .profile. Хотя установка ANDROID\_HOME в файле .bashrc отлично подойдет для Android



Импорт проекта Cordova в Android Studio занимает всего секунду, если делать его из командной строки.

Studio и командной строки, для NetBeans такой вариант может не сработать.

Возможно, вы спрашиваете: а зачем нужен путь ANDROID\_HOME? Да просто нужен! Даже Netbeans подключается к сайту Android Studio для загрузки инструментов SDK. Итак, вам достаточно установить Android Studio и настроить переменную среды.

По умолчанию после установки Android Studio в переменной среды ANDROID\_HOME будет содержаться путь, похожий на /home/user/Android/Sdk. Разумеется, вместо "user" будет указано ваше имя пользователя в Linux.

Следующие команды создают переменную среды ANDROID\_HOME в файле .profile:

```
export ANDROID_HOME=/home/user/Android/Sdk
export PATH=${PATH}:%ANDROID_HOME/tools:%ANDROID_HOME/platform-tools
```

## А теперь о Cordova

Давайте перейдем к установке Cordova. Поскольку в состав Ubuntu 16.04 входит nodejs, для установки Cordova можно использовать команду npm:

```
sudo npm install -g cordova
```

После установки Cordova можно перейти к каталогу, где вы хотите создать новое приложение. В приведенном ниже коде показано создание каталога myCordova и первого простого приложения Cordova:

```
$ cd
$ mkdir myCordova
$ cd myCordova
$ cordova create test.com.test.website myTest
```

Последняя из приведенных выше команд создала тестовый каталог.



**Скорая помощь**  
При тестировании приложений на телефоне Android может потребоваться изменить некоторые параметры. Работу телефона также можно проверить с помощью Google. Дополнительную информацию см. на сайте <https://developer.android.com/studio/run/device.html>.

```
$ cd test
$ cordova platform add android --save
Следующая команда проверяет, что у вас есть нужные инстру-
менты для работы:
$ cordova requirements
Requirements check results for android:
Java JDK: installed 1.8.0
Android SDK: installed true
Android target: installed android-19,android-20,android-
21,android-22,android-23,android-24,android-25
Gradle: installed
```

А эта команда выполнит сборку проекта:  
\$ cordova build

После запуска этой команды среди прочего вы должны увидеть строку 'BUILD SUCCESSFUL'. Если что-то пошло не так, возможно, это связано с неправильной установкой разрешений или проблемами с путем `ANDROID_HOME`. Посмотрите на вывод программы в консоли — из сообщения об ошибке обычно можно всё понять. Если у вас нет разрешений для каталога `/home/user/cordova`, настройте их с помощью следующей команды:

```
$ chown -R user:user /home/user/cordova
```

Для тестового запуска можно использовать приведенную ниже команду. Однако перед этим убедитесь, что вы подключили телефон Android к компьютеру. Обратите внимание, что на вашем телефоне может появиться подтверждение для открытия приложения. Просто нажмите ОК и наблюдайте за тем, как запускается приложение:

```
$ cordova run android
```

Итак, мы успешно собрали программу. Теперь надо изменить приложение, создать подписанный файл APK для Google Play и отправить приложение в Google Play. Создать файл APK и отправить его в Google Play нетрудно. Основная часть работы относится к изменению приложения.

Для изменения приложения можно пойти одним из двух путей. Первый вариант — использовать любой редактор и командную строку Linux, второй вариант — делать всё это в *Android Studio*. Что же выбрать, что же выбрать?..

Поскольку этот выбор может быть непростым, в следующих разделах мы расскажем о том, как использовать командную строку с любым текстовым редактором (даже *Vim*) и как сделать это в среде разработки, такой как *Netbeans* и *Android Studio*.

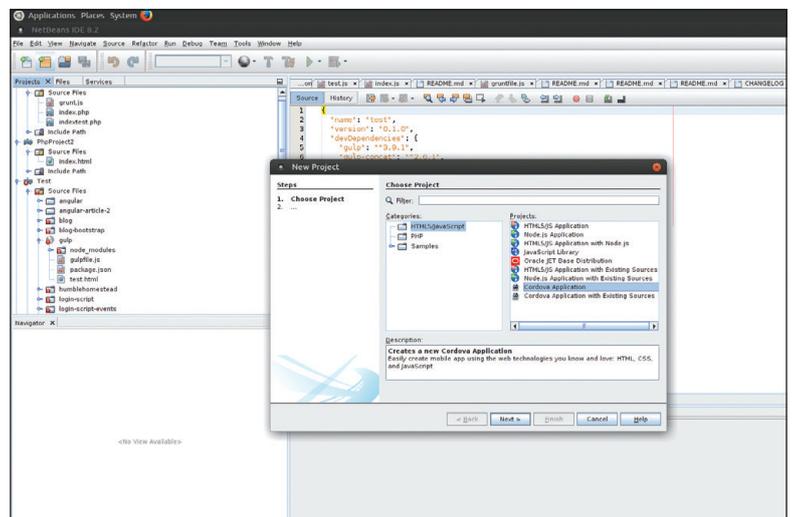
Несколько слов о сборке в *Netbeans*. Если вы читали мою последнюю статью о web-разработке в **LXF224**, то помните, что в ней я рассказывал о *Netbeans* и его использовании. В этой статье мы также расскажем об использовании *Netbeans* для создания приложений Cordova, так как это позволит web-разработчикам, возможно, тем, кто прочел последнюю статью, получить в свое распоряжение знакомый набор приемов.

## Сборка из командной строки

Раньше мы выполнили сборку приложения Cordova из командной строки. Основной рабочий каталог расположен в каталоге `home:/home/username/myCordova/test/www/`. Главный файл — `index.html`.

Для начала щелкните по файлу правой кнопкой мыши и откройте его в текстовом редакторе по вашему выбору. Затем измените какой-нибудь текст, например, 'Device is Ready' или 'Connecting to Device', и сохраните файл.

Чтобы увидеть изменения, перейдите в терминале в корневой каталог проекта, набрав команду `cd`. В данном случае корневой каталог представляет собой каталог `test`, находящийся в каталоге `myCordova`. Чтобы перейти в этот каталог, скомандуйте »



» *Netbeans* упрощает создание нового проекта Cordova или импорт уже существующего. А также имеет множество других функций: например, позволяет запуск приложения.

# Использование плагинов Cordova

В Cordova есть целый список плагинов, которые можно добавить к приложению. Список плагинов помещен на странице <https://cordova.apache.org/plugins/?platforms=cordova-android>.

Для начала добавим в приложение Cordova плагин геолокации. Перейдите в рабочий каталог приложения и введите команду

```
cordova plugin add cordova-plugin-geolocation
```

Теперь перейдите в редактор и откройте файлы `index.html` и `index.js`. Добавьте в файл `index.html` такой код:

```
<div><button id = "myPosition">My Position</button></div>
```

а в конец файла `index.js` — такой:

```
function myPosition() {
    var gedit = navigator.geolocation
```

```
getCurrentPosition(onSuccess, onError);
function onSuccess(position) {
    alert('Широта: ' + position.coords.latitude + '\n' +
        'Долгота: ' + position.coords.longitude
    + '\n' +
        'Высота: ' + position.coords.altitude + '\n' +
        'Точность: ' + position.coords.accuracy +
        '\n' +
        'Точность по высоте: ' + position.coords.altitudeAccuracy + '\n' +
        'Направление: ' + position.coords.heading
    + '\n' +
        'Скорость: ' + position.coords.speed
    + '\n' +
        'Время: ' + position.timestamp + '\n');
};
```

```
function onError(error) {
    alert('code: ' + error.code + '\n' + 'сообщение: ' +
        error.message + '\n');
}
```

Наконец, добавьте строку в строке 31 файла `index.js`:

```
document.getElementById("myPosition").
    addEventListener("click", myPosition);
```

Если коротко, JavaScript реагирует на нажатие кнопки и вызывает функцию `myPosition()`. В этой функции используется объект `geolocation`, который делает всю работу. Вызывается метод `navigator.geolocation.getCurrentPosition`, который выполняет обратный вызов метода `onSuccess()`. Этот метод выводит широту, долготу и временную отметку.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)



cd myCordova/test. Теперь просто выполните команду `cordova run android`, и вы увидите, как измененное приложение запустится на телефоне.

Мы не собираемся улучшать наше приложение, чтобы оно делало нечто полезное, так как это отвлечет нас от описания команд, необходимых для сборки и отправки приложения. Если у вас есть базовые навыки web-разработки с использованием HTML, CSS, Javascript и навыки редактирования изображений, то вы сможете без труда приступить к работе (надо лишь добавить несколько плагинов). Вы также можете добавить в свое приложение мобильную версию JQuery с сайта <https://jquerymobile.com/download/>. Мы поговорим об улучшении приложения и плагинах подробнее чуть позже, когда закончим рассказ об *Android Studio*.

Последний этап в создании приложения с командной строки — создание подписанного файла APK, который можно отправить в Google Play. Существует два простых способа создания подписанного файла APK: с помощью *Gradle* и с помощью *apksigner*.

## Gradle, я слепил это из глины

Для начала перейдите в каталог **Android**, где находится файл **gradle.build**. Затем выполните команду `keytool` для генерации ключевого файла. Используйте следующую команду:

```
user:~/myCordova/hello2/platforms/android# keytool
-genkey -v -keystore my-release-key.jks -alias myalias -keyalg
RSA -keysize 2048 -validity 10000
```

После запуска этой команды вам надо будет задать пароли. Обязательно запомните эти пароли, поскольку их придется добавить в файл **gradle.build**.

Сделаем это, откройте файл **gradle.build** в редакторе, который также находится в папке **Android**. Добавьте `signingConfigs` и `buildTypes` сразу после закрывающей скобки `defaultConfig`. Всё это расположено внутри скобок **Android** — `{}.`

Проще всего воспользоваться поиском: нажмите `Ctrl+F` и укажите в строке поиска 'defaultConfig'. При нажатии на открывающую фигурную скобку выделится закрывающая, и вы увидите, где следует добавить новый блок текста:

```
defaultConfig {Вставьте текст между этими скобками}
signingConfigs {
    release {
        storeFile file("my-release-key.jks")
```

```
storePassword "password"
keyPassword "password"
keyAlias 'myalias'
}
}
buildTypes {
    release {
        signingConfig signingConfigs.release
    }
}
```

Изменив файл **gradle.build** в папке **Android**, запустите команду, указанную ниже, чтобы создать подписанный файл APK. В строке ниже показан текущий каталог, в котором можно запустить команду, и сама команда:

```
user:~/myCordova/hello2/platforms/android# ./gradlew
assembleRelease
```

Заглянув в каталог **android/build/outputs/apk**, вы увидите появившийся файл APK, который можно загрузить в Google Play. В данном случае файл называется **android-release.apk**. Готово!

## Создаем приложения

В **LXF224** была подробная статья об использовании *Netbeans* для web-разработки. Кроме этого, *Netbeans* можно использовать для создания и импорта приложений Cordova для Android. Если у вас не установлен *Netbeans*, его можно загрузить и запустить за несколько минут. Обратите внимание, что *Netbeans* входит в состав Java JRE. Таким образом, вам потребуется использовать Java JDK для разработки приложений Cordova с *Netbeans*. Это не слишком сложно. Просто установите JDK и измените файл **netbeans.conf**, расположенный в каталоге *Netbeans*.

Путь к файлу будет примерно таким: **home/user/netbeans-8.2/etc/netbeans.conf**. Как видите, *Netbeans* находится в домашнем каталоге. Ниже приведено изменение в коде, которое нужно выполнить. Обратите внимание, что старый путь JRE, который ссылается на java, закомментирован, а ниже добавлен новый путь. Обычно JDK устанавливается в каталог **/usr/lib/jvm**. Всегда проверяйте этот каталог, чтобы определить, какая у вас версия:

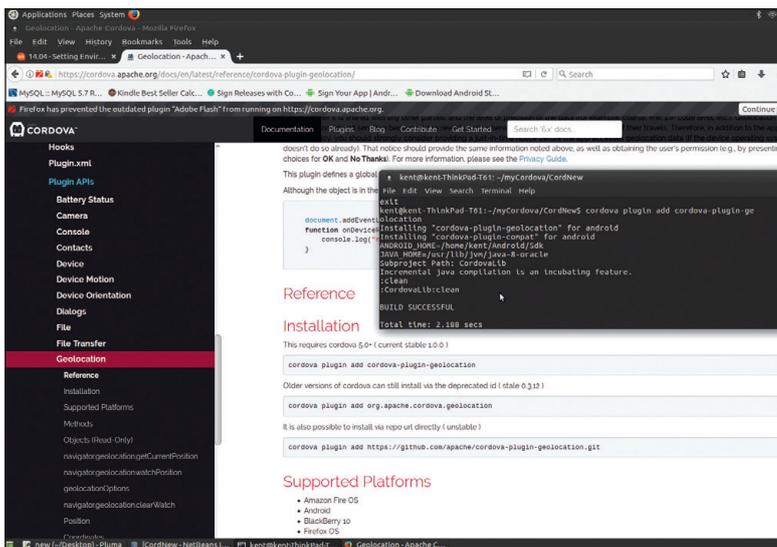
```
#netbeans_jdkhome="/home/user/netbeans-8.2/bin/jre"
netbeans_jdkhome="/usr/lib/jvm/java-1.8.0-openjdk-amd64"
```

В *Netbeans* можно создать новое приложение Cordova или импортировать приложение, созданное из командной строки. Однако для подписания приложения необходимо использовать командную строку. Таким образом, *Netbeans* находится где-то посередине между командной строкой и *Android Studio*.

Чтобы запустить новое приложение Cordova с *Netbeans*, выберите **File > New Project > Select HTML5/Javascript category > Cordova application > Next > Name Project > Select an Online template > Next > Finish [Файл > Новый проект > Выберите категорию HTML5/Javascript > Приложение Cordova > Далее > Укажите имя проекта > Выберите онлайн-шаблон > Далее > Готово]**. Если посмотреть в нижнее окно, вы увидите строку 'BUILD SUCCESSFUL' [Сборка выполнена успешно].

Хотя сборка приложения успешно завершилась, следует убедиться, что приложение собрано для платформы Android. Если другие варианты не работают, это можно сделать в терминале Linux или в командной строке с *Netbeans*. Лично мы создали бы проект Cordova с командной строки и импортировали существующий проект.

Обратите внимание, что при сборке и запуске приложения на устройстве Android с помощью *Netbeans* приложение становится совместимым с Android. Опять же, вы всегда можете добавить платформу Android через командную строку в качестве альтернативы.



➤ В проект Cordova можно добавить множество плагинов для выполнения целого ряда интересных задач, таких как геолокация вашего положения с помощью фотокамеры.

➤ Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

## Установка среды Ionic

Наряду с Cordova есть и другая среда для создания приложений, под названием Ionic. Она очень похожа на Cordova в установке и использовании.

Разумеется, первым делом надо ее установить.

Давайте это сделаем:

```
sudo npm install -g cordova ionic
```

Затем создайте приложение. Для примера можно создать каталог **mylonic** в своем домашнем каталоге.

```
cd mylonic
```

```
ionic start test sidemenu --v2
```

Теперь не помешает проверить приложение в браузере.

Просто войдите в корневой каталог приложения, запустите приложение и после этого откройте адрес <http://localhost:8100> в браузере для доступа к приложению:

```
ionic serve
```

Далее перейдем к устройству. Настроить приложение для Android и запустить его на мобильном устройстве можно двумя следующими командами:

```
cordova platform add android
```

```
ionic run android --device
```

Ну вот, начинайте работу над приложением в выбранном редакторе. Дополнительную информацию можно найти на сайте Ionic Framework — <https://ionicframework.com/docs/v1/guide/building.html>.

Чтобы открыть командную строку в *Netbeans*, перейдите в главное меню и выберите **Window > IDE Tools > Terminal** [Окно > Инструменты среды разработки > Терминал]. После открытия терминала перейдите в каталог с приложением командой `cd`.

Другой вариант — загрузить проект Cordova из существующих файлов и импортировать созданный проект Android в командной строке. Так, запустив всего несколько команд, вы легко импортируете проект, уже настроенный под устройства Android.

Создав проект Cordova, вы можете редактировать файлы, создавать и запускать приложения с помощью *Netbeans*, или изменять проект в *Netbeans* и создавать и запускать проект в командной строке.

## Перейдем в Studio!

Теперь импортируем новый проект в *Android Studio*. Откройте *Android Studio* и выберите **File > New > Import Project** [Файл > Создать > Импортировать проект]. Проект, который мы будем импортировать — это созданный каталог `test`. После выбора каталога откроется следующее окно. Установите галочку **Create Project From Existing Sources** [Создать проект из существующих источников], а затем нажмите **Next** [Далее].

Если вы увидите сообщение, что папка недоступна для записи, обеспечьте пользователю соответствующие разрешения, сделав ее доступной. Следующие команды решат и ту, и другую проблему:

```
cd
# chown -R user:user myCordova
# chmod -R 755 myCordova
```

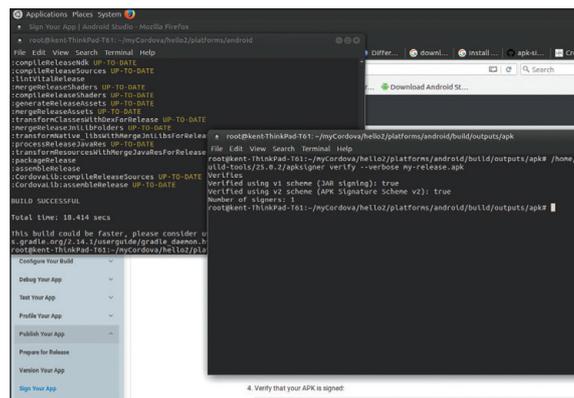
Дальнейшие действия не требуют особых пояснений. Вы можете изменить название проекта или просто нажать **Next** [Далее] до завершения настройки проекта. Сделав это, вы увидите структуру каталогов в *Android Studio*. Если этого не произошло, убедитесь, что в левой части меню и над файлами выбрано **Project** [Проект].

Всё становится проще. Каталог, в котором можно создать приложение, находится в каталоге `www`. Эта схема должна быть хорошо знакома web-дизайнерам и web-разработчикам, поскольку обычно сайты на Linux находятся в каталоге `var/www` или `var/www/html`.

Итак, чтобы точно объяснить ситуацию, скажу, что файл `index.html` — это файл, который загружается при запуске приложения. Как и для обычных сайтов, файлы CSS, изображения и файлы javascript находятся в каталогах `css`, `img` и `js`. Просто чтобы увидеть, как это быстро и легко, можете подключить свой телефон Android к компьютеру Linux. Вы увидите, как страница загрузилась.

Файл APK можно подписать с помощью *Android Studio* или с командной строки.

Следующий шаг — подписать файл APK. Для этого выберите **Build > Generate Signed APK** [Сборка > Создать подписанный файл APK]. Этот процесс не требует пояснений. Вам понадобится выбрать путь для хранения ключей на вашем компьютере. Сделав это, нажмите **Next** [Далее]. Убедитесь, что тип сборки — 'Release',



➤ Для проверки, что приложение корректно настроено для Google Play, оглядите файл APK. Это приложение было подписано и проверено в командной строке Linux.

а затем нажмите **Finish** [Готово]. Если всё пойдет по плану, вы увидите всплывающее окно без ошибок и предупреждений. Кроме того, файл APK, который будет отправлен в Google Developer, находится в каталоге `android/build/outputs/apk`. Другой способ — воспользоваться инструментом под названием *apksigner*.

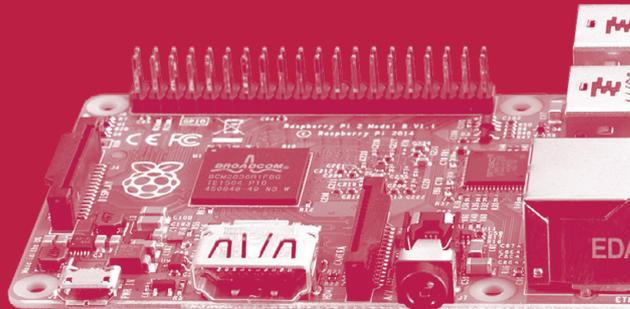
## Знаки и знамения

Простейший способ подписать файл APK с помощью *apksigner* — открыть в одном терминале каталог **Android**, а в другом — каталог **apk**. Дело в том, что для запуска команды вы должны находиться в каталоге **Android**, а для запуска других команд над файлами **apk** — в каталоге **apk**. Приведенные ниже команды можно использовать для создания подписанного файла APK. Учтите, что надо использовать инструмент сборки Android SDK версии выше 24.0.3, так как в последних инструментах есть *apksigner*, а в более ранних версиях — нет.

Вот что делают приведенные ниже команды. Первая создает неподписанный файл сборки **apk**. Команда `zipalign` копирует и выравнивает созданный файл. Третья команда делает всю трудную работу и завершает процесс:

```
user:~/myCordova/test/platforms/android# ./gradlew assembleRelease
user:~/myCordova/test/platforms/android/build/outputs/apk# zipalign -v -p 4 android-release-unsigned.apk androidrelease-unsigned-aligned.apk
user:~/myCordova/test/platforms/android/build/outputs/apk# /home/user/Android/Sdk/build-tools/25.0.2/apksigner sign --ks my-release-key.jks --out my-release.apk android-releaseunsigned-aligned.apk
Новый файл my-release.apk готов, и его можно проверить:
user:~/myCordova/hello2/platforms/android/build/outputs/apk# /home/kent/Android/Sdk/build-tools/25.0.2/apksigner verify --verbose my-release.apk
```

На этом всё. Теперь у вас есть компьютер Linux для разработки приложений Android с использованием Cordova. Удачи с вашим следующим приложением! **LXF**



Ваша порция смачных новостей, обзоров и учебников от Raspberry Pi

**ЛЕС ПАУНДЕР**  
Наш постоянный эксперт по pi'рожкам. А еще он пишет о Raspberry Pi.



Привет...

**Н**едавно я посетил Pcademy, образовательную программу Raspberry Pi для преподавателей. Pcademy только что отметила свой третий день рождения, за это время более 1000 человек в Великобритании и Америке сделали с их помощью первые шаги в мире изобретательства.

Логично предположить, что в Pcademy ответом на все случаи жизни будет Raspberry Pi. А вот и нет. От Джеймса Робинсона [James Robinson], главы Pcademy в Великобритании, мы узнали, что на рынке есть другие платы, и каждая вносит что-то свое. Micro:bit широко распространялась в школах: она проста и понятна, что детям и надо. Я могу написать свой проект на Python, JavaScript или PXT и тут же увидеть результат. Настраивать ее совсем или почти не требуется: micro:bit отображается как USB-накопитель, а значит, дети могут сразу браться за дело!

Raspberry Pi — отличная платформа, но с ней надо сначала разобратся. Записать SD-карту; изучив кое-какие команды терминала, попробовать Python... для многих сегодня это уже не проблема. Мы вышли на новый уровень проектов и задач, и эти «тривиальные» шаги уже забыты. Но нам следует помнить, что наше сообщество объединяет массу участников с разным уровнем способностей. И лучшее в сообществе Raspberry Pi — то, что все здесь друг другу помогают. Форумы, социальные медиа и блоги — всё это служит источником полезной информации.

Если вы можете помочь сообществу, сделайте это! Помогая друг другу, вы делаете свой вклад в копилку общих знаний.

## Премия Turkey Blocks, борцу с Интернет-цензурой

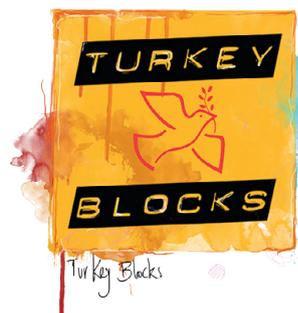
Система по борьбе с Интернет-цензурой на базе Pi получает премию за активность.

**В** Турции всё здорово, просто лучше всех, вы и не поверите, сколько прорывных открытий там совершается. Именно поэтому группа по борьбе с интернет-цензурой создала службу интернет-мониторинга, действующую по всей стране, чтобы отслеживать блокировки локального доступа со стороны правительства и предупреждать о них. За 2016 г. система выявила 14 случаев массовых блокировок. Эта система, разработанная Turkey Blocks, получила премию Digital Activism 2017, присуждаемую Index on Censorship.

Turkey Blocks — это независимая внепартийная организация, занимающаяся исследованиями в области технологий и выступающая за свободу цифровой информации. Они выявляют и публикуют отчеты по массовым случаям применения интернет-цензу-

ры в Турции. Подробности можно увидеть на сайте [www.turkeyblocks.org](http://www.turkeyblocks.org).

Turkey Blocks являются частью глобальной коалиции KeepItOn ([www.accessnow.org/keepiton](http://www.accessnow.org/keepiton)) по борьбе с ограничениями в Интернете.



► Создайте себе инструмент отслеживания цензуры и приглядывайте за благими намерениями правительства.

## PiPlay

Глядите, портативная консоль!

**P**iPlay — портативная игровая система на базе Pi Zero, уже превзошедшая свой запрос на Kickstarter. Состоящая из 2,4-дюймового дисплея, специальной печатной платы и питания из двух пальчиковых батареек, она предоставляет до трех часов игр в ретро-стиле. Узнайте больше здесь: <http://bit.ly/LXF225piplay>



► Консоль вам в руки.

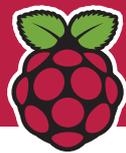
## Продажи Zero

На самом деле, далеко не нулевые...

**К**ак заявили в Pi Foundation, продажи Pi Zero W за первые девять недель превысили 250 000 экземпляров. Кроме того, добавилось еще 13 дистрибьюторов. Это должно решить затруднения с поставками Pi Zero — проблеме, которая заявила о себе сразу после его выхода. <http://bit.ly/LXF225pizero>



► Теперь есть и новый официальный корпус!



# Pimoroni Scroll Bot

Приходя в себя после очередной порции Pi, дождливым утром Лес Паундер решил соорудить себе ухмыляющегося робота. Желаем удачи...

## Вкратце

» Комплект, в состав которого входят новый Raspberry Pi Zero W, акриловые части с лазерной резкой и панель Scroll pHAT HD. Сам комплект — прекрасное средство для постепенного приобретения навыков, необходимых для создания физических проектов на базе Raspberry Pi Zero W. От вас нужны только SD-карта, питание и клавиатура/мышь.

**R**aspberry Pi Zero W вышел всего несколько месяцев назад, но уже успел наделать много шума. Перед компьютером за £10 с Wi-Fi и Bluetooth устоять трудно; ну, а если вы в этом совсем новичок? Не бойтесь: у таких производителей, как Pimoroni, есть целый арсенал продуктов, чтобы помочь вам освоиться в мире Raspberry Pi.

Scroll Bot — это комплект для самостоятельной сборки, в котором есть всё необходимое для создания простых гаджетов с Raspberry Pi Zero W. Вы получаете Pi Zero W и новейшую плату расширения Scroll pHAT HD (подробнее об этом дальше). Вы также получаете адаптеры, позволяющие использовать Pi Zero W с экраном HDMI и USB-устройствами. Вместо источника питания USB, в комплекте есть разъем USB A и micro USB. Учитывая скромные требования Pi Zero W к питанию, вы легко найдете источник у себя дома. Набор для сборки рамы робота производится лазерной резкой на том же оборудовании, что и для изготовления знаменитых корпусов Pirow, весь крепеж тоже включен в комплект.

Сборка набора деталей с лазерной резкой проста. Нужно припаять контакты Pi Zero W и Scroll pHAT HD, или, если не умеете, использовать заклепки, которые крепятся молотком. После сборки комплекта вам останется только установить Raspbian; разве что SD-карту придется взять свою, поскольку она в комплект не входит.

После установки и загрузки Raspbian надо будет установить библиотеки Python для платы Scroll pHAT HD с помощью инсталлятора, а затем запустить примеры,

содержащиеся в нем и позволяющие увидеть, что плата умеет делать.

Scroll pHAT HD пришла на смену предыдущей плате Scroll pHAT и представляет собой сетку светодиодов 17×7, в общей сложности — 119 белых светодиодов. И это куда лучше, чем предыдущая Scroll pHAT, размером 11×5, из 55 светодиодов.

Scroll pHAT HD может применяться для прокрутки текста, анимации или простых изображений на матрице светодиодов. Для нашего обзора мы сделали инструмент для чтения RSS-канала, который прокручивал пятерку главных новостей с разных сайтов, но вы можете легко использовать Scroll pHAT HD для отображения информации о статусе сервера, погоде или твитах с конкретным хэштегом. Имеющиеся примеры включают от простой прокрутки текста до впечатляющих эффектов плазмы, демонстрирующих возможности платы путем сложных вычислений.

## Доступное решение

Комплект Scroll Bot — отличный способ познакомиться с Raspberry Pi. Может показаться, что £35 — это дорого, но по отдельности Pi Zero W с адаптерами и Scroll pHAT HD обойдутся в £26, а за £9 вы получите свой собственный набор компонентов с лазерной резкой, USB-шнур и довольно удобную коробку, где вы сможете хранить свои все умножающиеся аксессуары к Raspberry Pi или Maker.

Набор несложно собрать, хотя некоторых и затруднит пайка Pi Zero W и Scroll pHAT HD, но вы всегда можете обратиться за помощью в местный Makerspace. Кроме SD-карты, которую вам придется туда добавить, Pi Zero W готов к работе и может использоваться и как встроенное устройство, и как обычный компьютер.

Scroll pHAT HD — это плата для развлечения. У нее не так много применений, как у других плат на рынке, но то, что она умеет, она делает хорошо. Всего несколько строк кода Python — и вы получаете



» Этот симпатичный набор прост в сборке и будет прекрасно смотреться на полке у вас в офисе, на столе или в библиотеке.

средство для прокрутки текстов, просмотра новостей или отображения милой улыбки для робота! **LXF**

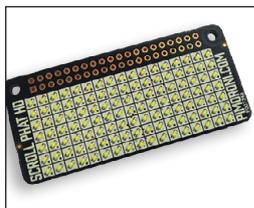


## Свойства навскидку



### Компоненты с лазерной резкой

Pimoroni славны компонентами лазерной резки; вот и комплект Scroll Bot добротен и прост в сборке.



### Scroll pHAT HD

Прокрутка текста и анимация способны вполне успешно справляться с задачами небольших объемов информации.

## LINUX FORMAT Вердикт

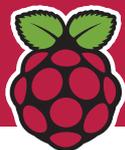
### Pimoroni Scroll Bot Kit

Разработчик: Pimoroni  
Сайт: <https://shop.pimoroni.com>  
Цена: £ 35

Функциональность	7/10
Производительность	8/10
Удобство в работе	8/10
Справедливость цены	8/10

» Увлекательное введение в Raspberry Pi Zero W. Послужит источником как информации, так и мягкого света в комнате.

**Рейтинг 8/10**



# GPIO: Для аналоговых сигналов

Лес Паундер показывает, как использовать аналоговые датчики и входы с Raspberry Pi для управления Neopixels.



**Наш эксперт**

Лес Паундер работает с Raspberry Pi Foundation в рамках обучения в Picademy и путешествует по Великобритании, помогая школам и учителям внедрять больше технологий. Его блог — <http://bigl.es>.



Все модели Raspberry Pi поставляются с GPIO — контактами, которые позволяют применять электронные компоненты и дополнительные платы. Но ни одна модель Raspberry Pi не может взаимодействовать с аналоговыми компонентами, поскольку Pi не поставляется с аналого-цифровым преобразователем (АЦП). Шагом вперед стал АЦП MCP3008. На нашем уроке мы используем его с тремя потенциометрами для управления кольцом neopixel.

## Оборудование

Ознакомьтесь с монтажной схемой (см. ниже «Вам нужны») для этого урока, в ней довольно много соединений.

Начнем сборку, вставив MCP3008 в макетную плату над центральной линией. Выемка на MCP3008 должна быть обращена к верхней части платы. MCP3008 подключается к Raspberry Pi по питанию и соединениями с шиной SPI. Эти соединения выполнены по одной стороне микросхемы (выводы с 9 по 16 согласно техническому описанию). Выводы с 1 по 8 зарезервированы для восьми каналов для работы с аналоговыми устройствами.

Потенциометры — это переменные резисторы с тремя выводами. Вращая ручку потенциометра, можно менять напряжение на центральном выводе. Выводы потенциометра — напряжение, выход и земля. Потенциометры бывают разных видов; мы используем однооборотные, подобные регуляторам громкости в усилителях. Есть еще движковые потенциометры — например, как на пульте микширования; а также подстроечные, используемые на печатных платах, где требуются нечастые регулировки.

Neopixels — это торговая марка, созданная Adafruit для светодиодов [LED] серии WS2811/12. У каждого светодиода в группе

индивидуально управляются: его цвет, яркость, включение и выключение. Для управления Neopixels требуется точная выдержка времени, и для этого надо использовать контакт GPIO на Pi, предназначенный для широтно-импульсной модуляции, ШИМ. Мы уже знаем, что это контакт 18. Для питания Neopixels также можно использовать выводы 3V и GND [Земля], которые были выведены на плату согласно схеме.

## Программное обеспечение

Неважно, какая версия Raspberry Pi используется, хоть Pi Zero; наша первая задача — отключить аудиовыход, так как он будет мешать Neopixels. Для этого надо изменить файл `config.txt` в каталоге загрузки, так что откройте терминал и введите

```
$ sudo nano /boot/config.txt
```

В конце файла в новой пустой строке добавьте следующие две строки, чтобы прокомментировать изменение для дальнейшего использования и отключить звуковой разъем 3,5 мм на Raspberry Pi:

```
#Для Neopixels. Шлет аудио на HDMI и выключает 3,5-мм jack.
hdmi_force_hotplug=1
```

Чтобы сохранить и выйти из редактора, нажмите Ctrl+O, затем Enter, затем нажмите Ctrl+X. Перезагрузите Pi, чтобы принять изменения.

После перезагрузки и возврата на рабочий стол можно установить библиотеку Python, которая позволяет управлять Neopixels. Наберите в терминале

```
$ sudo pip3 install rpi_ws281x
```

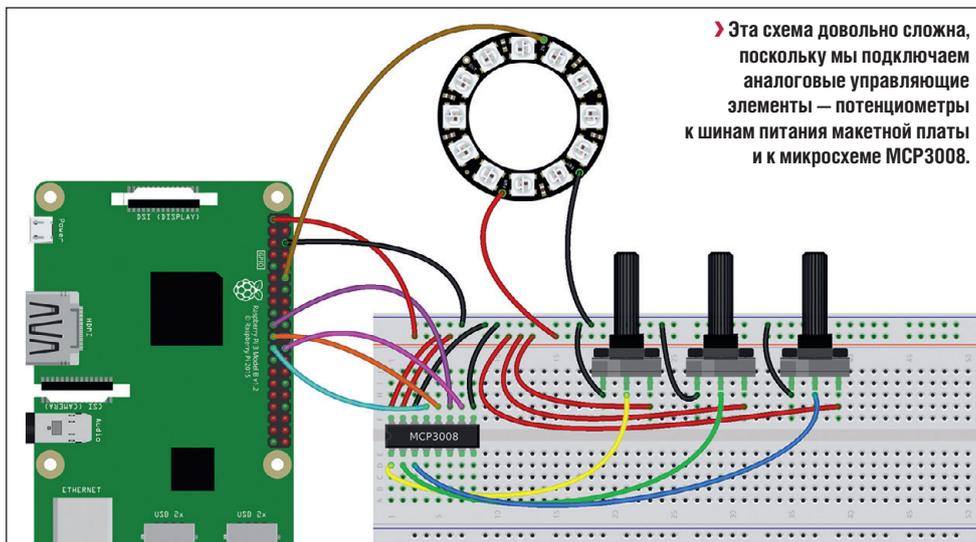
Через несколько мгновений библиотека установится. MCP3008 не требует какого-либо специального программного обеспечения,

## Вам нужны

- Любая модель Raspberry Pi
- Последний выпуск Raspbian Pixel
- Интернет-соединение для Pi
- АЦП MCP3008
- Любой Neopixels; мы использовали 12-пиксельное светодиодное кольцо фирмы Adafruit
- Оборудование для пайки
- Макетная плата
- 3 потенциометра 10 кОм
- 6 перемычек папа-мама
- 13 перемычек папа-папа

Все исходники и схемы для данного проекта можно найти на <http://bit.ly/2p5oN6A>.

Увидеть этот проект в действии можно на YouTube: <https://youtu.be/0W8un23Cl>.



➤ Эта схема довольно сложна, поскольку мы подключаем аналоговые управляющие элементы — потенциометры к шинам питания макетной платы и к микросхеме MCP3008.



## Neopixels, APA102 и другие светодиоды

Термин LED — сокращение от «светодиод» — несколько размытый. Есть скромный светодиод, используемый для индикации включения устройства, и чтобы сказать «привет, мир» при сборке вашей первой схемы, но есть и другие. Мы использовали Neopixels, торговую марку для светодиодов WS2811/12. Эти сверхяркие светодиоды работают от источника питания 3,3 или 5 В и используют импульсы тщательно подобранной

длительности для управления состоянием диода. Как мы только что узнали из урока, это вызывает проблемы, так как надо использовать вывод GPIO, способный выполнять ШИМ, и это заставило нас отключить аналоговый аудиовыход. Но есть и другие светодиоды, которые можно применить вместо Neopixel, например, серии APA102. Они используют аппаратный интерфейс SPI, способный передавать данные на светодиоды быстрее, а это удобно для

проектов, эксплуатирующих инерцию зрения. Светодиоды APA102 не требуют каких-либо изменений конфигурации, поскольку не влияют на аудиовыход. Многие компании, связанные с Pi, используют APA102 вместо WS2811/12, так как это устраняет неприятную проблему изменения настроек звука и обеспечивает более простой способ использования сверхярких индивидуально управляемых светодиодов с Raspberry Pi.

поскольку мы будем использовать его с GPIO Zero, у которого есть класс Python для работы с микросхемой.

### Программирование проекта

Начнем с открытия редактора Python 3, но его надо открыть через `sudo`, чтобы мы могли работать с Neopixels. Войдите в терминал и введите следующую команду, открывающую редактор:

```
$ sudo idle3 &
```

Когда он откроется, нажмите `File > New`, чтобы создать новый пустой файл; затем сразу нажмите `File > Save` и назовите файл `analogue-inputs.py`. Последующие сохранения теперь будут происходить гораздо быстрее.

Первый раздел кода Python — наш импорт. Сначала мы импортируем класс `MCP3008` из `GPIO Zero`, что позволит использовать чип. Затем импортируем функцию `sleep` из библиотеки `Time`, она поможет контролировать темп проекта. Наконец, импортируем класс `Adafruit_NeoPixel` из библиотеки `neopixel`, для управления `neopixel`.

```
from gpiozero import MCP3008
from time import sleep
from neopixel import Adafruit_NeoPixel
```

Наш следующий блок кода указывает Python, какой потенциометр подключен к какому каналу `MCP3008`. Мы также храним исходные значения, выводимые `MCP3008` в виде переменных `r` [`red` — красный], `g` [`green` — зеленый], `b` [`blue` — синий]. Они соответствуют цветам, используемым с Neopixels. Исходные значения находятся между 0.0 и 1.0.

```
r = MCP3008(channel=0)
g = MCP3008(channel=1)
b = MCP3008(channel=2)
```

Затем мы создадим две переменные — `LEDS`, соответствующую числу светодиодов в кольце, и `PIN`, указывающую контакт GPIO для управления Neopixels.

```
LEDS = 12
PIN = 18
```

Чтобы использовать Neopixels, нужно сообщить Python, куда они подключены — это делается созданием объекта `strip`. Затем мы даем указание Python начать их использовать.

```
strip = Adafruit_NeoPixel(LEDS, PIN)
strip.begin()
```

Цикл `while True` используется для постоянного выполнения кода, содержащегося внутри, и здесь первые несколько строк кода задают переменные `red`, `green` и `blue`. Они хранят значения, созданные поворотом потенциометров, но вы же помните, что эти значения находятся между 0.0 и 1.0? Чтобы подходить для Neopixels, они должны быть в пределах от 0 до 255, где 0 — выключено, а 255 — полная яркость. Поэтому мы берем исходное значение и умножаем его на 255, и это заключено в функцию `round`, которая округляет

число до ближайшего целого. Это дает нам значения, которые далее можно передать в функции `neopixels`.

```
while True:
    red = round(r.value * 255)
    green = round(g.value * 255)
    blue = round(b.value * 255)
```

Для целей отладки мы также выводим значения в оболочку Python, это поможет нам выявлять проблемы:

```
print(red, green, blue)
```

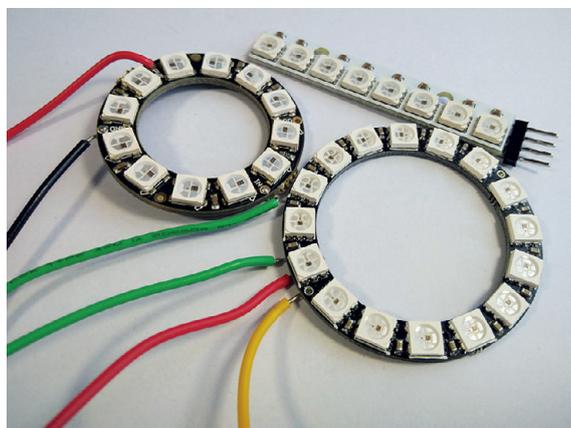
Цикл `for` используется для установки цвета всех светодиодов нашей полоски Neopixels. Мы взяли диапазон, который повторно использует переменную `LEDS` как число повторов цикла. Внутри цикла мы устанавливаем цвет каждого «пикселя», указывая, где он находится в полоске `i`, а затем даем значения красного, зеленого и синего, созданные поворотом потенциометров. Наконец, в цикле мы велим библиотеке Neopixel показать результаты. Если мы этого не сделаем, то не увидим никаких изменений.

```
for i in range(LEDS):
    strip.setPixelColorRGB(i, red, green, blue)
    strip.show()
```

Наша самая последняя строка кода находится вне цикла `for`, но всё еще внутри основного цикла. Это простая приостановка работы кода и сдерживание процессора от перегрузки!

```
sleep(0.1)
```

Завершив код, сохраните его и нажмите `Run > Run Module`, чтобы запустить код. Теперь поворачивайте потенциометры и следите за изменением цвета Neopixels. Вы только что научились принимать аналоговые входные значения и использовать их для управления выходом другого компонента. **LXF**

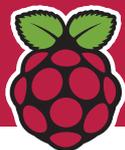


► Neopixels выпускаются в разных формах: здесь мы видим два кольца, подключенных и готовых к употреблению, и 8-пиксельную полосу с контактами.

### Скорая помощь

При использовании интегральных схем (ИС) следует понимать, для чего предназначен каждый вывод. Посмотрите на маркировку микросхемы, затем используйте свой любимый поисковик, чтобы найти технические характеристики.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)



# Explorer: ЖК-диоды и моторы

Лес Паундер показывает, как смело экспериментировать и мастерить с Explorer HAT Pro, универсальной платой, которую легко и дешево использовать.



## Наш эксперт

**Лес Паундер** — мастер и умелец, который путешествует с Raspberry Pi Foundation's Picademy, помогая педагогам сделать первые шаги с Pi, micro:bit и Arduino.

## Вам нужны

- » Любой Pi с 40 выводами, Raspbian Pixel
- » Explorer HAT Pro, 1 светодиод, 1 резистор 220 Ом, 1 кнопочный выключатель, двигатель постоянного тока, 6 проволочных переключек папа-папа, потенциометр 10 кОм
- » Скачайте код: [github.com/lesp/LXF225-ExplorerHAT/archive/master.zip](https://github.com/lesp/LXF225-ExplorerHAT/archive/master.zip)

» Потенциометр подает на аналоговый вход напряжение, зависящее от вращения ручки, которое меняет сопротивление.

В этом месяце мы проведем 10-минутные работы, выполнимые с помощью Explorer HAT Pro. Наши четыре проекта можно объединить. Мы используем аналоговые входы с потенциометрами, цифровые входы/выходы со светодиодами и кнопками, контроллер с H-мостом для управления двигателями постоянного тока и плату. Прежде чем подать питание на Raspberry Pi, надо подключить Explorer HAT Pro ко всем 40 выводам GPIO. Плата должна располагаться сверху и в габарит Raspberry Pi. Теперь подайте питание на Raspberry Pi и загрузите рабочий стол Pixel. Когда будете готовы, откройте терминал и введите такую команду, для установки библиотек Explorer HAT Pro:

```
$ curl https://get.pimoroni.com/explorerhat | bash
```

Следуйте инструкциям по установке, а если появится запрос на перезагрузку, сделайте это и вернитесь в рабочий стол Pixel. Теперь в главном меню перейдите в меню Programming. Нажмите Python 3, чтобы открыть редактор, затем нажмите File > New.

## Проект 1: переключение светодиода

Здесь мы будем использовать емкостные сенсорные кнопки (1–4) на Explorer HAT Pro для управления встроенными светодиодами (LED). Открыв пустой документ, нажмите File > Save и сразу сохраните файл как **Project1-LED-Toggle.py**. Не забывайте почаще сохраняться.

Мы начинаем код, импортируя библиотеку Explorer HAT Python3, а затем переименовываем ее в 'eh', чтобы сократить набор. Мы также импортируем функцию sleep из библиотеки time:

```
import explorerhat as eh
from time import sleep
```

Затем создаем бесконечный цикл — это основная часть кода:

```
while True:
```

Потом мы ставим серию проверок условий, отслеживая нажатие емкостных сенсорных кнопок (1–4). Каждой кнопке мы

назначаем цвет светодиода, который будет включаться/выключаться по нажатию на емкостную сенсорную кнопку. Затем вставим задержку, предотвращая случайное двойное нажатие кнопки:

```
if eh.touch.one.is_pressed():
    eh.light.blue.toggle()
    sleep(0.1)
```

Остальной код соответствует этому шаблону, но использует elif для предоставления проверок других условий:

```
elif eh.touch.two.is_pressed():
    eh.light.yellow.toggle()
    sleep(0.1)
elif eh.touch.three.is_pressed():
    eh.light.red.toggle()
    sleep(0.1)
elif eh.touch.four.is_pressed():
    eh.light.green.toggle()
    sleep(0.1)
```

Нажмите Run > Run Module, чтобы запустить код. Теперь нажимайте кнопки 1–4 на Explorer HAT Pro для переключения светодиодов.

## Проект 2: наглядное мигание

Для этого проекта вам понадобятся потенциометр и три переключки. Подключите согласно схеме (слева внизу). Потенциометр подключен к выводу ANALOG1 и подает с переменного резистора напряжение в соответствии с поворотом его регулятора. При полном напряжении будет чуть больше 5 В.

Создайте новый пустой файл с именем **Project2-Analog-Blink.py** и импортируйте библиотеку Explorer HAT:

```
import explorerhat as eh
```

Теперь внутри цикла while True создадим две переменные, считывающие и хранящие значение напряжения с ANALOG1:

```
while True:
    on = eh.analog.one.read()
    off = eh.analog.one.read()
```

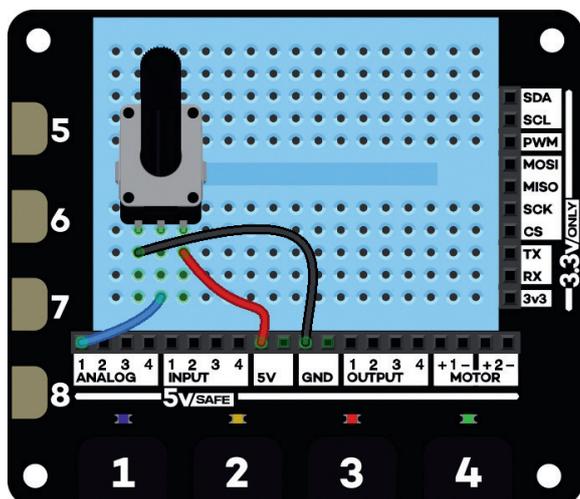
Затем мы применяем небольшой трюк, предотвращающий сбой кода Python из-за ошибки деления на ноль на основе аналоговых показаний. Если значение наших переменных on и off когда-либо достигает нуля, мы меняем их значения на 0,01:

```
if on == 0 and off == 0:
    on = 0.01
    off = 0.01
```

Наша последняя строка кода вызывает вспышки (мигание) встроенных светодиодов со временем включения/выключения, соответствующему напряжению, считываемому с ANALOG1. Это означает, что при низком напряжении светодиоды будут мигать быстрее, а при высоком — медленнее.

```
eh.light.blink(on,off)
```

Для запуска кода нажмите Run > Run Module. Вращайте ручку потенциометра, чтобы изменить частоту мигания светодиодов.





## Explorer'ы бывают разными

Здесь мы сосредоточились на Explorer HAT Pro, которая является лучшей в линейке плат. Она имеет восемь емкостных сенсорных площадок, четыре аналоговых входа, четыре цифровых входа и четыре выхода. Имеются контроллер двигателя DRV8833 и выводы для I2C/SPI и последовательных соединений. И мы получаем макетную плату для прототипирования.

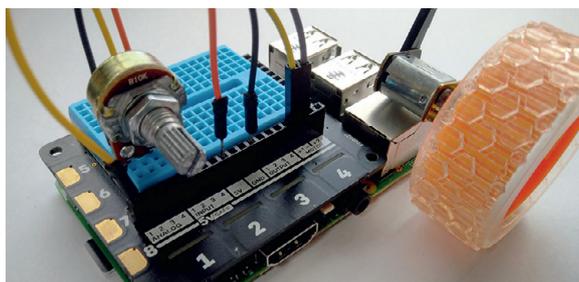
Для емкостных входов 1–4 лучше всего использовать руки, но те из вас, кто захочет проверить

электропроводность банана или использовать фольгу, или медную ленту для создания более крупных цепей, могут использовать 5–8 с зажимами-крокодилами и превратить всё в интересную форму ввода. Это очень похоже на Makey Makey — устройство, эмулирующее клавиатуру/мышь, способное превращать проводящие объекты в разную видность ввода.

Другим вариантом является Explorer pHAT, предназначенный для Pi Zero, но будет работать

и со всеми 40-выводными GPIO Pi. Explorer pHAT не имеет емкостного сенсорного интерфейса, макетной платы, светодиодов и выводов для I2C. Зато он вдвое дешевле версии Pro.

Итак, если вам необходимо решение «всё в одном» для роботов и аналоговых датчиков, то пригодится плата pHAT. Но если вы хотите, чтобы для вашего следующего проекта были доступны все функции, не исключено, что плата Pro вам подойдет больше.



Мы использовали микродвигатель с редуктором, так как он совместим с 5 В и отвечает требованиям робототехники.

### Проект 3: управление мотором

В этом проекте мы оставим потенциометр подключенным и добавим двигатель постоянного тока, подключив его к плюсу и минусу MOTOR1. Добавим простой контроллер скорости для двигателя постоянного тока, используя потенциометр из Проекта 2 для подачи входного сигнала. Explorer HAT Pro поставляется с контроллером двигателя DRV8833, который предлагает два канала двигателя и полное управление вперед/назад.

В начале этого проекта создайте файл с именем **Project3-Analog-Motor.py** и импортируйте библиотеку Explorer HAT:

```
import explorerhat as eh
```

Затем создадим бесконечный цикл:

```
while True:
```

Внутри цикла создадим переменную `speed` и в ней храним значения с `ANALOG1`, но для удобства использования мы округляем значение до одного десятичного знака. Затем мы умножаем значение на 20, чтобы получить процентное значение, которое затем передается в последнюю строку кода, двигающую мотор вперед и принимающую процентное значение для определения его скорости:

```
speed = (round(eh.analog.one.read()) * 20)
```

```
eh.motor.one.forward(speed)
```

Закончив код, нажмите `Run > Run Module`, чтобы его запустить. Теперь вращение ручки потенциометра изменяет скорость вращения двигателя.

### Проект 4: нажмите кнопку

Наш последний проект показывает, как входы и выходы Explorer HAT работают со внешними компонентами, используя кнопку для переключения встроенных светодиодов и стандартного светодиода.

Для этого проекта требуется кнопка, светодиод, резистор 220 Ом и четыре проволочных перемычки. Цепь собрана на макетной плате, как показано на рис. справа: кнопка подключена к 5V и INPUT1, длинный вывод светодиода подключен к выводу 5V, а короткий — к OUTPUT1 через резистор. Выводы OUTPUT Explorer HAT — это «земля [Ground]», и работают только при включении.

Создайте новый пустой файл с именем **Project4-Input-Output.py**, импортируйте библиотеку Explorer HAT, затем импортируйте функцию `sleep` из библиотеки `time`:

```
import explorerhat as eh
from time import sleep
```

Теперь давайте создадим цикл с условным тестом внутри, который будет проверять, нажаты ли емкостные сенсорные кнопки (1–4). Для кнопки 1 это выглядит так:

```
while True:
    if eh.touch.one.is_pressed():
```

Если эта кнопка была нажата, мы меняем состояние включен/выключен синего светодиода под кнопкой 1. Мы также добавляем короткий `sleep`, чтобы не реагировать на случайные двойные нажатия:

```
eh.light.blue.toggle()
sleep(0.1)
```

Затем создадим еще условные тесты `else if`, которые будут проверять, нажаты ли кнопки 2–4, и переключать светодиоды под каждой кнопкой:

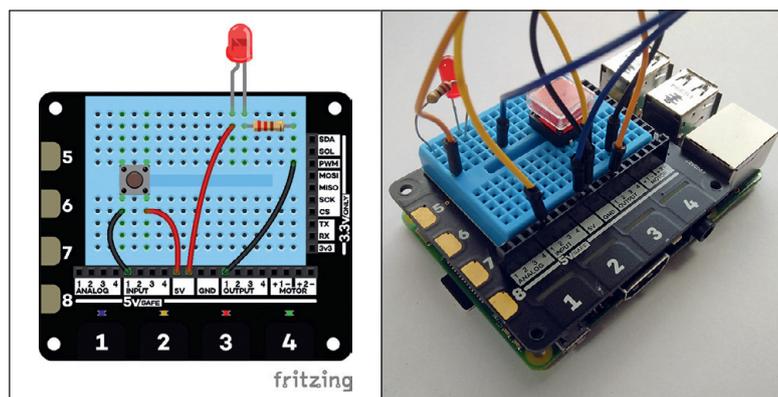
```
elif eh.touch.two.is_pressed():
    eh.light.yellow.toggle()
    sleep(0.1)
elif eh.touch.three.is_pressed():
    eh.light.red.toggle()
    sleep(0.1)
elif eh.touch.four.is_pressed():
    eh.light.green.toggle()
    sleep(0.1)
```

Нажмите `Run > Run Module` для запуска кода. Теперь можете нажимать емкостные кнопки 1–4, чтобы включать и выключать светодиоды.

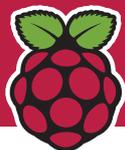
И на этом мы завершаем наши четыре проекта с Explorer HAT Pro! **LXF**



И к Explorer HAT Pro, и к Explorer pHAT можно подключать два мотора. Но вам понадобятся микродвигатели с редуктором, а не большие желтые двигатели постоянного тока, имеющиеся во многих наборах роботов.



Кнопки и светодиоды будут частыми компонентами многих проектов. Они обеспечивают селективный ввод, а вывод легко отлаживается.



# Python: Свет для дискотеки

Лес Паундер показывает, как оторваться ярчайшим образом в своей жизни, создав небольшой код Python для индивидуального контроля светомузыки.



## Наш эксперт

Лес Паундер — хакер и умелец, который любит повозиться с технологиями. Он ездит по Великобритании с Raspberry Pi Foundation, организует Picademy, и пишет о своих проектах на <http://bigl.es>.



## Скорая помощь

Если вы используете много NeoPixels, то у них будет высокое энергопотребление. Вы можете использовать внешний источник питания 5 В, если соединены заземление источника питания, NeoPixels и Raspberry Pi. В материалах для скачивания по данному проекту имеется схема примера, показывающего, как это сделать.

Когда обычный светодиод (LED) слабоват, приходится использовать WS2812 NeoPixel. На этом уроке мы узнаем, как управлять LED через индивидуальный графический интерфейс пользователя Python. Для нашего проекта понадобится любая модель Raspberry Pi, свежий релиз Raspbian Pixel и цепь WS2812 LED NeoPixels. (Мы обратились на <http://bit.ly/NeoPixelsOnEbay>.) Вам также понадобится соединительный кабель «штырь—гнездо (папа—мама)», а если вам необходима большая по объему цепь с большей яркостью, есть опциональное внешнее питание 5 В 4 А. Весь код и диаграммы в высоком разрешении можно найти на <https://github.com/lesp/LXF-Lightshow/archive/master.zip>.

Чтобы использовать LED WS2812 с нашим Pi, надо установить настройку Python. В Pi откройте терминал, и в этом терминале мы введем две команды:

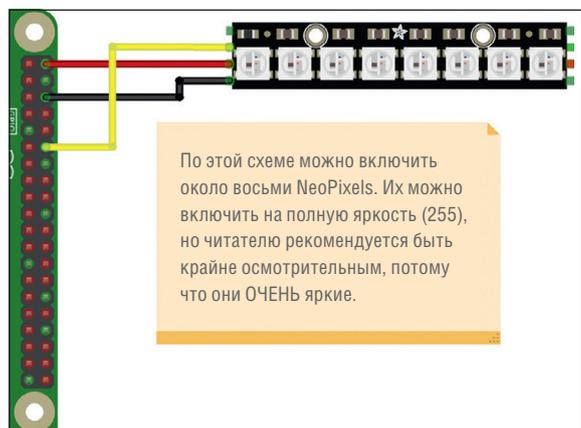
```
sudo apt update
sudo apt install build-essential python-dev git scons swig
```

Первая команда обновляет наш список программных пакетов. Вторая устанавливает инструменты для компиляции нашей библиотеки `rpi_ws281x`. Далее надо скачать экземпляр `rpi_ws281x` с GitHub, изменить директорию, куда был скачан код, и код скомпилировать:

```
git clone https://github.com/jgarff/rpi_ws281x.git
cd rpi_ws281x
scons
```

Затем перейдите в поддиректорию `python` по `cd python` и установите готовую к использованию библиотеку Python, командой `sudo python setup.py install`

Наш проект создаст графический интерфейс пользователя [Graphical User Interface] (GUI), через который мы будем контролировать цвета линейки светодиодов WS2812. Поэтому мы начнем



➤ Схема для этого проекта довольно проста, всего три соединения с Raspberry. Версия этой диаграммы с высоким разрешением находится в материалах для скачивания по данному проекту.

с того, что откроем редактор Python 2, который находится в разделе Programming [Программирование] главного меню. Открыв его, сразу же нажмем на File > Open [Файл > Открыть], чтобы создать новый пустой документ, и на File > Save [Файл > Сохранить]. Файл назовем `colourmixer.py`. Это значительно ускорит последующее сохранения. Наши первые несколько строк кода являются серией импортов:

```
import time
from neopixel import *
from Tkinter import *
```

## Почувствуйте бит

Первая строка импорта используется для контроля за скоростью нашего проекта. Импорт `neopixel import` — это библиотека `rpi_ws281x`, установленная ранее. Наш последний импорт — `Tkinter`, это библиотека Python для создания элементов GUI, таких, как окна и диалоговые окна.

Наш следующий раздел кода содержит переменные, используемые для хранения настройки нашего светодиода WS2812:

```
LED_COUNT = 60
LED_PIN = 18
LED_BRIGHTNESS = 8
```

`LED_COUNT` используется для показа, сколько диодов у нас в линейке. В нашем случае это было 60. `LED_PIN` обозначает вывод GPIO для подключения Pi к выводу Data In нашей линейки. Это должен быть вывод, поддерживающий широтно-импульсную модуляцию (PWM), поскольку мы используем его для отправки сигнала на ленту. `LED_BRIGHTNESS` — это значение между 0 и 255, где 255 — максимальная яркость. Чем ярче диоды, тем больше тока они потребляют, и для больших линеек это может стать важным. Мы взяли значение 8 — достаточно ярко, чтобы диоды было видно, но при этом не особо прожорливо в плане питания.

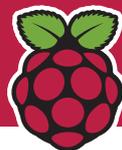
Далее у нас есть функция, которая группирует ряд шагов в один объект, который можно исполнить, вызвав его имя:

```
def color(strip, color):
    for i in range(strip.numPixels()):
        strip.setPixelColor(i, color)
    strip.show()
```

Наша функция называется `color` и принимает две опции (они же — аргументы). В данном случае это переменная с именем `strip`, которую мы создадим позднее. Второй аргумент — `color`, используемый для задания цвета диодов. Внутри функции мы используем цикл `for`, который будет перебирать все диоды в нашей линейке. При каждом прогоне цикла будет изменяться каждый диод в линейке, приобретая новый цвет. Чтобы это произошло, надо показать линейке изменения.

Далее у нас имеется функция с именем `show_values`, используемая для печати значений наших ползунков (их мы создадим позднее):

```
def show_values(value):
```



## Мерцающие огни!

LED (Light Emitting Diodes — светодиоды) являются простейшими компонентами; они пропускают ток в одном направлении, и результатом этого является видимый нами свет. Но это всё, на что способен скромный обычный диод, и если нам нужно нечто более мощное и гибкое, то здесь в игру вступает диод WS2812. Стандарт WS2812, обычно именуемый NeoPixels, является именем бренда, созданного Adafruit и используемого для обозначения линейки диодов с контролируемыми цветами и яркостью.

Светодиоды WS2812 поставляются в разнообразных формах, от простых «пикселей» до многих метров пикселей, соединенных вместе. В каждом пикселе есть контакт для ввода и вывода данных. Эти данные общаются через вывод 18 нашего Raspberry Pi, используя строгую временную последовательность, управляемую с помощью широтно-импульсной модуляции [Pulse Width Modulation]. Если тайминг нарушается, пиксели могут начать работать с перебоями.

Светодиоды поддерживаются многими устройствами. Raspberry Pi, micro:bit, Codebug, Arduino, ESP8266 — все они могут их контролировать, и они работают от 5 В, но длинным линейкам нужен внешний источник питания, способный обеспечить требуемую силу тока. Если Земля (-) источника питания соединена с Землей ленты и Raspberry Pi, общим заземлением, можно подключить к линейке источник питания 5 В и запитать NeoPixels, чтобы они сияли еще ярче!



```
print (red.get(), green.get(), blue.get())
color(strip, Color(green.get(),red.get(),blue.get()))
```

Чтобы прочитать и затем распечатать данные ползунков, мы должны их получить. Мы вызываем функцию `color` и передаем ей значения для зеленого, красного и голубого ползунков. Это запускает смену цвета в линейке NeoPixels. Далее мы переходим к двум строкам кода.

```
strip = Adafruit_NeoPixel(LED_COUNT, LED_PIN, LED_FREQ_HZ,
LED_DMA, LED_INVERT, LED_BRIGHTNESS)
strip.begin()
```

### Пусть горят

Первая строка используется для создания объекта с именем `strip`, использование которого мы уже не раз видели. Это объект, который относится к нашей линейке диодов и содержит всю необходимую информацию, например, сколько там диодов, какой вывод GPIO используется, и т.д. И так, теперь, когда бы мы ни использовали объект `strip`, вся его настройка проводится автоматически. Вторая строка дает задание нашему коду начать использовать линейку диодов.

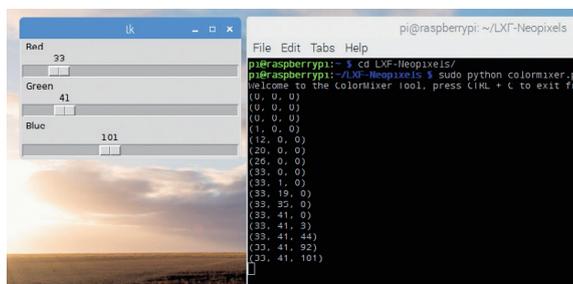
Теперь мы переходим к основной части и начинаем с оператора `try`, чтобы запустить код, написанный внутри него. Он начинается с простой функции `print`, которая покажет пользователю инструкции для ввода и выхода.

```
try:
    print ('Привет от инструмента ColorMixer, для выхода
из терминала жмите Ctrl+C.')
```

Для показа, что наш код работает, мы добавили простую анимацию, которая сделает все диоды зелеными, красными, а потом голубыми. Чтобы изменить цвет, мы вызываем функцию `color`:

```
for i in range(3):
    color(strip, Color(255, 0, 0)) # Зеленый
    time.sleep(0.3)
...
```

Мы даем функции команду изменить все диоды в линейке на цвет, получающийся смешением всех трех цветов. Это смесь красного, зеленого и голубого, производимого диодами. Мы применяем диоды GRB, поэтому первое число в последовательности из 0,0,0 поставлено для зеленого [Green], потом для красного [Red] и голубого [Blue]. Если ваши диоды — RGB, поменяйте местами первые две цифры. Между сменами цвета у нас сделана задержка 0,3 секунды, чтобы увидеть изменение. Прерывая цикл `for`, мы перенастраиваем линейку диодов, чтобы все они отключили цвет на данный момент — `color(strip, Color(0,0,0))`.



► Создание приложения *tkinter*, вроде наших ползунков, относительно безболезненное. И оно пригодится в других проектах.

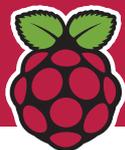
Отложим на минутку управление NeoPixel: мы начнем теперь использовать *tkinter*. Это библиотека GUI Python, и в данном случае для управления смесью красного, зеленого и голубого света мы употребим три ползунка.

```
master = Tk()
red = Scale(master, from_=0, to=255, orient=HORIZONTAL,
label= "Red", command=show_values, length= 300)
red.pack()
mainloop()
...
```

Сначала мы создаем объект с именем `master` для хранения функции `Tk()`. Далее мы создаем объекты с именами `red`, `green` и `blue`. Каждый из этих цветовых объектов является подвижной шкалой от 0 до 255. В диалоговом окне они горизонтальны, и каждый наименован в соответствии с цветом. Мы также даем каждому ползунку длину в 300 пикселей. Для каждого цвета вы также увидите нашу функцию `show_values`. Это означает, что ползунок передает свое значение функции, которая, в свою очередь, передает функции `color` и изменяет смесь цветов для наших диодов. Каждый цвет также имеет `pack`, создающий строку для каждого ползунка. Завершающая строка — `mainloop()`. В *tkinter* это специальный бесконечный цикл, который будет держать диалоговое окно открытым. Теперь нам нужен раздел `except`, который получает ввод с клавиатуры комбинации `Ctrl+c` для выхода из приложения и обеспечивает выключение линейки.

```
except KeyboardInterrupt:
    print("Выход")
    color(strip, Color(0,0,0))
```

Закончив код, сохраните свою работу, откройте Терминал, перейдите туда, где вы сохранили свою работу, и введите `sudo python colormixer.py`. Вы увидите, как загорится линейка NeoPixel, после чего на экране появится диалоговое окно. Поздравляем! Это наш интерфейс управления с ползунками для изменения цвета. **LXF**



# Minecraft Pi: СОТВОРИМ МИР

Джонни Бидвелл покажет вам удивительный и творческий мир *Minecraft*. А потом научит создать свой сервер, чтобы туда не лезла всякая шантрапа.



### Наш эксперт

Джонни Бидвелл в этом месяце в долгу перед Нейтом Дрейком, который, собственно, и написал данный учебник. Обычно Джонни делает свою работу сам, но в этом месяце он как-то завоевался с пластиковыми чашками.

Raspberry Pi стал источником вдохновения для нового поколения мастеров-самоучек, но помимо этого он ценится и как игровая платформа. Наличие проектов наподобие RetroPie, в сочетании с дешевой самой Pi, означает, что на нем прекрасно пойдет вся игровая классика. И даже кое-какие новинки. Включая *Minecraft*.

Если вдруг вы жили в отрыве от цивилизации с 2011 г., то *Minecraft* — это самая продаваемая компьютерная игра за всю историю. Это живущий своей жизнью абстрактный мир, в котором наш герой Стив [Steve] перемещается по пространству, составленному из блоков. Ресурсы можно добывать и создавать из других ресурсов, так что Стив может строить дома, собирать урожай, готовить еду и сражаться с врагами (и все эти удовольствия в пиксельном формате).

Специальная версия, *Minecraft: Pi Edition*, появилась в 2013 г. и с 2014 г. получила привязку к Raspbian. Эта версия является в известном смысле урезанной, поскольку в ней нет крафтинга или подвохов, равно как и ограничения на доступные ресурсы. Но есть в ней и функции, в основной версии отсутствующие — в первую очередь, Python API, дающий возможность управлять миром по ходу игры. Также вам можно не волноваться по поводу каких-либо злобных существ, пытающихся вас убить, или того, где добыть еду, что придется очень по душе юным игрокам, поскольку можно полностью сосредоточиться на творческих аспектах. API Python делает ее идеальной платформой для желающих научиться кодированию, в любом возрасте.

Чтобы начать работу, сначала перейдите в меню, затем откройте Игры > Minecraft Pi. Выберите Single Player [Режим одного игрока], а затем New Game [Новая игра]. Вы увидите, что окно игры немного смещено от окна приложения за ним. Убедитесь, что курсор мыши параллелен верхней части окна приложения, и удерживайте ее, чтобы переместить окно.

Нажмите Start Game [Начать игру], а затем Create New [Создать новый]. Ваш новый мир загружен. Можете смело исследовать и строить. Элементы управления для игры довольно просты и выглядят так:

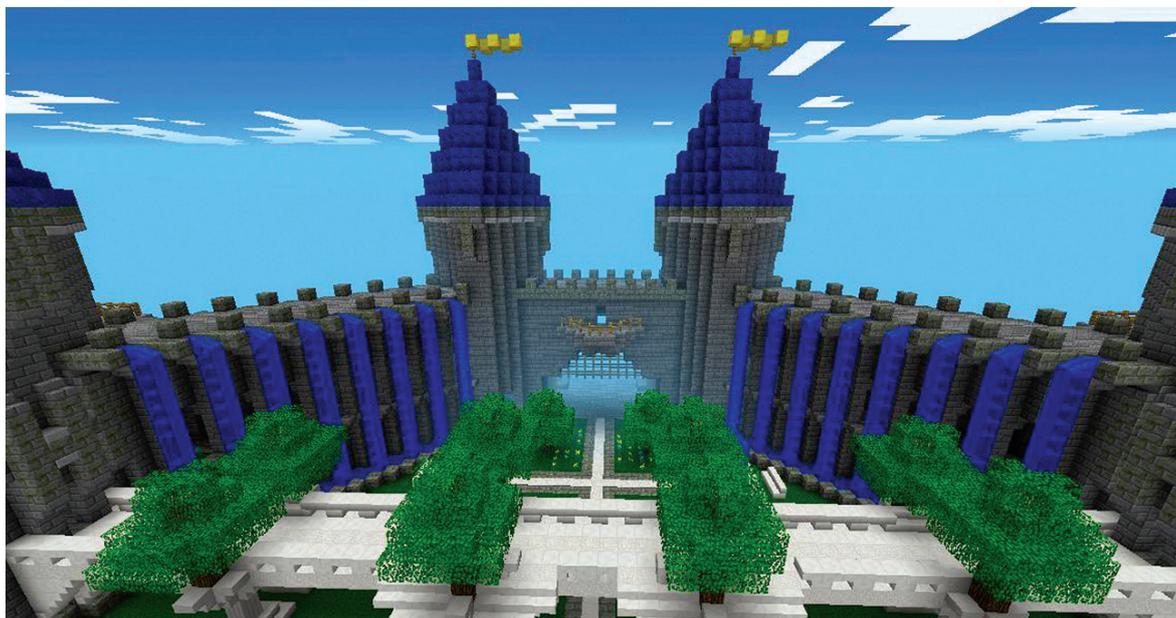
Клавиша	Действие
W	Вперед
A	Налево
S	Назад
D	Направо
E	Инвентарь
Пробел	Прыжок
Двойной пробел	Полет/Падение
Esc	Пауза / Игровое меню
Tab	Использовать курсор мыши

С помощью мыши можно смотреть по сторонам. Мышь также выбираются предметы из инвентаря. По умолчанию вам выдается меч. Чтобы уничтожить блоки, щелкните по ним. С помощью меча можно еще и копать. Выбрав блок из своего инвентаря,



### Скорая помощь

Нажмите двойной пробел, чтобы летать по своему миру. Удерживайте его, чтобы подниматься выше (подобно Икару). Дважды нажмите его снова, чтобы перестать летать и снова стать рабом лучшего друга мысли, силы тяжести.



▶ Немного времени, терпения и кода для автоматизации размещения блоков — и ваш личный замок готов.



вы можете начать строить. Добыть блок можно правым щелчком, а уничтожить — левым.

## Кодируем блоки

Чтобы приступить к созданию кода в *Minecraft: Pi Edition*, нажмите клавишу Tab во время запуска игры. Перейдите в Меню > Программирование. Нажмите Python 3. Постарайтесь разместить текстовые поля так, чтобы они были рядом друг с другом.

Здесь вы можете вводить команды, меняющие игровой мир. Со временем вы также научитесь писать сценарии для автоматизации задач, например, такой, который будет размещать каменный блок везде, где вы идете. Итак, без долгих предисловий, давайте перепрограммируем игру, чтобы отобразить простое сообщение.

Введите следующие команды, нажимая после каждой из них Enter.

```
from mcpi.minecraft import Minecraft
mc = Minecraft.create()
mc.postToChat("Привет всем, я играю в Minecraft Pi!!!")
```

Нажмите Enter, чтобы отобразить ваше первое внутриигровое сообщение. Затем посмотрите на верхний левый экран *Minecraft Pi*. Там показаны координаты X, Y и Z. Это отличный способ определить свое местоположение, а также точно разместить блоки.

С помощью Python можно получить любое количество блоков из различных материалов. Сперва давайте попробуем поместить один блок золота сразу за собой. Введите такую команду:

```
x, y, z = mc.player.getPos()
```

Каждый раз, когда вы двигаетесь, вам надо будет выполнять эту команду, чтобы дать игре обновленные координаты. Нажмите Enter, затем введите

```
mc.setBlock(x+1, y, z, 41)
```

Число 41 в конце последней команды представляет собой инвентарный номер золота. Каждый блок или элемент в игре имеет уникальный идентификатор. Если вы знаете идентификатор определенного типа блока, назначьте ему имя, чтобы его было легче запомнить. Например:

```
gold=41
```

Нажмите Enter, затем запустите команды для создания гигантского куба из литого золота 10×10×10:

```
x, y, z = mc.player.getPos()
mc.setBlocks(x+1, y+1, z+1, x+11, y+11, z+11, gold)
```

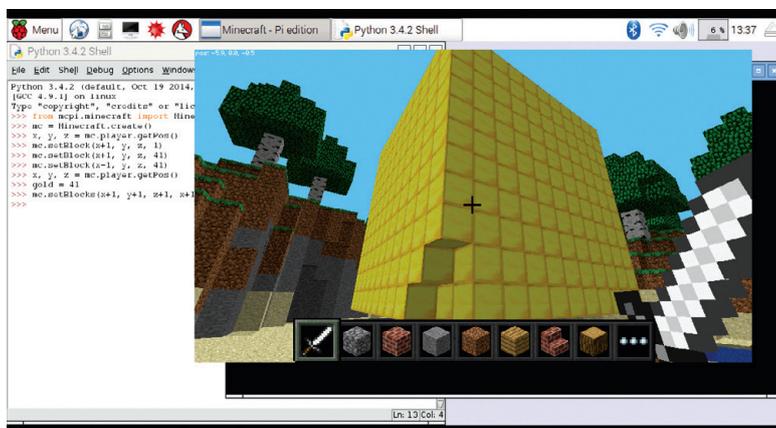
Это лишь некоторые из команд, запускаемых на *Minecraft Pi*. Также можно создавать сложные формы, телепортировать игрока и даже отбрасывать блоки при движении. Чтобы получить полный список команд *Minecraft Pi*, а также инвентарные номера для всех типов блоков, перейдите на сайт [www.stuffaboutcode.com/p/minecraft-api-reference.html](http://www.stuffaboutcode.com/p/minecraft-api-reference.html).

## В обход минных полей

Как уже было сказано, *Minecraft Pi* очень похож на Pocket-версию *Minecraft*, в которую играют на мобильных устройствах, таких как смартфоны на Android и iPhone. Взаимодействие с другими игроками проходит через официальный клиент *Minecraft*. Согласно

### Скорая помощь

Если вам нужно больше дисковой памяти, *Minecraft* может работать на внешнем диске. Просто закройте игру и перетащите папку *Minecraft* на диск.



➤ Если вы устанете, как Мидас, превращать всё в золото, попробуйте повторить команду куба с другим идентификатором блока, изменив стройматериал.

## Установка полной версии Minecraft на Pi 3

Откройте терминал на своем Raspberry Pi и начните с установки ряда инструментов:

```
sudo apt-get -y install xcompmgr libgl1-mesa-dri &&
sudo apt-get -y install libalut0 libalut-dev && sudo apt-get -y install mesa-utils
```

Затем надо включить экспериментальный графический драйвер. Запустите

```
sudo raspi-config
```

и прокрутите вниз до дополнительных опций. Нажмите Enter. Затем выберите опцию драйвера AB-GL и активируйте ее.

Теперь загрузите официальный клиент *Minecraft*, вот так:

```
mkdir ~/Minecraft; mkdir ~/Minecraft/Natives;
cd ~/Minecraft && wget https://s3.amazonaws.com/Minecraft.Download/launcher/Minecraft.jar
```

Запустите его, выполнив

```
$ java -jar Minecraft.jar
```

Теперь вам надо войти в систему со своим именем пользователя и паролем. Затем перейдите на вкладку Редактор профиля и щелкните по полю, расположенному под Версией, чтобы открыть новое окно. Найдите раскрывающееся меню с именем

Использовать версию и выберите 1.8.9. Затем нажмите Сохранить профиль.

Нажмите кнопку Играть внизу, чтобы начать загрузку файлов. По завершении следует закрыть панель запуска и снова открыть терминал. Выполните такие команды:

```
$ cd ~/Minecraft/Natives && wget https://www.dropbox.com/s/4oxcvz3ky7a3x6f/liblwjgl.so
```

```
$ wget https://www.dropbox.com/s/m0r8e01jg2og36z/libopenal.so
```

```
$ cd /home/pi/.minecraft/libraries/org/lwjgl/lwjgl/lwjgl/2.9.4-nightly-20150209 && rm lwjgl-2.9.4-nightly-20150209.jar
```

```
$ wget https://www.dropbox.com/s/mj15sz3bub4dmr6/lwjgl-2.9.4-nightly-20150209.Jar
```

```
$ cd ~/Minecraft/
```

```
$ wget https://www.dropbox.com/s/khr58apwa7pt1w/run.sh
```

```
$ sudo chmod +x run.sh
```

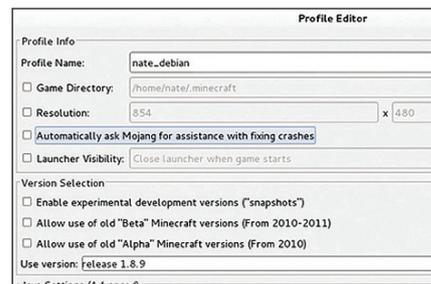
Далее вам необходимо внести изменения в один из загружаемых нами файлов, следующим образом:

```
$ sudo nano ~/Minecraft/run.sh
```

Прокрутите стрелкой вниз, и после введите личные данные *Minecraft*: адрес электронной почты, имя пользователя и пароль. Закончив, нажмите Ctrl+X, затем Y и Enter, чтобы подтвердить изменения.

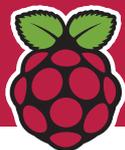
Для запуска *Minecraft* в любое время, введите команду:

```
cd ~/Minecraft && ./run.sh
```



➤ В раскрывающемся меню Выбрать версию выберите 1.8.9. Она не самая новая, но лучше всего подходит для Raspberry Pi.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)



## Скорая помощь



Строки `\u00A74` и `\u00A7l` перед самим приветствием определяют просто цвет и форматирование текста. Полный же список цветовых кодов см. по адресу [http://minecraft.gamepedia.com/Formatting\\_codes#Color\\_codes](http://minecraft.gamepedia.com/Formatting_codes#Color_codes).

отчета на сайте Mojang, клиент довольно давно не обновлялся, но если вы хотите просто посоздать структуры или поучиться кодировать, без каких-либо расширенных функций полноценного клиента *Minecraft*, то это идеальный вариант.

Хотя, как мы уже убедились, на Raspberry Pi 3 можно запустить полную версию *Minecraft* 1.8.9 (см. врезку на стр. 97 внизу), Mojang этот метод официально не поддерживает. Это означает, что производительность на обычной настольной машине будет ниже, то есть иногда могут возникать помехи и сбои.

Экспериментальный графический драйвер, который должен быть включен на Raspberry Pi для использования полной версии *Minecraft*, останавливает работу официального клиента *Minecraft Pi*, но вы можете снова отключить его, запустив `sudo raspi-config` и перезагрузив компьютер.

При доступе к Pi через VNC все версии *Minecraft* идут неважно, поэтому лучше всего играть в нее напрямую.

Если вы хотите использовать полную версию *Minecraft*, надо купить учетную запись в Mojang (на данный момент это стоит £17,95). Если вы уже подписели на *Minecraft* и хотите изучить ее дальше, читайте подробную информацию о том, как настроить себе сервер *Minecraft* на Pi и создать свой цифровой мир блок за блоком.

## Сервер Minecraft

Мы обсудили, как играть в *Minecraft* непосредственно на Pi, отчасти для удовольствия, а отчасти в качестве введения в программирование с Python. Дальнейшая часть учебника касается установки программного обеспечения сервера *Minecraft* на Raspberry Pi 3. Это позволяет другим пользователям клиента *Minecraft* на своих компьютерах подключаться к вашему собственному онлайн-миру и играть без воров и вредителей.

*Minecraft* работает на Java, в последней версии Raspbian предустановленном, а значит, настроить сервер стало проще. Требования у сервера довольно низкие, но для лучшей производительности рекомендуем использовать Raspberry Pi 2 или 3. Имея в своем распоряжении Pi и последнюю версию Raspbian, подключитесь к нему через SSH. Мы объясняли, как это делается, в **LXF224**. Обратите внимание, что по умолчанию имя пользователя и пароль — **raspberrypi** и **pi** соответственно. Запишите свой IP-адрес Pi, потому что он вам понадобится позже.

Затем создайте каталог для *Minecraft* и откройте его командой `$ mkdir minecraft && cd minecraft`

Скачайте последнюю версию *SpigotMC*, глубоко настраиваемую и легковесную версию *Minecraft Server*.

```
$ wget https://hub.spigotmc.org/jenkins/job/BuildTools/lastSuccessfulBuild/artifact/target/BuildTools.jar
```

Далее велите Pi собрать инструменты сервера:

```
$ java -jar BuildTools.jar
```

По завершении процесса вы увидите сообщение Сохранено как **spigot-1.x.x.jar**, где 1.x.x — текущий номер загруженной версии *Spigot* (на момент написания статьи — 1.11.2). Затем запустите сервер *Minecraft*, с помощью

```
$ java -jar -Xms512M -Xmx1008M spigot-1.x.x.jar nogui
```

Опять же, замените 1.x.x номером версии *Spigot*. Программа сообщит, что вам надо согласиться с EULA, чтобы запустить сервер. Для этого введите

```
$ nano eula.txt
```

Прокрутите вниз клавишами курсора и удалите слово **false**. Замените его на **TRUE**. Нажмите Ctrl+X, затем Y и Enter, чтобы сохранить изменения.

Запустите серверное ПО той же командой, что и раньше:

```
$ java -jar -Xms512M -Xmx1008M spigot-1.x.x.jar nogui
```

Появится сообщение **Загрузка библиотек, подождите...** и основная информация об игре. Обратите внимание, что игровой режим по умолчанию — **Выживание**, а область генерации медленно загружается в процентах. Возможно, в ходе генерации различных уровней игра несколько раз перезапустится. По завершении вы увидите сообщение **Готово!**

Настала пора протестировать ваш сервер. Перейдите на компьютер с установленным клиентом *Minecraft* и запустите программу. На главном экране выберите режим **Мультиплеер**. Затем нажмите **Добавить сервер**. Щелкните на поле **Имя сервера** и введите имя по вашему выбору — например, **Pi Minecraft Server**. Нажмите **Адрес сервера** и введите IP-адрес своего Pi. Закончив, нажмите **Готово**. Теперь вы должны увидеть список серверов *Minecraft*.

Поводите над ним указателем мыши и нажмите синюю кнопку **Играть**. Когда игра загрузится, можете приступать к освоению территории и даже выместить свой гнев, разбив несколько блоков, для гарантии, что игра чутко реагирует. Убедившись, что вас всё устраивает, закройте окно *Minecraft* и вернитесь к своему SSH-клиенту, который всё еще подключен к Pi. Введите **stop**, чтобы пока остановить сервер.

Остановить сервер командой **stop** можно в любое время, например, если надо внести какие-либо изменения. В будущем, если вы хотите запустить серверное ПО, подключитесь к Pi через SSH, а затем используйте следующую команду:

```
$ cd ~/minecraft && java -jar -Xms512M -Xmx1008M spigot-1.x.x.jar nogui
```

Снова замените 1.x.x фактическим номером установленной версии *Spigot*. Если вы в ней не уверены, введите команду `cd ~/minecraft && ls`, чтобы просмотреть содержимое папки **minecraft** — там вы можете увидеть точную версию **spigot-1.x.x.jar**.

## Простейшие настройки

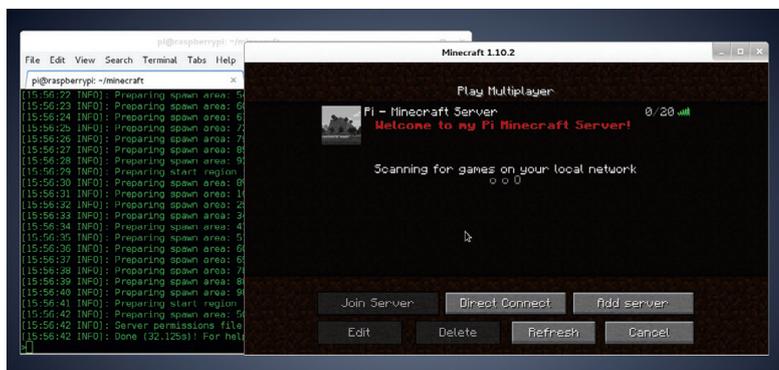
Запустив сервер и подготовив его к работе, не худо узнать, как настраиваются некоторые параметры. Основные настройки содержатся в файле с именем **server.properties**. Вы можете просмотреть и отредактировать этот файл, выполнив команду

```
$ cd ~/minecraft && nano server.properties
```

Если вы намерены произвести максимально точную настройку мира *Minecraft*, посетите <http://minecraft.gamepedia.com/server.properties>, чтобы узнать все значения всех потенциальных настроек. Пока что мы внесем только простейшие изменения. С помощью клавиш курсора прокрутите до самой нижней части файла — до значения **motd=**. Это просто сообщение, которое появляется при первом подключении к серверу *Minecraft*. По умолчанию это **Новый сервер Minecraft**.

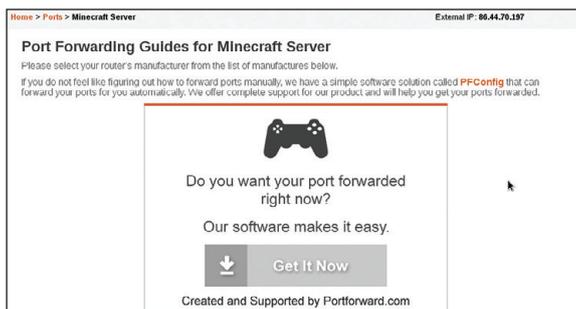
Удалите эту строку и замените следующим сообщением:

```
motd=\u00A74 \u00A7l Welcome to my Pi Minecraft Server!
```



➤ Многопользовательский экран *Minecraft* на Debian Linux. Приветственное сообщение было обновлено (см. основной текст).

➤ Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.



➤ На сайте Port Forward содержатся инструкции по настройке перенаправления портов для большинства моделей роутеров.

Также, в особенности для тех, кто в *Minecraft* пока новичок, будет полезно изменить режим игры с Выживания [Survival] на Творческий [Creative]. Творческий режим позволяет использовать неограниченное количество строительных материалов. Кроме того, игроки не испытывают голода и не расходуют энергию, так что могут сразу начать создавать собственные объекты.

Чтобы изменить режим, прокрутите список до `gamemode=0` и измените значение 0 на 1. Также не забудьте поменять значение `force-gamemode=false` на `true` — тогда все игроки, которые будут впрямь присоединяться к игре, тоже окажутся в Творческом режиме.

Если вы решили остаться в режиме выживания, но хотите разобраться с ним постепенно, стоит заменить `spawn-monsters=true` на `false`. Кроме того, можно изменить сложность игры, подставив вместо `difficulty=1` значение от 0 до 3, где нулевой уровень будет самым простым.

Удовлетворившись внесенными изменениями, жмите Ctrl+X, затем Y, затем Enter, чтобы сохранить изменения и загрузить сервер.

## Расширенные настройки

По мере вашего освоения в *Minecraft*, или если вы уже опытный игрок, основных параметров конфигурации в `server.properties` вам станет не хватать. Тут-то *SpigotMC* и проявит себя во всей красе, вместе с расширенным файлом конфигурации. Здесь предусмотрен целый ряд настроек, способных обогатить ваш опыт игры в *Minecraft*. Просто остановите сервер, а затем выполните следующую команду:

```
$ cd ~/minecraft && nano spigot.yml
```

В качестве примера возможностей настройки *Spigot*, прокрутите вниз до места, где сказано `zombie-aggressive-towards-villager:true`, и измените его на `false`. Этот параметр определяет, будут ли зомби норовить убить жителей деревни *Minecraft*. «Дружелюбные» зомби требуют меньше ресурсов ИИ, что может ускорить работу вашего сервера.

Полный список различных параметров, которые можно изменить в *Spigot*, а также полное описание их действия можно найти на [www.spigotmc.org/wiki/spigot-configuration](http://www.spigotmc.org/wiki/spigot-configuration).

## Замедление работы сервера

Теперь система настроена по вашему вкусу, однако по мере присоединения других игроков и создания более сложных структур вы можете обнаружить, что *Minecraft* начинает тормозить. Как уже говорилось ранее, один из способов добыть свободное место — это переместить папку *Minecraft* на внешний диск. Синтаксис команды (где MYUSB — это имя вашего USB-накопителя) —

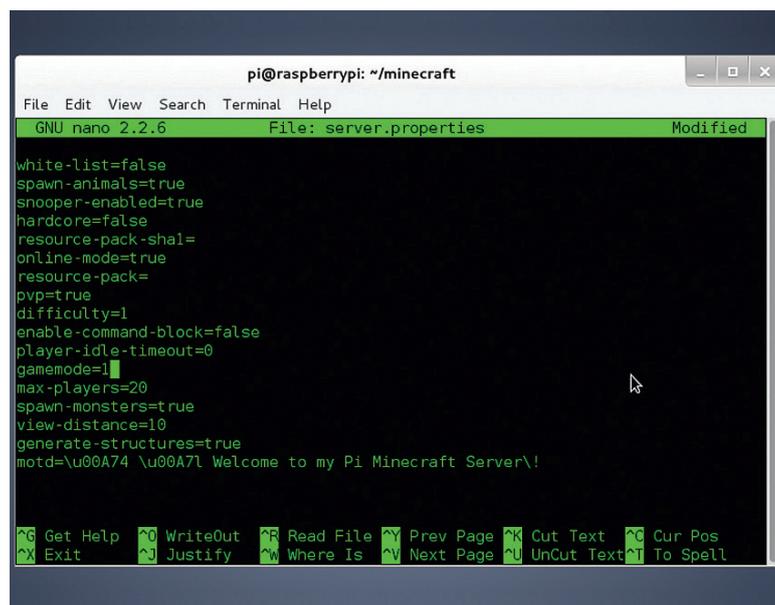
```
$ sudo mv ~/minecraft /media/pi/MYUSB
```

Имейте в виду, что приведенные выше команды для запуска ПО включают команду `cd`, которая ссылается на каталог `minecraft` в домашней папке, поэтому обязательно измените соответствующие команды — например, вместо `cd ~/minecraft` будет `cd /media/pi/MYUSB/minecraft`.

Если проблема не исчезла, попробуйте запустить сервер следующей командой, которая (не вдаваясь в технические подробности) инструктирует Java, как лучше использовать ваш процессор:

```
$ cd ~/minecraft && java -jar -Xms512M -Xmx1008M
-XX:+UseConcMarkSweepGC -XX:+UseParNewGC
-XX:+CMSIncrementalPacing -XX:ParallelGCThreads=2
-XX:+AggressiveOpts -jar spigot-1.x.x.jar nogui
```

Чтобы производительность еще улучшилась, попробуйте повторно отредактировать файл `server.properties`, на сей раз уменьшив расстояние видимости с 10 до 5. Оно определяет, какая часть карты будет загружаться при движении. Можно также изменить параметр `max-players`, ограничив число пользователей, способных войти в систему одновременно. **LXF**



➤ Основные настройки игры находятся здесь. Для начала лучше всего подойдет Творческий режим, но есть также Выживание, Приключение и даже Хардкор.

## Minecraft в глобальном пространстве

На нашем уроке рассказывается, как настроить *Minecraft* в вашей локальной сети, чтобы кто угодно мог подключиться к ней и играть вместе с вами. Однако вы можете разрешить другим пользователям присоединиться к вашему серверу через Интернет.

Сперва надо присвоить своему Pi постоянный IP-адрес в вашей сети, отредактировав `/etc/network/interfaces`. Затем используйте функцию, известную как переадресация (проброс) портов на вашем

маршрутизаторе, чтобы перенаправлять входящий трафик из внешнего Интернета на сам Pi.

Поскольку у каждого маршрутизатора своя специфика, лучше всего узнать, как это точно делается, у вашего производителя, либо привлечь кого-то более подкованного в технике. В любом случае скажите, что вам надо открыть порт 25565 для TCP и UDP. Кроме того, вам требуется внешний IP-адрес вашего маршрутизатора, который вы можете узнать, посетив [www.whatismyip.com](http://www.whatismyip.com).

На сайте Port Forward есть несколько руководств по настройке перенаправления портов: [http://portforward.com/english/applications/port\\_forwarding/Minecraft\\_Server](http://portforward.com/english/applications/port_forwarding/Minecraft_Server).

Ваш интернет-провайдер может периодически менять IP-адрес вашего роутера, из-за чего у других игроков способны возникнуть сложности. Вы можете заплатить провайдеру за аренду статического IP-адреса или получить адрес бесплатно через какой-нибудь сервис, например, [www.noip.com](http://www.noip.com).



# LXF HotPicks



Александр Толстой

Otter Browser » Simon » Persepolis » Elisa » Nylas Mail » CMST » Redshift » Libva » GNU Emacs » OpenTTD » Warzone 2100

## Браузер

# Otter Browser

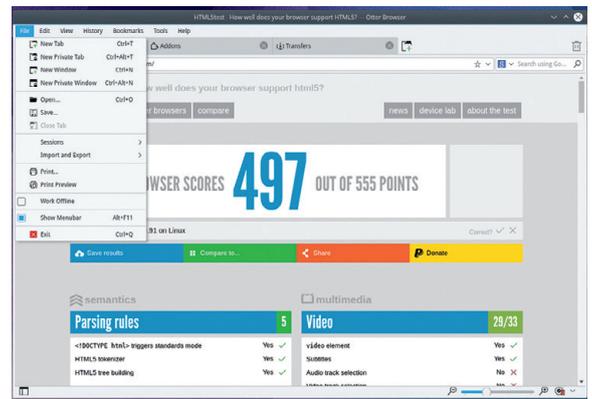
Версия: 0.9.91 Сайт: <https://otter-browser.org>

Хотя относительно недавно вышли новые релизы *Firefox* и *Chromium*, мы решили пропустить эти и без нас известные и не самые новые браузеры ради свежей альтернативы. Исторические корни *Otter Browser* были важным моментом в 2012 г. и относились к решению Opera Software отказаться от своего движка *Presto*. *Otter Browser* появился, чтобы воссоздать «атмосферу» исходного браузера *Opera 12.x* на *WebKit* и *Qt5*.

Нам понравились ранние версии *Otter Browser* (см. LXF191), но, честно говоря, на тот момент он еще не был готов для повседневного использования. Новый взгляд на проект спустя два с половиной года

обнаружил титанический прогресс, сделанный разработчиком *Otter*.

Не мешает заглядывать на сайт проекта по вторникам, чтобы познакомиться с обновленным еженедельным отчетом о планах, отладках и оставшихся проблемах, и найти двоичные версии *Otter Browser* для всех основных ОС. Годами аккуратно поддерживаемый блог еженедельно сообщает об обновлении состояния. «Тихо, тихо



» Лучшие результаты тестирования означают меньше проблем на многих современных сайтах и сайтах с тяжеловесным контентом.

## В нем масса улучшений по удобству, некогда бывших в Opera 12.

ползи, улитка, по склону Фудзи — вверх, до самых высот!»

Вы всё правильно поняли: сейчас *Otter Browser* намного лучше, чем был. Мы устроили ему тест-драйв на пару дней и не обнаружили особого количества ошибок. С движком по умолчанию *QtWebKit* работа с браузером не идеальна, но вы можете переключиться на более новый *QtWebEngine* (технически это просто урезанный движок *Chromium*), и работа с современными и более тяжеловесными сайтами сразу улучшится.

В *Otter Browser* также масса улучшений по удобству в использовании, некогда бывших в ныне оставленном *Opera 12*. Закладки, заметки, cookie, пароли, списки блокировки рекламы, а также журнал и закладки имеют отличные удобные разделы управления, очень простые в использовании. Браузер предлагает традиционные древовидные настройки Web в *About:config* и отдельный интерфейс *Tools > General Settings* для тонкой настройки самого браузера.

Легковесный, продуктивный и полнофункциональный браузер *Otter Browser* выглядит многообещающе и поладит с любым рабочим столом. Функция синхронизации со стороны сервера отсутствует — потому что это вы должны управлять браузером, а не наоборот.

## Исследуем интерфейс Otter Browser

### Вкладки и кнопки

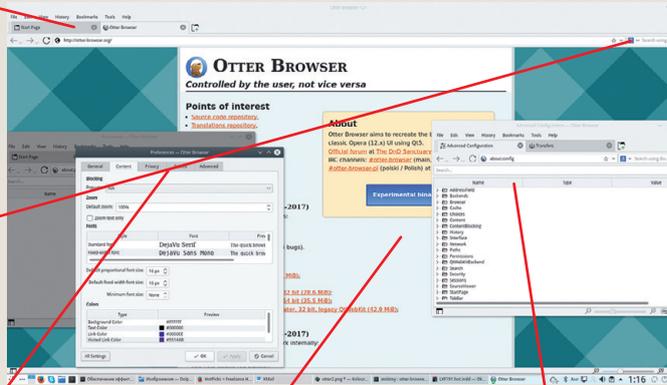
Открывайте сайты в новых вкладках и осуществляйте навигацию с помощью удобно расположенных кнопок.

### Выберите поисковик

*Otter* использует по умолчанию *DuckDuckGo*, но вы можете изменить это на свое усмотрение.

### Личные предпочтения браузера

Шрифты, настройки конфиденциальности и множество расширенных параметров находятся здесь.



### Ждите лучшего рендеринга

Хотя *Otter* является нишевым браузером, он храбро перемалывает мегабайты web-страниц с JavaScript.

### Расширенная настройка

Измените движок рендеринга здесь, а также точно настройте работу браузера.

## Музыкальный плеер

## Elisa

Версия: GIT Сайт: <https://github.com/KDE/elisa>

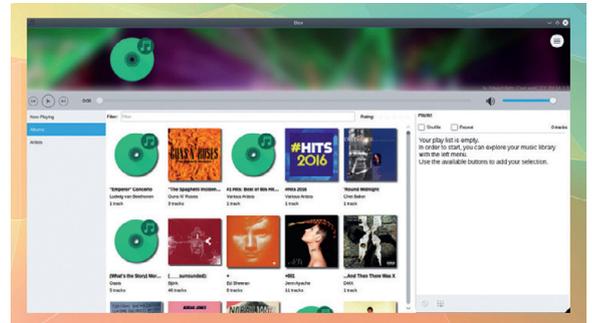
**М**ы питаем слабость к тем, кто дружит с современным рабочим столом KDE Plasma. Музыкальных плееров, ориентированных на KDE и Qt, великое множество — например, *Amarok*, *Juk* и *Clementine*, но худо ли иметь выбор, особенно если он показывает весьма сильную активность разработки? Разработчик KDE Маттье Гайен [Matthieu Gallien] недавно выпустил *Elisa*, музыкальный плеер, соответствующий принципам визуального стиля Plasma, используя в нем уже знакомые технологии, такие как Valoo для индексации и QML для общего интерфейса.

Проект пока что очень юн и не так богат функциями, как перечисленные нами плееры. Логичнее сравнивать *Elisa* с *Babe* — минималистским музыкальным плеером на Qt5, о котором мы рассказали в LXF222. Два этих плеера созданы с абсолютно разными подходами: *Babe* пытается упаковать как можно больше функций в суперкомпактный GUI с крошечными

кнопками, а *Elisa* показывает более крупные кнопки управления и специальные области, чтобы вы чувствовали себя комфортнее. Мы считаем, что предпочтения у всех свои, но должны признать, что в *Elisa* имеются определенные преимущества. Во-первых, он автоматически определяет все воспроизводимые пункты в вашей директории `~/Music`, что очень удобно и совсем не мешает. Вам ничего не надо делать, чтобы при первом запуске импортировать свою музыкальную библиотеку: *Elisa* позаботится об этом самостоятельно. Еще одна приятная функция — интерфейс *Elisa*, на QML, благодаря которому плеер выглядит естественно на любом устройстве.

В остальном *Elisa* придерживается принципов простоты и легкости в использова-

**Интерфейс на QML  
выглядит естественно  
на любом устройстве.**



► *Elisa* автоматически индексирует всё в `~/Music`.

нии. Он предлагает категории Now Playing [Сейчас воспроизводится], Albums [Альбомы] и Artists [Исполнители] слева, область контента категории в середине и область воспроизведения справа. В *Elisa* вообще нет опций или настроек, так что это, возможно, плеер с самым простым GUI, который мы вообще видели. Загрузите систему с пакетом зависимостей KF5 и смело компилируйте этот музыкальный плеер из исходника, потому что пакетов или мало, или вообще нет.

## Программа распознавания речи

## Simon

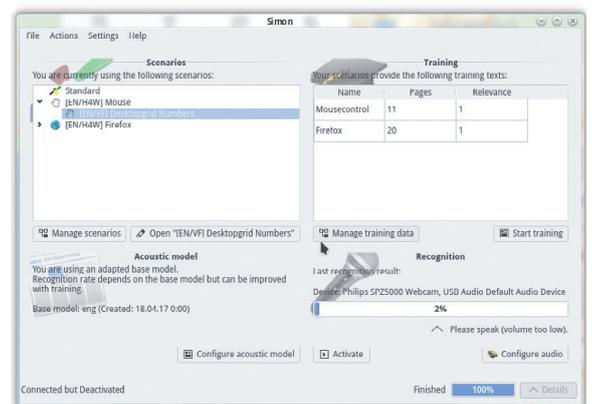
Версия: 0.4.1 Сайт: <https://simon.kde.org>

**П**ервые попытки превратить текст в речь и, наоборот, речь в текст, предпринимались уже в середине 1990-х, когда ПК обзавелись функциями мультимедиа. Но даже сегодня возможность голосового управления системой по-прежнему кажется футуристичной, или, по крайней мере, редко используется. Уж не знаем, в сложности ли тут дело; однако благодаря *Simon* всё становится вполне доступным для обычного пользователя.

*Simon* — система распознавания речи с открытым кодом, пригодная для развлекательных проектов, но также и для тех случаев, когда вы в силу неких причин не можете использовать мышь и клавиатуру для ввода. *Simon* использует библиотеки KDE, CMU SPHINX *w/или Julius*, вместе с HTK, и работает в Windows и Linux. Приложение предлагает приятный GUI на Qt4 и множество средств, позволяющих сделать сложную задачу распознавания речи не такой страшной для новичка. Мы употребили слово

«доступный», а не «простой», сознательно, потому что вам всё же придется проделать некую подготовительную работу, прежде чем *Simon*, в свою очередь, сможет хоть что-то сделать. Две основные вещи, которые вы должны подготовить для *Simon* — это речевая (или акустическая) модель и сценарий. *Simon* умеет применять два распознавателя речи — *Julius* и *PocketSphinx*; в большинстве дистрибутивов Linux предусмотрены оба. Настроив распознаватель, вы получаете только статическую модель речи, что вряд ли можно считать оптимальным, поэтому вы, возможно, решите адаптировать модель под свой голос. Когда вы установите пакет *SphinxTrain* и обеспечите набор языковых правил, известный как

**Чтобы все заработало,  
нужно время, но результаты ошеломляют!**



► Функция заказа пиццы с помощью голосового контроля планируется на следующий релиз *Simon* (мы очень надеемся!).

Hidden Markov Model Toolkit (HTK), для создания речевой модели, *Simon* позволит вам обучить свою модель. Затем вам надо соединить модель со сценарием, чтобы *Simon* мог реально что-то контролировать на вашем рабочем столе. И модель, и сценарий можно найти с помощью функции Get New Stuff, которая соединяет *Simon* с <http://store.kde.org>, или поищите имеющиеся модели вручную на [www.voxforge.org](http://www.voxforge.org).

Чтобы всё заработало, нужно некоторое время и усилия, надо добавить в модель настоящие образцы вашего голоса и, наконец, использовать сценарий для управления мышью, браузером и т. д., но результаты просто ошеломляют!

## Менеджер загрузки

## Persepolis

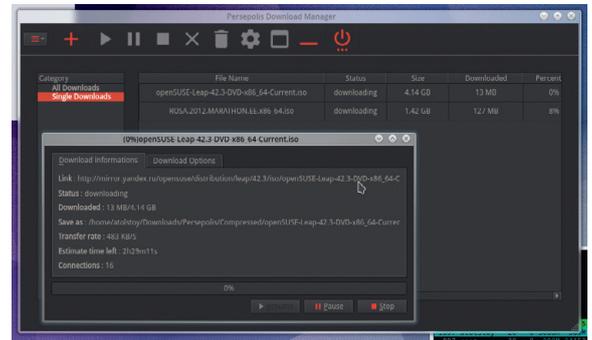
Версия: 2.4.2 Сайт: <https://github.com/persepolisdm/>

Пока вы не задали традиционный вопрос типа «Зачем мне еще один менеджер загрузки для рабочего стола Linux в 2017 г.?», спешим упомянуть, что *Persepolis* — не просто очередной элемент кучи аналогов, а расширенный GUI для *Aria2*. Чтобы было понятно, *Aria2* — это альтернатива *Wget*, которую вы, вероятно, знаете, как команду консоли для скачивания любого файла из интернет. *Aria2* был создан Тацухиро Цуджикава [Tatsuhiko Tsujikawa] почти десять лет назад в качестве расширенного менеджера загрузки и скоро обрел поддержку множества протоколов, таких как HTTP, HTTPS, FTP, BitTorrent и Metalink. Среди наиболее ярких функций *Aria2* — возможность скачивать несколько файлов внутри одного экземпляра команды и грамотное использование доступной ширины канала. Как видите, *Aria2* является не только программой скачивания, но также менеджером загрузки и клиентом BitTorrent. Неудивительно,

что он победил в Сравнении *Linux Format* в 2008 г. (LXF108), и мы подозреваем, что сегодня у него тоже были бы неплохие шансы. Ирония в том, что в *Aria2* издавна имел интерфейс на древнем *GTK1*, который потом ликвидировали в пользу чисто командной строки, и с тех пор каких-либо интересных графических интерфейсов для *Aria2* не появлялось.

*Persepolis* вышел, чтобы это исправить, и предлагает вам профессионального вида программу на Python и *Qt5*. У нее приятная и опрятная структура со списком закачек в центре окна и панелью быстрого действия наверху. В простейшем сценарии всё, что вам придется делать в *Persepolis* — скопировать ссылку на файл, который вы хотите

**Все, что вам придется делать — скопировать ссылку на файл.**



Мы выбросили *Wget* ради того, чтобы использовать этот роскошный темно-красный интерфейс *Aria2*.

скачать, и нажать на кнопку +. Как и следует ожидать, всплывающее окно автоматически вставит вашу ссылку в нужное поле, и вам останется только подтвердить свои намерения, нажав OK.

*Persepolis* открывает всю внутреннюю мощь *Aria2* и позволяет вам наслаждаться многосегментным скачиванием, планированием задач и поддержкой торрентов. Его сайт предлагает инструкции по установке *Persepolis* на Ubuntu, Fedora, Arch и openSUSE.

## Почтовый клиент

## Nylas Mail

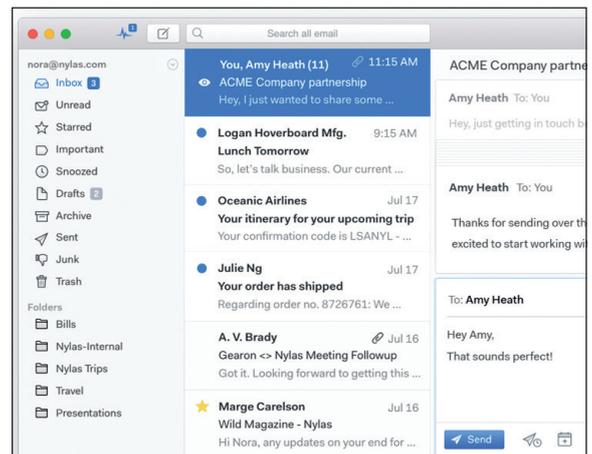
Версия: 2.0 Сайт: <https://github.com/nylas/nylas-mail>

Почтовый клиент нового поколения, известный как *Nylas N1*, предложил свежий подход к работе с электронной почтой посредством установки специального прокси (*Nylas Sync Server*) между сервером IMAP и графическим клиентом на стороне пользователя. В последнее время *Nylas N1* находился в активной разработке как часть бизнес-проекта калифорнийской команды. На первом этапе команда предоставила всем бесплатный доступ к их домашнему *Sync Server*, но позже они сделали его платной функцией для предприятий. Чтобы *N1* работал на домашнем настольном ПК, надо развернуть свой экземпляр *Sync Server* и затем подключить к нему клиент. Честно говоря, для среднестатистического админа локального хоста это многовато. Однако появилась хорошая новость: вышло приложение под немного измененным названием, *Nylas Mail*, без всяких прокси и без всего лишнего. Новый *Nylas Mail 2.0* может подключаться к серверам IMAP напрямую, что

позволяет за считанные секунды настроить свою учетную запись Google Mail, Yahoo или iCloud. Впрочем, при желании использовать отдельный движок синхронизации всё равно можно — все части *Nylas* по-прежнему с открытым кодом.

Приложение предлагает много дополнительных плагинов для расширенной фильтрации и классификации входящих сообщений, а также уведомлений, планировщиков, почтового «повтора», статистики сбора, автофильтра и т.д. Некоторые из перечисленных функций несколько месяцев назад были доступны только в коммерческой версии '*Pro*', а сейчас переключались в основное дерево. Есть плагины, особо интересные торговым агентам или аналитикам: скажем, можно отследить,

**Использовать отдельный движок синхронизации можно.**



Вы можете укротить этот инструмент, изначально предназначенный для торговых профессионалов, и заставить его работать на вас.

открыл ли получатель ваше сообщение и проследовал ли на URL из него. Вам также ничего не надо делать, чтобы отслеживали ваши действия. Однажды попробовав *Nylas Mail 2.0* со своей реальной учетной записью, вы можете рассчитывать на внимание директора по развитию команды, которому очень нужна обратная связь по приложению, и поэтому он отправляет вам вопросы, на которые вы не ответили. Мы предпочли использовать наш собственный внутренний сервер синхронизации, чтобы этого избежать.

## Менеджер подключения

## CMST

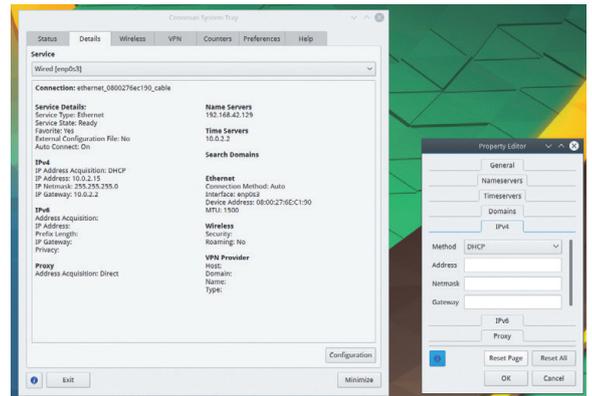
Версия: GIT Сайт: <https://github.com/andrew-bibb/>

За этими четырьмя буквами стоит «графический интерфейс для ConnMan со значком в системном лотке [ConnMan with System Tray icon]». A ConnMan — это менеджер сетевого подключения, способный заставить вас задуматься о его месте в мире, где властвует Network-Manager (NM). Оба выполняют ту же работу, и если ваш Linux использует NM, нет смысла это менять. Но главное в Linux — свобода выбора, особенно если вы хотите создать индивидуальный супер-занятый рабочий стол с десятками компактных и надежных программных компонентов. ConnMan был разработан в Центре Технологий с Открытым Кодом Intel Марселем Хольтманном [Marcel Holtmann], тем же человеком, который написал BlueZ. Хотя в ConnMan меньше функций, чем в NM, он умен, прост и отлично подходит для мобильных решений, встроены устройств или — если хотите — легковых рабочих столов Linux. Сейчас обновления для ConnMan

выходят не так часто, зато они есть для его основанного на Qt5 интерфейса под названием CMST. Мы скомпилировали свежую копию из репозитория Git проекта и обнаружили этот приятный и хорошо организованный менеджер для ConnMan.

CMST работает безотказно, как только вы включите системный сервис ConnMan и отключите NM. Там имеется семь вкладок, и Status отображается по умолчанию. Активировать или оборвать существующее подключение можно одним щелчком мыши по кнопке внутри столбца Powered. Вкладка Details показывает всё, что вы когда-либо требовали от утилиты ifconfig, а то и больше. Мы признаем, что просто видеть информацию и статистику сети недостаточно,

**Приятный и хорошо организованный менеджер для Connman.**



Хорошая альтернатива NetworkManager со множеством дополнений.

поэтому не пропустите кнопку Configuration внизу справа. Здесь вы можете настроить IP, DNS, сетевую маску и прокси, так же, как в настройках NM. Нам особенно понравилась вкладка Preferences с немалым числом функций управления автозапуском, поведением, внешним видом и уведомлениями CMST. Вы даже можете настроить разные опции автозапуска для разных сессий рабочего стола или скрыть значок в системном лотке, чтобы работа CMST оставалась невидимой для пользователя.

## Программа защиты зрения

## Redshift

Версия: 1.11 Сайт: <http://jonls.dk/redshift>

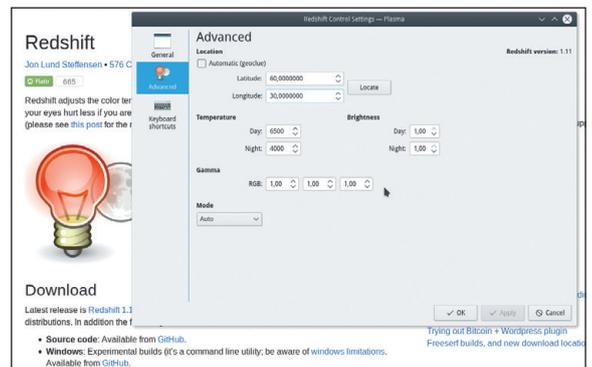
Возможно, вы уже читали о фишках нового релиза Gnome 3.24, который будет блистать в грядущем Fedora 26 (он уже должен выйти, когда вы будете читать эти строки). Мы говорим о фильтре голубого света или, в терминологии Gnome, «ночном свете». Суть в том, чтобы автоматически регулировать цветовую температуру вашего экрана в зависимости от времени суток. По научным данным, более теплые, желтоватые тона намного лучше для глаз, если нам приходится часами сидеть за компьютером в ночное время. Ну, а если вы работаете в другой среде рабочего стола, эта функция доступна для вас через стороннее приложение Redshift.

По сути, у приложения есть две части: демон, выполняющий саму работу, и GUI, чтобы управлять настройками Redshift без перехода в Терминал. Автоматическое исправление гаммы не зависит от физического интерфейса вашего дисплея (т.е. старый кабель VGA — не проблема); вместо этого он использует поддержку RANDR

вашего видеодрайвера. Иными словами, должны поддерживаться абсолютно все системы настольного Linux. Redshift предполагает, что нейтральная цветовая температура — 6500 K, а самый «теплый» ночной режим соответствует 4000 K. Демон рассчитывает показатель для определенного времени суток и настраивает гамму с мягким, практически незаметным переходом. Чтобы угадать время суток, Redshift необходимо знать ваше местонахождение. Вы можете вручную установить широту и долготу или использовать автоматический механизм, который использует Geoclue для определения вашего местонахождения по IP-адресу.

Большинство дистрибутивов Linux предлагают Redshift в своих стандартных

**Более теплые, желтоватые тона лучше для глаз в ночное время.**



Этот отлично настраиваемый настольный инструмент избавит вас от синдрома усталых глаз.

репозиториях. После установки, проверьте работу сервиса с помощью

```
$ sudo systemctl status redshift.service
```

Если Redshift не работает, надо заменить status на enable.

Затем вы можете приступить к настройке программы с помощью GTK-индикатора Redshift или плазмоида Redshift, в зависимости от вашего рабочего стола. Вам незначит ждать, пока стемнеет, чтобы увидеть Redshift в действии — значок в системном лотке позволяет вручную настроить температуру с помощью колеса прокрутки мыши, и вы сразу поймете, как им пользоваться.

## HotGames Развлекательные приложения

## Транспортный симулятор

## OpenTTD

Версия: 1.7 Сайт: [www.openttd.org](http://www.openttd.org)

**В**ряд ли вы обнаружите другую игру с открытым кодом, которую можно было бы с таким же правом назвать истинным убийцей Linux'оидов. *OpenTTD* — преемник *Transport Tycoon Deluxe*, транспортного симулятора начала 1990-х, вот почему он буквально летает даже на очень слабых машинах. Если вам нравится перевозить грузы, оптимизировать логистику поставок и создавать сложные транспортные сети, *OpenTTD* может на недели и даже месяцы вырвать вас из повседневной жизни — она просто бесконечна.

Игра начинается с того, что вы пролагаете маршрут между двумя точками для автобуса, грузовика, поезда, корабля или самолета, чтобы доставлять грузы или пассажиров из одного места в другое и зарабатывать на этом деньги. Она интегрируется с онлайн-репозиторием

с лучшей графикой, звуком, сценариями и игроками ИИ, с которыми вам можно соревноваться. Расширением, о котором стоит упомянуть, является серия NewGRF с дополнительными объектами, чтобы вы получили еще большее удовольствие от *OpenTTD* с лучшей графикой. Объекты NewGRF добавляют здания, вокзалы, транспортные средства и множество способов развлечься. Можно воссоздавать реальные исторические маршруты, выбирая транспортные средства, которые выглядят реалистично и имеют соответствующую скорость, грузоподъемность, срок действия и т. д.

**OpenTTD может вырвать вас из жизни на недели и месяцы.**



► Идеальная городская инфраструктура с пассажирскими и грузовыми маршрутами потребует тщательной многодневной работы.

Каждому игроку разрешается выбрать собственный вектор развития: например, сеть трамвайного движения по городу или железнодорожные маршруты дальнего следования со сложными узлами и настройкой семафоров, или транспортировка нефти с буровой в море до наземного нефтеперерабатывающего завода, или что-то еще — игра работает в режиме одного игрока (*OpenTTD* также поддерживает многопользовательский режим).

Разработка продолжается. Козыри этого релиза — больше спрайтов, реалистичное ускорение и производительность. Linux-сборки уже на сайте *OpenTTD*.

## Стратегия

## Warzone 2100

Версия: 3.2.2 Сайт: <https://wz2100.net>

**П**орой нам надоедают головоломки, и хочется взорвать что-нибудь, особенно на пост-ядерном ландшафте будущего. *Warzone 2100* — идеальная площадка для бесконечных военных действий, в которых вы можете применить свои навыки в тактике и стратегии. Это классическая RTS с видом сверху с высококачественной графикой и звуком, и чтобы в ней выжить, приходится думать.

Классическая кампания «Человек против ИИ» начинается с группы грузовиков и нефтяного бассейна неподалеку. Вам надо построить буровую вышку, электрогенератор и исследовательский центр, который поможет восстанавливать технологии из артефактов, найденных вашими грузовиками на карте. Новые технологии позволяют разрабатывать новые боевые машины и производить их на заводах. Главная цель — произвести больше

боевых машин за меньшее время, чтобы подготовиться к схватке с врагами. Для знакомых с RTS тут нет ничего нового, но таких игр не так уж много для Linux, особенно таких, которые предлагают высококачественную графику, звук и сюжет. Наземные объекты отлично детализированы, движения камеры естественные и ровные, и снова по карте, открывая артефакты — одно удовольствие.

Для новичков *Warzone 2100* предлагает очень хорошее голосовое руководство, оно поможет вам освоить механику за считанные минуты. В реальных миссиях ключ к успеху заключается в быстром строительстве заводов и электростанций,

**Снова по карте, открывая артефакты — удовольствие.**



► Вы можете приблизить изображение, чтобы в деталях рассмотреть свои здания и транспортные средства.

которые помогут вам восстановиться в случае, если враг уничтожит ваши боевые единицы. Следовательно, чтобы победить соперника, вам нужно в первую очередь нацелиться на его инфраструктуру, возможно, даже обходя его основные войска и избегая прямых столкновений с его основными силами. *Warzone 2100* может увлечь вас военной стратегией на долгие часы, и мы счастливы, что обнаружили эту чудесную игру.

## Библиотека видео

## Libva

Версия: 1.8.1 Сайт: <https://github.com/01org/libva>

Для нас становится любопытно, когда мы видим небольшое мультимедийное устройство с низким напряжением и CPU с пассивным охлаждением, которое успешно воспроизводит видео Full HD, не загружая свои ядра по самые уши. Секрет в том, что CPU перекладывает часть своей тяжелой работы на графический чип, и это было бы невозможно без соответствующего программного компонента поддержки аппаратного ускорения.

*Libva* — это графическая библиотека, которая реализует VA-API (Video Acceleration API) и позволяет вам быстро и энерго-эффективно декодировать видео. Разработчики *Libva* непрерывно реализуют современные видеокодеки и улучшают производительность библиотеки. Самое очевидное место размещения *Libva* в течение долгих лет — браузер *Google Chromium*, который активно использует кодеки VP8 и VP9 для обработки онлайн-видео. После того, как YouTube избавился от Flash ради следующего поколения HTML5 и кодеков

VP8/VP9, для достойной производительности онлайн-видео *Libva* стала крайне важна. Кодек VP8 изначально разрабатывался On2 Technologies, и позднее его приобрел Google, чтобы сделать технологией с открытым кодом. С другой стороны, *Libva* поддерживалась Intel в их проекте с открытым кодом Медиа и Аудио для Linux.

Практически все пользователи Linux, которым нравится воспроизводить видео с высоким качеством, вероятно, уже являются пользователями *Libva*, даже не подозревая об этом. Если вы не уверены, что это про вас, проверьте, сильно ли тормозит ваша система при воспроизведении видео 1080p или 4K. *Libva* соединяет ваш видеоплеер с GPU и обеспечивает воспроизведение с аппаратным ускорением. Настройка

## Библиотека Libva соединяет ваш видеоплеер с GPU.

```

atolstoy@atolstoy-desktop ~ $ vainfo
libva info: VA-API version 0.38.0
libva info: va_getDriverName() returns 0
libva info: Trying to open /usr/lib64/dri/i965_drv_video.so
libva info: Found init function __vaDriverInit_0_38
libva info: va_openDriver() returns 0
vainfo: VA-API version: 0.38 (Libva 1.7.3)
vainfo: Driver version: Intel i965 driver for Intel(R) Sandybridge Desktop - 1
vainfo: Supported profile and entrypoints
VAProfileMPEG2Simple      : VAEntrypointVLD
VAProfileMPEG2Main       : VAEntrypointVLD
VAProfileH264ConstrainedBaseline: VAEntrypointVLD
VAProfileH264ConstrainedBaseline: VAEntrypointEncSlice
VAProfileH264Main        : VAEntrypointVLD
VAProfileH264Main        : VAEntrypointEncSlice
VAProfileH264High        : VAEntrypointVLD
VAProfileH264High        : VAEntrypointEncSlice
VAProfileH264StereoHigh  : VAEntrypointVLD
VAProfileVC1Simple       : VAEntrypointVLD
VAProfileVC1Main         : VAEntrypointVLD
VAProfileVC1Advanced     : VAEntrypointVLD
VAProfileNone            : VAEntrypointVideoProc
  
```

► Выясните возможности ускорения видео вашей системы командой `$ vainfo`.

будет разной в зависимости от модели вашего видеочипа. Иными словами, чистая VA-API работает с Intel Graphics, тогда как Nvidia и некоторые современные чипы AMD требуют движка VDPAU. Чтобы вся эта хрупкая система заработала, нужна комбинация видеодрайвера, рабочей реализации *Libva* и правильно настроенного видеоплеера, распознающего ускоренный вывод видео. Проверьте вывод команды `$ vainfo`, чтобы узнать, нужна ли вашей системе тонкая настройка.

## Операционная система

## GNU Emacs

Версия: 25.2 Сайт: [www.gnu.org/software/emacs](http://www.gnu.org/software/emacs)

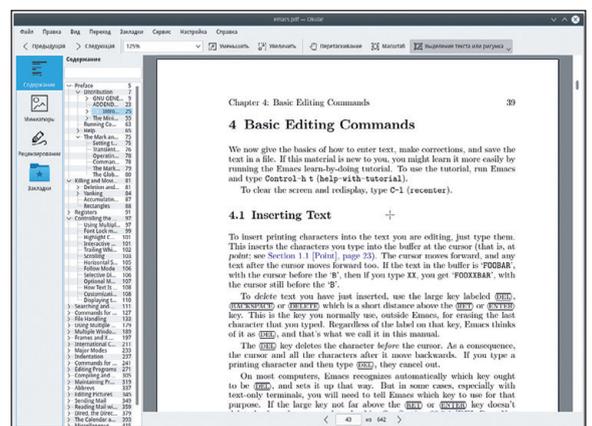
Мы уделяем незаслуженно мало внимания *GNU Emacs*, одному из краеугольных камней свободного ПО. *Emacs* существует уже 41 год — только представьте: когда появилась первая версия *Emacs* (еще не GNU), Линусу Торвалдсу было всего семь лет. Новая версия 25.2 является релизом исправления ошибок для 25.0, а этот последний был первым релизом *Emacs* вне личного контроля Ричарда Столлмена [Richard Stallman] (он отступил в сторону на версии 24.5).

Вызывающий в равной степени любовь и ненависть, *Emacs*, вероятно, мощнее любого другого текстового редактора из когда-либо созданных. В отличие от своего основного соперника, *Vim*, *Emacs* располагает только одним режимом ввода, а дополнительные функции вызываются комбинациями клавиш. На самом деле, многие приложения стремятся предложить совместимые с *Emacs* привязки к клавишам, чтобы вы почувствовали себя как дома

в данном текстовом редакторе, даже если никогда раньше им не пользовались.

Руководство *GNU Emacs* описывает его как «расширяемый, настраиваемый, самодokumentированный редактор отображения в реальном времени» и сообщает, что в нем более 2000 встроенных команд, которые можно комбинировать и использовать для автоматизации обычной работы; поэтому неудивительно, что многие называют его «основанной на символах операционной системой». Невозможно перечислить здесь все практические случаи применения, но *Emacs* умеет практически всё: вести заметки, управлять проектами, планировать события и, конечно, помогать в форматировании и печати документов,

## Emacs, вероятно, мощнее любого другого текстового редактора.



► В справочнике 642 страницы. Угадайте, в чем он был написан!

как текстовый редактор (взаимодействуя с такими внешними программами, как *LaTeX*, *Ghostsript* или *Brzr*).

Мистер Столлмен разрабатывал *GNU Emacs* с упором на производительность, поэтому легко фиксировать нажатия на клавиши и вставлять их в макросы, автоматизируя сложные повторяющиеся задачи. Редактор заглядывает в `~/.emacs`, чтобы найти там макросы и переменные и загрузить расширения. Индивидуальная настройка *GNU Emacs* — совсем другая история (скорее длинная, чем сложная), а наша цель здесь — поощрить применение *Emacs* в работе, если вы этого избегали. LXF

# На диске

Дистрибутивы, приложения, игры, книги и всякое-разное...

Лучшее из Интернета, упакованное в 8 ГБ качественного DVD.



## Дистрибутивы

В этом месяце Gnome-фицированный Ubuntu заставил меня задуматься. Я не особый фанат Ubuntu, но аплодирую их смелому шагу насчет Unity. Когда Ubuntu с рабочим столом Gnome 2 был безусловно самым популярным настольным дистрибутивом, им хватило смелости отвергнуть этот рабочий стол и перейти на нечто новое. Правда, по сути под капотом был Gnome 3, но даже это было большим скачком в сторону от Gnome 2. Gnome 3 на тот момент не считался готовым, и Ubuntu мог застрять на старом рабочем столе еще на один-два релиза, но он так не сделал.

Сказать, что Unity приняли без восторга, будет слишком мягко. Было похоже, что Ubuntu отдаст немалую часть своих пользователей дистрибутивам вроде Linux Mint, но там остались верны своему взгляду на унифицированный интерфейс для всех устройств.

Семь лет спустя понадобилось еще больше мужества, чтобы признать: идея не сработала, и объявить о переходе на более традиционный рабочий стол, хотя за эти годы определение такового сильно изменилось. Я желаю Ubuntu удачи с его новым рабочим столом; и, возможно, вернется наша нормальная ситуация, когда пользователи KDE и Gnome неустанно поносят друг друга...

Neil

## » Важно ВНИМАНИЕ!

### Порченные диски

В маловероятном случае какого-то дефекта вашего LXF DVD обращайтесь, пожалуйста, по адресу [disks@linuxformat.ru](mailto:disks@linuxformat.ru) или телефону +7 (812) 309-0686.

Дистрибутив с пятью рабочими столами на выбор

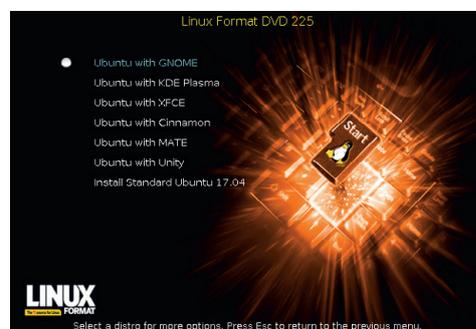
# Ubuntu 17.04 Desktop Remix

Новость прошлого месяца о том, что Ubuntu отказывается от Unity и переходит на Gnome 3, наделала много шума, но что это означает в плане удобства в использовании? Что ж, уже есть Gnome-версия Ubuntu вместе с другими рабочими столами, и мы могли бы принести вам ее. Но потом мы подумали: «Какой смысл показывать вам один альтернативный рабочий стол, если можно дать пять?». И мы решили сделать еще один из наших ремиксов рабочих столов.

На сей раз это Ubuntu 17.04 с Gnome, KDE Plasma, XFCE, Cinnamon, Mate и стандартным рабочим столом Unity на выбор. При загрузке с DVD можно выбрать рабочий стол в меню Ubuntu.

Попробовать с DVD можно любой из рабочих столов, но вы заметите, что в большинстве из них нет значка установки. Причина в нашем теперешнем способе создавать ремиксы рабочих столов, чтобы не нарушать ставших более строгими правил лицензирования Ubuntu, которые не разрешают установку из ремикса. Мы взяли самую простую опцию и удалили возможность установки, чтобы вы не тратили напрасно времени на запуск программы установки, которая выдает отказ в самом конце. Вы всё равно сможете установить Ubuntu 17.04 с этого DVD, но получите только стандартный рабочий стол Unity.

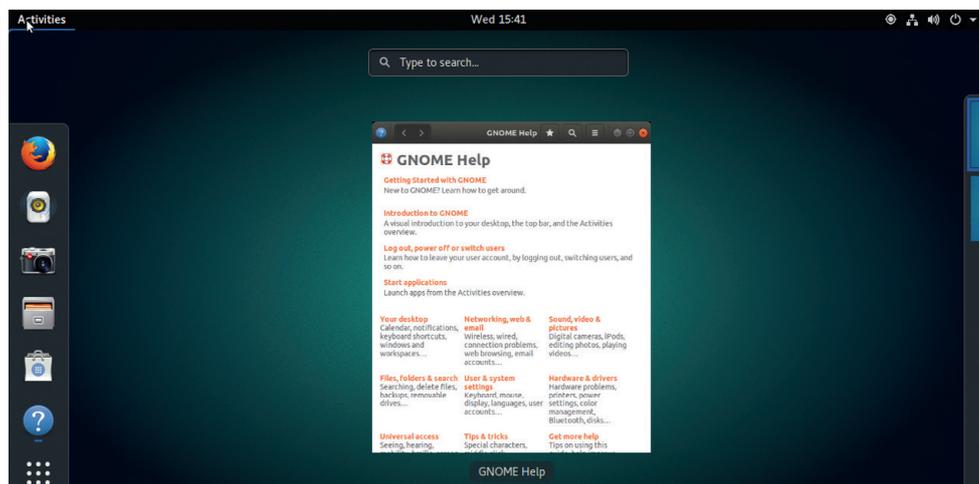
Добавить любой другой рабочий стол несложно, особенно если вы сначала установите менеджер пакетов Synaptic, который предлагает больше функций, чем



» Выберите рабочий стол, любой рабочий стол...

Ubuntu Software Centre. Откройте менеджер пакетов и добавьте репозитории multiverse и universe. Затем перезагрузите список пакетов, и вы увидите, что стали доступными другие рабочие столы. Пакеты, которые вам надо установить: *ubuntu-gnome-desktop*, *kubuntu-desktop*, *xubuntu-desktop*, *cinnamon-desktop-environment* или *mate-desktop-environment*. Каждый из них установит всё необходимое для использования данного конкретного рабочего стола. Затем можете выйти и выбрать свою новую среду. В отличие от live DVD, ваш выбор сохранится и при следующей загрузке будет выбран по умолчанию.

Аутентификационная информация: имя пользователя *ubuntu*, и можете оставить поле пароля пустым.



**ЭКСКЛЮЗИВ**  
**ОСОБЫЙ UBUNTU**  
 Пробуйте будущий Gnome и еще 5 жизненно важных рабочих столов Linux!

UNTY XFCE GNOME CINNAMON KDE MATE

**Solus 2017**  
 Совсем новый дистрибутив с крутым рабочим столом Budgie — ну нельзя не попробовать!

**Android 6.0**  
 Доступ к устройствам с сенсорными экранами прямо с настольного ПК

**LIVE-ДИСК: ГОТОВ К РАБОТЕ**  
 ВСЕ НЕОБХОДИМОЕ ДЛЯ СТАРТА В LINUX

# Новичок в Linux? Начните отсюда!

- » Что такое Linux? Как его установить?
- » Есть ли в нем эквивалент *MS Office*?
- » Зачем нужна командная строка?
- » Как устанавливать программы?

Ответы приводятся в [Index.html](#) на диске.

Дистрибутив, который обновляется сам

## Solus 2017.04.18.0

Многие используемые нами дистрибутивы являются респинами или ответвлениями других дистрибутивов. Это не обязательно плохо; Ubuntu предложил нечто отличное от Debian, а затем на Ubuntu был создан Mint; но иногда приятно видеть оригинальный дистрибутив. Solus — как раз такой. Он разработан специально для домашнего пользования, с понятным, легким и отзывчивым рабочим столом Budgie (в этой версии). Это дистрибутив с возобновляемым релизом, то есть когда вы его установите, менеджер пакетов сам будет всё обновлять — вам не придется ждать нового релиза дистрибутива. Аутентификационная информация: имя пользователя — **live**, пароль пустой.

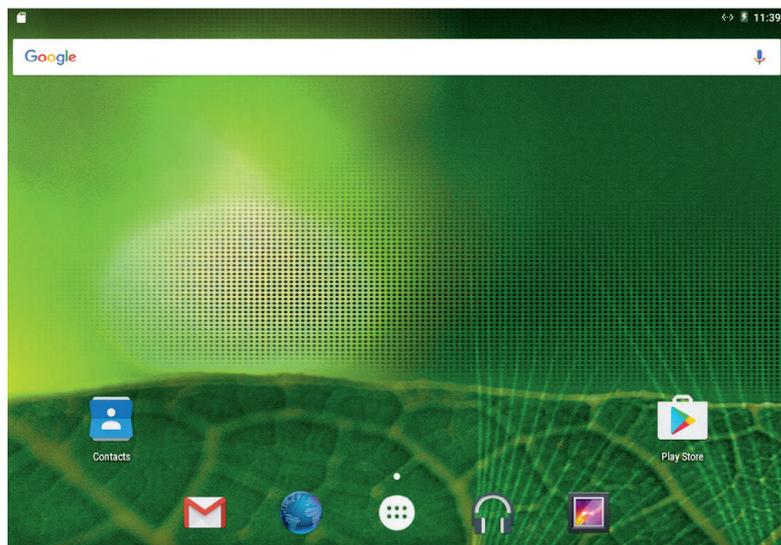


Многоцелевая версия Android

## Android-X86 6.0-r3

Linux уже используется большим числом потребителей, чем Windows. Конечно же, мы имеем в виду Android. Хотя Android предназначен для мобильных устройств, обычно работающих на ARM CPU, имеется порт частей Android с открытым кодом на семейство процессоров x86 (32- и 64-битных). Здесь у нас самый последний релиз, который может работать на чем угодно, от планшета x86 до настольного ПК. Помните, что

немногие мобильные телефоны поставляются с мышью, так что вы не сможете щелкать по интерфейсу, как обычно. Однако Android также дружелюбен к клавиатуре, и вы можете использовать клавиши со стрелками и Tab для перемещения между элементами и Enter для их выбора. Но во всей красе он проявляет себя при использовании с тачскрином, чего я никогда не использовал в Linux, однако с Android это отлично работает.



» У вас где-то завлялся старый ноутбук? Верните его к жизни с этой версией Android.

## И еще!

Системные инструменты

### Главное

**Checkinstall** Установка tar-архива с помощью менеджера пакетов.

**GNU Core Utils** Основные утилиты, которые обязаны присутствовать в каждой операционной системе.

**Hardinfo** Инструмент тестирования системы.

**Kernel** Исходный код самого последнего стабильного релиза ядра.

**Mentest86+** Проверьте ОЗУ на предмет неисправностей и сбоев.

**Plop** Простой менеджер загрузки для запуска ОС с CD, DVD и USB.

**RaWrite** Создавайте загрузочные дискиеты в MS-DOS в Windows.

**SBM** Независимый от ОС менеджер загрузки с несложным интерфейсом.

**WvDial** Соединяйтесь с Интернетом через телефонный модем.

Чтение

### Книжная полка

**Расширенное руководство по скриптам Bash** Изучите написание скриптов еще лучше.

**Руководство Bash для начинающих** Овладейте написанием скриптов *Bash*.

**Руководство по созданию скриптов Bourne Shell** Начните осваивать скрипты оболочки.

**The Cathedral and the Bazaar [Собор и Базар]** Классический текст Эрика С. Реймонда [Eric S Raymond], объясняющий преимущества открытой разработки.

**Справочник администратора Debian** Базовое руководство для системных администраторов.

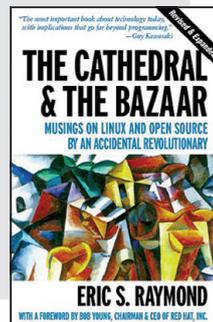
**Введение в Linux** Удобное руководство со множеством подсказок для начинающих пользователей Linux.

**Словарь Linux** Терминология Linux от А до Я.

**Linux Kernel in a Nutshell [Ядро Linux в Двух Словах]** Введение в ядро, написанное великим магистром ядра Греггом Кроа-Хартманом [Greg Kroah-Hartman].

**Руководство системного администратора Linux** Контролируйте свою систему.

**Обзор инструментов** Полный обзор инструментария GNU.



# Пропустили номер?

**LINUX**  
FORMAT  
Главное в мире Linux

Закажите его на сайте [www.linuxformat.ru](http://www.linuxformat.ru) в «ГНУ/Линуксцентре»!  
Журналы доставляются и в печатной, и в электронной форме,  
так что получение нужного вам выпуска LXF может занять всего  
пару минут с момента открытия браузера!

## LXF224 Июнь 2017



- » **Конец Unity** Ubuntu перешел на Gnome
- » **Где KDE краше** Лучший дистрибутив с KDE
- » **Боронися, бабка** Брандмауэр + роутер = защита
- » **Кому верить?** Надежные ключи шифрования

**LXF DVD:** Ubuntu, Linux Lite, Manjaro, Netrunner, 12 книг о Linux (на английском), горячие новинки и прочее...

**Печатная версия:** [shop.linuxformat.ru/lxf\\_224/](http://shop.linuxformat.ru/lxf_224/)  
**PDF-версия:** [shop.linuxformat.ru/elxf\\_224/](http://shop.linuxformat.ru/elxf_224/)

## LXF223 Май 2017



- » **Ставим на Zero** Raspberry Pi Zero W уже с Wi-Fi и Bluetooth
- » **Дизайн для 3D-принтера** Создание моделей
- » **В стиле Blu-ray** Заставим работать в Linux
- » **Стеганография** И чтоб никто не догадался

**LXF DVD:** feren OS, IPFire, openSUSE Tumbleweed, RancherOS, Rescatux, Ubuntu Studio, XenialDog, 12 книг о Linux (на английском), горячие новинки и прочее...

**Печатная версия:** [shop.linuxformat.ru/lxf\\_223/](http://shop.linuxformat.ru/lxf_223/)  
**PDF-версия:** [shop.linuxformat.ru/elxf\\_223/](http://shop.linuxformat.ru/elxf_223/)

## LXF222 Апрель 2017



- » **Стать Linux-джедаем** Изучаем терминал
- » **В Интернет — невидимкой** Дистрибутивы конфиденциальности
- » **Серверы на все лады** Два учебника и статья
- » **Сайт с галереей** Адаптивный web-дизайн

**LXF DVD:** Devuan, Linux Mint, ROSA, Scientific Linux, Ubuntu Budgie, 12 книг о Linux (на английском), горячие новинки и прочее...

**Печатная версия:** [shop.linuxformat.ru/lxf\\_222/](http://shop.linuxformat.ru/lxf_222/)  
**PDF-версия:** [shop.linuxformat.ru/elxf\\_222/](http://shop.linuxformat.ru/elxf_222/)

## LXF221 Март 2017



- » **Прощай, Google!** Мы найдем получше
- » **Ваш добрый демон** Выбор ОС BSD
- » **Сервер на халяву** CentOS, клон Red Hat
- » **Linux + Windows** Сожительство возможно

**LXF DVD:** ExTiX, Porteus, Rebecca Black Linux, Siduction, Ubuntu, Zorin, 12 книг о Linux (на английском), горячие новинки и прочее...

**Печатная версия:** [shop.linuxformat.ru/lxf\\_221/](http://shop.linuxformat.ru/lxf_221/)  
**PDF-версия:** [shop.linuxformat.ru/elxf\\_221/](http://shop.linuxformat.ru/elxf_221/)

## LXF219/220 Февраль 2017



- » **Linux навсегда** А Windows — на мыло
- » **Дистрибутивы в весе пера** Легковесный не значит убогий
- » **Графика будущего** Переходим на Wayland
- » **Рация-переноска** Ностальгируем с Raspberry Pi

**LXF DVD:** antiX, Bodhi, Fedora, Manjaro, openSUSE, 12 книг о Linux (на английском), горячие новинки и прочее...

**Печатная версия:** [shop.linuxformat.ru/lxf\\_219-220/](http://shop.linuxformat.ru/lxf_219-220/)  
**PDF-версия:** [shop.linuxformat.ru/elxf\\_219-220/](http://shop.linuxformat.ru/elxf_219-220/)

## LXF218 Январь 2017



- » **Приручение Яка** Он же — Ubuntu 16.10
- » **Новые, хромовые** Выбираем хромбук
- » **Помирить нам рановато** Шанс 32-битных машин
- » **Телефон в терминале** Управляем с Linux-ПК

**LXF DVD:** BunsenLabs, Calculate Linux, GeckoLinux, PelicanPC, Ubuntu, 12 книг о Linux (на английском), горячие новинки и прочее...

**Печатная версия:** [shop.linuxformat.ru/lxf\\_218/](http://shop.linuxformat.ru/lxf_218/)  
**PDF-версия:** [shop.linuxformat.ru/elxf\\_218/](http://shop.linuxformat.ru/elxf_218/)

Подпишитесь на печатную версию журнала на [www.linuxformat.ru/subscribe](http://www.linuxformat.ru/subscribe)  
или [www.linuxcenter.ru](http://www.linuxcenter.ru), и получите электронную версию в подарок!

Телефоны отдела подписки

- » Санкт-Петербург: (812) 309-0686
- » Москва: (499) 271-4954



Linux Format ВКонтakte:  
[vk.com/linuxform](https://vk.com/linuxform)

## » Содержание

Linux  
Format

### Страница 1

#### ДИСТРИБУТИВЫ

**Ubuntu 17.04 Desktop Remix** (64-битный) —  
Ubuntu, KDE, Xfce, Sinnamon, MATE, Unity  
Полноценная ОС Linux для настольных компьютеров,  
свободно доступная с поддержкой сообщества  
и профессионалов. Согласно Manifestу  
Ubuntu, ПО должно быть доступно, программы  
интерпретры должны предоставляться людям  
на их родном языке и не зависеть от ограничени-  
их физических возможностей, и пользователи  
должны иметь право настраивать и изменять свое  
ПО любым способом на свое усмотрение.

#### НОРРИКС

**СМСТ** (Sn) Менеджер подключения  
<https://github.com/sergej-tyshin/smsst>  
**Elisa** (Gh) Музыкальный плеер  
<https://github.com/KDE/elisa>  
**GNU Emacs 25.2** Операционная система  
[www.gnu.org/software/emacs/](http://www.gnu.org/software/emacs/)  
**Линья 1.8.1** Библиотека видео  
<https://github.com/Djag/djag>  
**Muzas Mail 2.0** Почтовый клиент  
<https://github.com/Lukas/Duvas-Mail>  
**Ориент Д 1.7** Транспортирующий симулятор  
[www.orientd.org](http://www.orientd.org)  
**Open Browser 0.9.91** Web-браузер  
[www.openb.org](http://www.openb.org)

#### Solus 2017.04.18.0

Оригинальный дистрибутив с возобновляемым  
ресурсом, разработанный специально для домашнего  
пользования, с понятным, легким и отзывчивым  
рабочим столом Vidgie.

#### Android-x86

Порт Android (исходно) — ОС для мобильных  
устройств с открытым кодом на семейство  
процессоров x86, способный работать на чем угодно,  
от планшета x86 до настольного ПК.

#### Marzette 2100 3.2.2

Игра-стратегия  
<https://vz2.100.net>

**Redshift 1.11** Программа защиты зрения  
<https://github.com/jk/Redshift>

**Simon 0.4.1** Программа расписания речи  
<https://github.com/sergej-tyshin/simon>

**Persepolis 2.4.2** Менеджер загрузки  
<https://github.com/sergej-tyshin/persepolis/>

**Persepolis 2.4.2** Менеджер загрузки  
<https://github.com/sergej-tyshin/persepolis/>

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. Поэтому, если в вашей системе возникли проблемы с зависимостями, первым делом следует заглянуть именно туда.

### Форматы пакетов

Мы стараемся включать как можно больше различных типов установочных пакетов: RPM, Deb или любых других. Просим вас принять во внимание, что мы ограничены свободным пространством и доступными двоичными выпусками программ. По возможности, мы будем включать исходные тексты для любого пакета, чтобы вы могли собрать его самостоятельно.

### Документация

На диске вы сможете найти всю необходимую информацию о том, как устанавливать и использовать некоторые программы. Пожалуйста, не забывайте, что большинство программ поставляются вместе со своей документацией, поэтому дополнительные материалы и файлы находятся в соответствующих директориях.

### Что это за файлы?

Если вы новичок в Linux, вас может смутить изобилие различных файлов и расширений. Так как мы стараемся собрать как можно больше вариантов пакетов для обеспечения совместимости, в одном каталоге часто находятся два или три файла для различных версий Linux и различных архитектур, исходные тексты и откомпилированные пакеты. Чтобы определить, какой именно файл вам нужен, необходимо обратить внимание на его имя или расширение:

- » **имя\_программы-1.0.1.i386.rpm** — вероятно, это двоичный пакет RPM, предназначенный для работы на системах x86;
- » **имя\_программы-1.0.1.i386.deb** — такой же пакет, но уже для Debian;
- » **имя\_программы-1.0.1.tar.gz** — обычно это исходный код;
- » **имя\_программы-1.0.1.tgz** — тот же файл, что и выше этажом по списку: «tgz» — это сокращение от «tar.gz»;
- » **имя\_программы-1.0.1.tar.bz2** — тот же файл, но сжатый bzip2 вместо обычного gzip;
- » **имя\_программы-1.0.1.src.rpm** — также исходный код, но поставляемый как RPM-пакет для упрощения процесса установки;
- » **имя\_программы-1.0.1.i386.FC4.RPM** — двоичный пакет RPM для x86, предназначенный специально для операционной системы Fedora Core 4;
- » **имя\_программы-1.0.1.ppc.Suse9.rpm** — двоичный пакет RPM, предназначенный специально для операционной системы SUSE 9.x PPC;
- » **имя\_программы-devel-1.0.1.i386.rpm** — версия для разработчиков.

### Если диск не читается...

Это маловероятно, но если все же прилагаемый к журналу диск поврежден, пожалуйста, свяжитесь с нашей службой поддержки по электронной почте: [disks@linuxformat.ru](mailto:disks@linuxformat.ru)

## ЭКСПЛЮЗИВ

# ОСОБЫЙ UBUNTU

64-битный



Пробуйте будущий Gnome и еще 5  
жизненно важных рабочих столов Linux!



UNITY



XFCE



GNOME



CINNAMON



KDE



MATE

## Solus 2017

Совсем новый дистрибутив с круглым рабочим столом Vidgie — ну нельзя же пропустить!

64-битный



## Android 6.0

Доступ к устройствам с сенсорным экраном прямо с настольного ПК

32-битный



Linux  
Format  
**LIVE-ДИСК: ГОТОВ К РАБОТЕ**  
ВСЕ НЕОБХОДИМОЕ ДЛЯ СТАРТА В LINUX

# Содержание



## Страница 2

### УЧЕБНИКИ

Swift

### ГЛАВНОЕ

Checkinstall  
Coreutils  
Hardinfo  
Kernel  
Memtest86+  
Plop  
SBM  
WVDial

### ПОМОЩЬ

Руководство новичка  
Руководства  
Ответы  
ЧаВо (FAQ)

### ДОКУМЕНТАЦИЯ: 12 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  
**Bourne Shell Scripting** Начальное руководство по программированию на Bash  
**The Cathedral and the Bazaar** Классический текст Эрика Реймонда [Eric S Raymond] «Собор и базар»  
**The Debian Administrator's Handbook** Руководство администратора, написанное разработчиками Debian  
**Dive Into Python** Учебник по программированию на Python  
**Intro to Linux** Начальное руководство по Linux  
**Linux Dictionary** Словарь Linux, объясняющий специфическую терминологию

### Linux Kernel in a Nutshell

Описание ядра Linux, созданное одним из его выдающихся разработчиков — Греггом Краа-Хартманом [Greg Kroah-Hartman]

**Rule** Знаменитый справочник администрирования Linux  
**System Administrators Guide** Руководство по базовому администрированию Linux  
**GNU Tools Summary** Руководство по работе в командной строке и обзор основных утилит GNU

**Архив номеров журнала Linux Format за 2005–2015 гг.**

Все дистрибутивы представлены ISO-образами, который можно записать на отдельный носитель, и загрузить в live-режиме прямо с LXF DVD. У всех присутствует возможность установки на жесткий диск.

**Пожалуйста, перестаньте использовать один и тот же диск с инструкцией, опубликованной в журнале на стр. 109!**

**КОММЕНТАРИЙ** Присылайте ваши пожелания и предложения по электронной почте: [info@linuxformat.ru](mailto:info@linuxformat.ru)

**ДЕФЕКТИВНЫЕ ДИСКИ** В маловероятном случае обнаружения дефектов на вашем диске, обращайтесь, пожалуйста, по адресу [disk@linuxformat.ru](mailto:disk@linuxformat.ru)

Настоящий диск тщательно тестировался и проверялся на всех старых платформах, однако, как и в случае с любым новым ПО, мы рекомендуем вам использовать аппаратный сканер. Мы также рекомендуем всегда иметь под рукой актуальную резервную копию данных вашего жесткого диска. К сожалению, редакция Linux Format не в состоянии принимать на себя ответственность за любые повреждения, разрушения или иные убытки, которые могут повлечь за собой использование этого DVD, представленных на нем программ или данных. Прежде чем устанавливать какое-либо ПО на компьютер, пожалуйста, скачайте программу для загрузки с нашего сайта [www.linuxformat.ru](http://www.linuxformat.ru)

Тираж издательства ООО «Марком», 188652, Россия, Ленинградская область, Всеволожский р-н, дер. Юрки, Школьная ул., 7-а. Лицензия ИДТР ВАО № 77-03.

## Создание установочных дисков при помощи cdcrecord

Самый быстрый способ записать ISO-образ на чистую матрицу — это обратиться к программе *cdcrecord*. Для всех перечисленных ниже действий потребуются права суперпользователя-*root*. Сначала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdcrecord -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:

```
cdcrecord dev=0,3,0 -v /путь к образу/image.iso
```

Чтобы упростить дальнейшее использование *cdcrecord*, сохраните некоторые настройки в файле */etc/default/cdcrecord*. Добавьте по одной строке для каждого устройства записи (скорее всего, в вашей системе присутствует только одно такое устройство):

```
Plextor= 0,3,0 12 16M
```

Первое слово в этой строке — метка; затем после адреса SCSI-устройства вы должны указать скорость и размер буфера. Теперь можете заменить SCSI-адрес в командной строке на выбранную вами метку. Все будет еще проще, если вы добавите следующее:

```
CDR_DEVICE=Plextor
```

Для записи ISO-образа вам осталось набрать команду

```
cdcrecord -v /path/to/image.iso
```

Если вы не принадлежите к любителям командной строки, в таком случае вам придет на помощь утилита *gcombust*. Запустите ее от имени *root* и выберите вкладку *burn* и ISO 9660 Image в верхней части окна. Введите путь к образу, который вы хотите записать на диск, и смело нажимайте на *Combust!* Пока ваш образ пишется на диск, можете выпить чашечку кофе.

### Другая ОС?

Использовать Linux для записи компакт-диска не обязательно. Программы вроде *cdcrecord* просто переносят двоичные данные на чистую матрицу. Все необходимые файлы уже включены в ISO-образ, который распознается любой операционной системой, будь то Linux, Windows, Mac OS X или AmigaOS.

### Нет устройства для записи дисков?

Если у вас нет устройства, с помощью которого можно было бы записать образ на диск, можно найти какого-нибудь друга или организацию, у кого есть компьютер с дисководом, и прожечь диск у них. Опять-таки, вам подойдет любая операционная система, способная распознать пишущий привод (см. выше).

Некоторые дистрибутивы умеют монтировать образы дисков и выполнять сетевую установку или даже установку с раздела жесткого диска. Конкретные методы, конечно, зависят от дистрибутива. За дополнительной информацией обращайтесь на web-сайт разработчика дистрибутива.

# LINUX FORMAT

# Подписывайтесь и читайте Linux Format на iPad или iPhone!

Доступно  
в AppStore



А если у вас Android, подпишитесь  
на Linux Format через Zinio!

 zinio™  
Доступно в Google Play





# 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 1200 экз., распространение электронной версии 30 000 экз.

## РЕДАКЦИЯ РУССКОЯЗЫЧНОЙ ВЕРСИИ

### Главный редактор

Кирилл Степанов [info@linuxformat.ru](mailto:info@linuxformat.ru)

### Литературный и выпускающий редактор

Елена Толстякова

### Переводчики

Елена Ессяк, Светлана Кривошеина, Валентин Развозжаев, Валерий Смирнов, Елена Толстякова

### Редактор диска

Александр Баракин

### Верстка, допечатная подготовка

Сергей Рогожников

### Технический директор

Андрей Смирнов

### Директор по рекламе

Владимир Савельев [advert@linuxformat.ru](mailto:advert@linuxformat.ru)

### Генеральный директор

Павел Фролов

### Учредители

Частные лица

### Издатель

ООО «Линус Формат»

### Отпечатано в типографии ООО «ЛД-ПРИНТ»

196644, Санкт-Петербург, Колпинский р-н, пос. Саперный, территория предприятия «Балтика», д. 6/н, лит. Ф  
Тел. (812) 462-8383, e-mail: [office@ldprint.ru](mailto:office@ldprint.ru)  
Заказ 16215

## РЕДАКЦИЯ АНГЛОЯЗЫЧНОЙ ВЕРСИИ

Редактор Нейл Мор [Neil Mohr] [neil.mohr@futurenet.com](mailto:neil.mohr@futurenet.com)

Научный редактор Джонни Бидвелл [Jonni Bidwell]

[jonni.bidwell@futurenet.com](mailto:jonni.bidwell@futurenet.com)

Выпускающий редактор Клиффорд Хоуп [Cliff 'Our Only' Hope]

[clifford.hope@futurenet.com](mailto:clifford.hope@futurenet.com)

Художественный редактор Эфраин Эрнандес-Мендоса

[Efrain Hernandez-Mendoza] [efrain.hernandez-mendoza@futurenet.com](mailto:efrain.hernandez-mendoza@futurenet.com)

## ПОДГОТОВКА МАТЕРИАЛОВ

Матс-Тэге Аксельссон [Mats Tage Axelsson], Джонни Бидвелл [Jonni Bidwell], Нейл Ботвик [Neil Bothwick], Стюарт Бернс [Stuart Burns], Джейсон Каннон [Jason Cannon], Шон Конвей [Sean D. Conway], Нейт Дрейк [Nate Drake], Кент Ельчук [Kent Elchuk], Энди Келли [Andy Kelly], Ник Пирс [Nick Peers], Лес Пандер [Les Ponder], Шашанк Шарма [Shashank Sharma], Александр Толстой [Alexander Tolstoy], Михалис Цукалос [Mihalis Tsoukalos], Евгений Балдин, Андрей Гондаренков, Алексей Федорчук, Максим Черепанов, Лада Шерышова, Игорь Штомпель

Иллюстрации Шейн Коллиндж [Shane Collinge]

Иллюстрация с обложки [www.magictorch.com](http://www.magictorch.com)

## КОНТАКТНАЯ ИНФОРМАЦИЯ

UK: Linux Format, 30 Monmouth Street, Bath BA1 2BW

Тел. +44 (0) 1604 251045, email: [linuxformat@myfavouritemagazines.co.uk](mailto:linuxformat@myfavouritemagazines.co.uk)

РОССИЯ: Санкт-Петербург, пр. Медиков, 5, корп. 7

Тел. +7 (812) 309-0686, e-mail: [info@linuxformat.ru](mailto:info@linuxformat.ru)

По вопросам сотрудничества, партнерства, оптовых закупок:

[partner@linuxcenter.ru](mailto:partner@linuxcenter.ru)

**Авторские права:** статьи, переведенные из английского издания Linux Format, являются собственностью или лицензированы Future Publishing Ltd (Future plc group company). Все права зарегистрированы. Никакая часть данного журнала не может быть повторно опубликована без письменного разрешения издателя.

Все письма, независимо от способа отправки, считаются предназначенными для публикации, если иное не указано явно. Редакция оставляет за собой право корректировать присланные письма и другие материалы. Редакция Linux Format получает неэксклюзивное право на публикацию и лицензирование всех присланных материалов, если не было оговорено иное. Linux Format стремится оставлять уведомление об авторских правах всюду, где это возможно. Свяжитесь с нами, если мы не упомянули вас как автора предложенных вами материалов, и мы постараемся исправить эту ошибку. Редакция Linux Format не несет ответственности за опечатки.

Ответственность за содержание статьи несет ее автор. Мнение авторов может не совпадать с мнением редакции.

Все присланные материалы могут быть помещены на диски — CD или DVD, поставляемые вместе с журналом, если не было оговорено иное.

**Ограничение ответственности:** используйте все советы на свой страх и риск. Ни при каких условиях редакция Linux Format не несет ответственности за повреждение или ущерб, нанесенные вашему компьютеру и периферии вследствие использования тех или иных советов.

Linux — зарегистрированный товарный знак Линуса Торвальдса [Linus Torvalds].

"GNU/Linux" заменяется на "Linux" в целях сокращения. Все остальные товарные знаки являются собственностью их законных владельцев. Весь код, опубликованный в журнале, лицензирован на условиях GPL v3. См. [www.gnu.org/copyleft/gpl.html](http://www.gnu.org/copyleft/gpl.html)

За информацией о журнале, издаваемом Future plc group company, обращайтесь на сайт [www.futureplc.com](http://www.futureplc.com).

В августовском номере

Издание 2017 г.:

# Долой Windows!

Для новичков: руководство по установке Linux, быстрому старту и превращению в эксперта. Все — бегом вслед за Туксом!

## Сборка дистрибутива Arch

Новичок? Не беда! Следуйте по арочному заводу и сами изготовьте идеальный для себя дистрибутив.

## Выбираем провайдера VPN

Правительства Великобритании и США следят за каждым вашим шагом. Надо отступить в тень.

## Головоломки LXF!

Раз просили, получайте: вбрасываем первую часть наших загадок для программеров! (Если честно, не особо трудных.)

Содержание будущих выпусков может меняться — вдруг из-за множества установок нам будет некогда писать...



© Linux Format 2005

© Future Publishing Ltd 2005

BATH • LONDON • MILAN • NEW YORK • PARIS • SAN DIEGO • SAN FRANCISCO

16+

# 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](http://www.robboclub.ru)

8 800 777 2985

[vk.com/robboclub](https://vk.com/robboclub) ● [info@robboclub.ru](mailto:info@robboclub.ru)

**HETZNER**  
ONLINE

**БЫСТРЫЙ КАК МОЛНИЯ**

**ОСТРЫЙ КАК ЛЕЗВИЕ**

**НОВИНКА**



## Выделенный сервер EX51-SSD-GPU

Intel® Core™ i7-6700 Quad-Core Skylake Processor  
64 ГБ DDR4 RAM  
2 x 500 ГБ SATA 6 Gb/s SSD  
Видеокарта GeForce® GTX 10800  
100 ГБ место для резервных копий  
50 ТБ трафик\*  
Без минимального контракта  
Установка 7200 рублей



**7200** рублей в месяц

## Идеальное решение для ресурсоёмких вычислений.

Новый выделенный сервер EX51-SSD-GPU содержит невероятно мощную графическую карту GeForce® GTX 1080 для ускорения графических приложений и быстрого 3D-рендеринга.

[www.ru.hetzner.com](http://www.ru.hetzner.com)

\* Нет платы за превышение. При превышении 50 ТБ/месяц скорость соединения ограничивается (подсчёт ведётся по исходящему трафику, входящий и внутренний трафик не учитывается). Опционально можно снять ограничение, подтвердив оплату 85 руб. за каждый дополнительный ТБ.

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.