

DVD ДАРОМ!



LINUX ДЛЯ НОВИЧКА!

# LINUX FORMAT

Главное в мире Linux

Август 2017 № 8 (226)

**72** страницы  
учебников  
и статей!

- » Идеал настольного Ubuntu
- » Свой дистрибутив Arch
- » Взлом ПК через USB-брелок

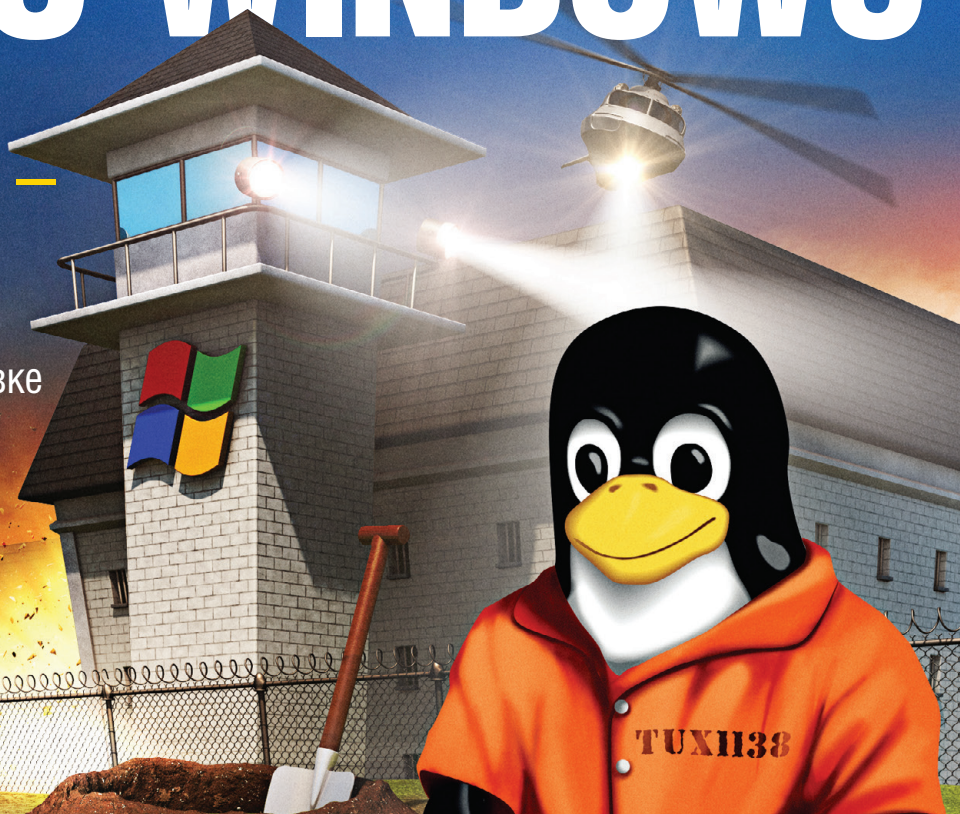


Академия кодига: Разгадаете ли вы загадку LXF?

# ПОБЕГ С WINDOWS

Долой раздутое,  
шпионское и вредное ПО —  
переходите на Linux!

- » Полный комплект для беглеца
- » Простое руководство по установке
- » Рабочий стол: Начало
- » Изучаем открытые программы
- » Возврата к Microsoft не будет!



ЕСТЬ  
ЭЛЕКТРОННАЯ  
ВЕРСИЯ  
ДЛЯ ВАШЕГО  
ПЛАНШЕТА!



## Приятные пакеты

“Люди продолжают переделывать  
одни и те же вещи, каждый раз  
совершая одни и те же ошибки”

Джо Дамато — об управлении пакетами

## Плюс: Для Pi

7 страниц в копилку Raspberry Pi

- » Покадровая съемка
- » Хорош ли Ubuntu Mate?
- » Сетевая игра DOOM



Творчество

### Кинопроизводство

» Делаем свои первые шаги  
в анимации со свободным ПО



Сравнение

### Лучший сервис VPN

» Тестируем лучшие  
VPN-сервисы для Linux



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

Linux center  
www.linuxcenter.ru



# iTeleRadio

ИНТЕРАКТИВНОЕ РАДИО&ТЕЛЕВИДЕНИЕ



## 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 месяце мы помогаем людям послать Windows далеко и надолго, а что вы изгнали бы из мира Linux?



### Джонни Бидвелл

Я бы хотел изгнать морочку с выключением и включением звука своей звуковой карты, чтобы та перестала издавать ужасный скрежет при попытке воспроизвести музыку. Уж лучше *PulseAudio* и его выходы при пробуждении после приостановки, а также психотранс (психоделический транс), который мои домашние обзывают ужасным скрежетом.



### Нейл Ботвик

Мнение, что Linux — это только для пользователей Windows, которые не хотят платить за свое программное обеспечение. Акцент на свободном (в смысле, как раздача пива) аспекте умаляет качество и охват доступных программ. Пожалуйста, бегите из Windows в Linux исключительно потому, что Linux лучше, а не дешевле.



### Нейт Дрейк

Меня озадачивает безумное рвение защитников различных версий Linux, особенно когда они критикуют Ubuntu за то, что он имеет наглость являться гибким и простым в использовании. Мы хотим дать людям Linux или показать, что мы ужасно умные, поскольку умеем компилировать свое собственное ядро?



### Ник Пирс

Я бы хотел избавиться от большинства сервисов, основанных на подписке. Меня достала навязчивая идея заставлять нас время от времени платить за привилегию использования определенной программы, просмотра видео или слушания музыки. Если я что-то покупаю, я это храню. И хочу иметь возможность его форматировать.



### Лес Паундер

Знаете, я мог бы вставить здесь шутку о том, что надо бы изгнать *Vim*, `Esc: wq!`, но не буду... ой, подождите, вот я ее и вставил. Не хочу ничего худого сказать о *Vim/vi*, но я предпочитаю *pico*. Поэтому мой совет для людей, которые хотят улучшить свои игры в редактирование текстов, таков: узнайте, как выйти из *Vim/vi*, а затем поучитесь использовать его.



### Валентин Синецын

Пожалуй, поддержку Нейта: нам нужно меньше нетерпимости во всех возможных проявлениях. Даже если кто-то не использует Linux совсем, это не значит, что он чем-то хуже «Пользователя Linux #12345». Агрессия к «инакомыслящим» и чувство собственного превосходства — не наш путь.



## Конвергенция двух систем

» Полвека назад все советские газеты дружно клеймили как «антинаучную» идею конвергенции социализма и капитализма, сторонники которой утверждали, что в будущем во всех странах установится одинаковый общественный строй, взявший многое (и не всегда лучшее) из обоих названных. Не прошло и 25 лет, как история расставила всё на свои места.

15 лет назад Билл Гейтс, обращаясь к разработчикам, утверждал, что свободный код убивает рабочие места, а маркетинговые специалисты Microsoft изводили километры бумаги, доказывая, что Linux обречен, так как для его эксплуатации требуются слишком «дорогие» админы. Сегодня — уже другие люди, занимающие те же должности в той же компании, с гордостью рапортуют о выходе Windows Subsystem for Linux, позволяющей выполнять код, написанный для Linux, непосредственно на Windows без виртуализации. Столь же пафосно напоминают публике, что две ключевых технологии Microsoft — .Net и PowerShell — портированы на Linux под свободной лицензией. Ну и про Docker для Windows, «наше всё» в эпоху микросервисов и прочего DevOps, тоже не забывают...

Если так пойдет и дальше, то старинная первоапрельская шутка про выход Microsoft Linux станет былью. Объявлялся же четверть века назад «флагманским продуктом в линейке ОС» от Microsoft свой UNIX по имени Xenix.

### 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

«Все умрут, все, вот где цирк! За одно это мы должны любить друг друга.» Чарльз Буковски

## Обзоры

### Slackel 7.0 Openbox ..... 14

Старый версии были не слишком впечатляющими, и мы старательно вглядывались в произошедшие изменения: что ж там хорошего?

### OBRevenge 2017.05 ..... 15

Странное у этого дистрибутива имечко... но кое-какие предположения по его расшировке сделать удастся.

### Skype для Linux Beta ..... 16

Microsoft сулил, что Skype для Linux будет ничуть не беднее функциями, чем Skype на других платформах. Вот и проверим.

### AMD Ryzen 5 1500X ..... 17

Этот четырехъядерный процессор разрушил конкуренцию с Intel, а заодно и собственную линейку! Поясняем, почему вам лучше доплатить.

### Sapphire RX 580 Nitro+ ..... 18

Новый графический процессор от AMD взбодрит ваши игры, но меньше, чем мы надеялись. Это история о техпроцессе и перенумерации.



» Если вы затеваете обновление своей машины, AMD будет кстати.

### Kentucky Route Zero ..... 19

Обычно слова «магический реализм» нас бесят, но тут мы не зря ждали три года — игра бросает вызов подлинной реальности.

### Lords of Xulima ..... 20

Еще один удачный проект с Kickstarter выполнил свои обещания, создав классическую ролевку в стиле «старой школы».



» Со временем даже озеро лавы не будет для нас препятствием.

## Побег с Windows

Выберите свободу, покиньте мир Windows и установите Linux прямо сейчас. Мы поясним, как установить и использовать Linux, а затем и проникнуть в его глубины.

» Открываем Linux на с. 30

» Как Linux загружается с. 36

» Установим Linux на с. 46

## Сравнение: Сервисы VPN с. 24

### From Anywhere With IVPN

Online freedom and privacy from the leaders in VPN security

- Safeguard your private data from snoopers and government surveillance
- Protect your identity from hackers and ID thieves
- Use our VPN service from anywhere on almost any device

LEARN MORE

Buffered VPN - Mullvad Fixation

Download Buffered VPN

For Desktop Systems

- Download for Windows
- Download for Mac OS X

For Routers

- Get Configuration Files

Welcome to Mullvad

Your internet traffic is now encrypted and anonymized using the Mullvad servers in order to deny third parties access to your communication.

The account expires in 3 hours. After that, time must be added to the account for continued use of the service.

Add time to the account

Close

Settings for Mullvad

Status: Settings

Account

Account number: 35362585800

Change account number

Account management

Network

Ports: none [port] Manage

Block the internet on connection failure

Stop DNS leaks

Tunnel IPv6

Country: Sweden

Advanced

Error report Help

Perfect Privacy VPN

Connected

Location: Amsterdam

Available Bandwidth

- 2560/3000 mbps
- 7/10 mbps
- 972/1000 mbps
- 99/100 mbps
- 1968/2000 mbps



## Интервью

« Я за то, чтобы начинать с простого, а уже потом задумываться об оптимизациях »

Мы обсудили пакеты с Джо Дамато с. 42

# На вашем бесплатном DVD

Побалуйте себя и любимых подпиской на LXF!

ЛУЧШИЙ ДИСТРИБУТИВ  
ДЛЯ НОВИЧКА

Elementary OS 0.4.1

Простой и понятный дистрибутив, который облегчит вам переход с MacOS или Windows

Быстрый Открытый Многофункциональный

Manjaro 17.0

Zorin OS 12.1

Выглядит профессионально, работает дружелюбно

Идеально настроенный, быстрый и очень красивый

LINUX LIVE-ДИСК: ГОТОВ К РАБОТЕ

ВСЕ НЕОБХОДИМОЕ ДЛЯ СТАРТА В LINUX

Elementary OS 0.4.1 64-битный

Zorin OS 12.1 64-битный

Manjaro 17.0.1 32-битный

» Каждый месяц — только лучшие дистрибутивы

ПЛЮС: HotPicks, код и библиотека с. 106

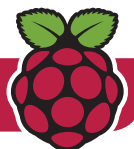


Доступно в AppStore!



www.linuxformat.ru/subscribe

## Пользователям Raspberry Pi



### Новости Pi ..... 92

Рассматриваем метеостанцию на Pi и новый вариант корпуса — ну и Pi дождался своего личного вредоносного ПО (повезло...).

### Ubuntu Mate 16.04 LTS ..... 93

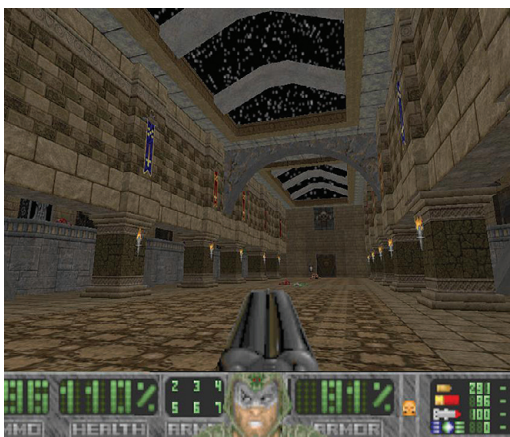
Мы проверили, загружается ли настольный Ubuntu на Raspberry Pi — да, это так! Потрясающе.

### Замедленная съемка ..... 94

Освойте цейтраферную (она же — покадровая) съемку, изучив наш удачный проект, использующий камеру Pi.

### DOOM в шоколаде ..... 96

Chocolate DOOM, свободный движок популярнейшей стрелялки, старается воспроизвести все ощущения от игры в мельчайших подробностях.



## Ищите в номере

### Ставим Linux сию минуту! ..... 46

В дополнение к основной статье данного номера, прилагаем руководство по быстрой и простой установке Linux в лице Ubuntu.

### История стартапа ..... 50

Денег не было, но они держались — и их проект дошел аж до Парижа. А добытый опыт останется с ними навсегда.

## Учебники

### Основы терминала

### Музыка по команде ..... 56

Управляемся со всеми вашими медиа из сумрачных глубин командной строки.

### Анимация

### В движеньи, в движении ..... 58

Изучим основы мультипликации, начиная с классики: отскоков мяча и цикла ходьбы.

### Сетевые атаки

### Не только взлом ..... 62

Разбираемся в перехвате данных через посредника и ARP-подтасовку. Всегда помните про букву s...

### Хакинг

### Привет от Уточки ..... 64

Приглашаем поиграть с резиновой уточкой и разъемом USB. (Не поддавайтесь на провокацию!!!)

### Модули ядра Linux

### Ядерная реакция ..... 66

Разберемся в устройстве ядра: почему в Linux не требуется установка драйверов?

### Настройка системы

### Рабочие столы Linux ..... 68

Вот так истинные линуксоиды работают с настольным Linux, не отягощая себя образом мыслей прошлого века.

### Дистрибутивы

### Свой Argh ..... 72

Изготовим себе собственный дистрибутив на базе Argh, для восстановления данных.

### Интернет вещей

### Царствуй, лежа на боку ..... 76

Применяем для Послушного Дома удаленный пульт на инфракрасных лучах.

### Тайны командной строки

### В дебрях Sed ..... 80

Оказывается, этот загадочный редактор командной строки не так страшен, как его малюют.



## Академия кодига

### Загадка LXF! ..... 84

В нашей первой (авось, и не последней) загадке генерируются сложные пароли на разных языках программирования. А не слабб вам сделать круче?

### Нарисуем число π ..... 88

Опять же на разных языках программирования, от Python и до R, попытаемся отобразить число π визуально — в виде скопления цветных псевдослучайных пикселей.

```
1 #!/usr/bin/env python3
2
3 import sys
4 import random
5
6 if len(sys.argv) != 2:
7     print("I need just one argument!")
8     sys.exit(0)
9
10 if sys.argv[1].isdigit():
11     count = int(sys.argv[1])
```

## Постоянные рубрики

### Новости ..... 6

И опять сошла ROSA, телефония попала на Эльбрус, а отечественные БИС — на спутник, AMD освоила 7-нм техпроцесс, мейнфрейм IBM свершил прорыв в шифровании, Intel создала принципиально новую архитектуру, а дата-центры выстраиваются по вертикали (из экономии) и получают супер-емкий SSD.

### Вести мобильных ОС ..... 22

Смартфон приходит из курятника, Nokia ставит на свои устройства оптику Zeiss, телефоны ждет голографический контент, а модемы Samsung рвутся к 5G.

### Сравнение ..... 24

Не верьте никому, а нам — в особенности. Выбирайте VPN-провайдера себе по сердцу: Buffered VPN, IVPN, Mullvad, Perfect Privacy, SlickVPN.

### Интервью LXF ..... 42

Джо Дамато рассказывает о невидимых ужасах, которые творятся при выполнении с виду приличного кода, и о провалах известных менеджеров пакетов.

### HotPicks ..... 100

Отведаете горяченького! Лучшие в мире новинки свободного ПО: ANGRYsearch, Curlw, Extreme Tux Racer, Feedreader, Flowblade,

KMail, Konv, Min, Screencloud, SpeedLazer, Vitunes.

### Диск Linux Format ..... 106

Содержимое двустороннего DVD этого месяца.

### Пропустили номер? ... 108

Услыхали подозрительный скрежет?.. Скорее хватайте LXF225 — там сказано, как обороняться от злых хакеров.

### Через месяц ..... 112

Всё на свете виртуально! Запихиваем Тукса в контейнеры Docker, внутрь его личной VirtualBox, проводим через графический процессор, и не только!



**В ЭТОМ НОМЕРЕ:** Новая ROSA » Звонки с Эльбруса » БИС на спутнике » 7 нм — не шутки » Прорыв от IBM » Новации Хеон » По вертикали: ЦОД » Викинги атакуют

## ДИСТРИБУТИВЫ

# РОСА для бизнеса

Обновлен защищенный отечественный дистрибутив, поддерживающий серверные аппаратные платформы и системы хранения.



» Рубрику готовил  
**АНДРЕЙ  
ГОНДАРЕНКОВ**

**Р**оссийская компания «НТЦИТ РОСА» выпустила новую версию дистрибутива ROSA Enterprise Linux Server (RELS) 6.9, на базе CentOS 6.9. Добавлено немало компонентов (в т. ч. оригинальных).

Одновременно, согласно политике компании, прекращена продажа лицензий ОС RELS 6.7, а релизы до 6.9 переведены в открытый формат. Для всех пользователей, приобретавших лицензию на ОС RELS 6.7, доступно бесплатное обновление до ОС RELS 6.9 включительно. Новые ключи для доступа к репозиториям такие пользователи могут бесплатно запросить в отделе продаж или в отделе техподдержки компании. На сайте компании также есть инструкция

по обновлению ОС RELS 6.x до текущей актуальной версии ОС RELS 6.9.

Ключевые отличия ОС RELS 6.9:

- » введен инструмент резервного копирования как ISO-образов — Relax-and-Recover;
- » реализация IPsec переведена с Openswan на Libreswan, что открыло доступ к расширенным средствам защиты VPN;
- » в SSSD улучшена совместимость с Active Directory и серверами идентификации;

**Добавлено немало компонентов (в том числе оригинальных).**

- » расширены средства для работы в роли гостевой системы под управлением гипервизора Hyper-V: реализована поддержка TRIM, поддержка 2-го поколения виртуальных машин, поддержка протоколов Windows 10 и Windows Server 2016; в virtio-win добавлены драйверы, позволяющие запускать Windows 10 в гипервизоре KVM;
- » в OpenSSL, NSS, GnuTLS, rsyslog и vsftpd добавлена поддержка протокола TLS 1.2;
- » обновлены драйверы оборудования, введена поддержка новых процессоров Intel;
- » учтены уязвимости, пакетная база доработана с учетом требований безопасности;
- » сохранена полная двоичная совместимость с версиями ОС CentOS/RedHat 6.x.

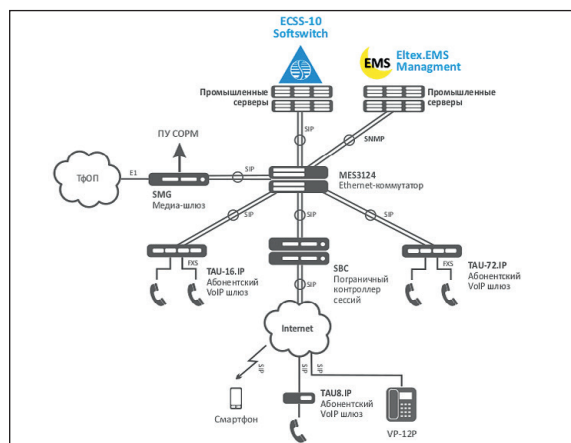
## IP-ТЕЛЕФОНИЯ

# Идем на Эльбрус

ПАК ECSS-10 Softswitch портирован на отечественную микропроцессорную платформу.

**Р**азработчик и производитель коммуникационного оборудования ООО «Предприятие ЭЛТЕКС» (г. Новосибирск) при поддержке АО «МЦСТ» (разработчик высокопроизводительных российских микропроцессоров и вычислительных комплексов на их основе) завершил портирование своего универсального программно-аппаратного комплекса Softswitch ECSS-10 для построения систем связи операторского и корпоративного уровня на базе технологии коммутации пакетов на отечественную 64-разрядную микропроцессорную архитектуру «Эльбрус». Тестирование решения на серверах

Эльбрус-4.4 с процессорами Эльбрус-4С (1891ВМ8Я) подтвердило расчетные характеристики комплекса для данной комплектации: производительность системы — от 25 вызовов в секунду; абонентская емкость — от 25 тыс.; поддержка видеозвонов; транскодирование медиа-поток; возможности масштабирования, резервирования в режиме active-active и построения систем с поддержкой географического резервирования. На ближайшее время запланированы работы по тестированию решений на новом восьмиядерном микропроцессоре Эльбрус-8С (1891ВМ10Я). Т.е. в активе ЭЛТЕКС появилось современное



» Комплексное решение для построения интегрированной информационно-коммуникационной сети ECSS-10 SOFTSWITCH переведено на отечественные процессоры Эльбрус.

масштабируемое решение IP-телефонии на серверной архитектуре отечественного производства. Серверы разработаны ПАО «ИНЭУМ им. И. С. Брука» и производятся на территории Российской Федерации.



ИМПОРТОЗАМЕЩЕНИЕ

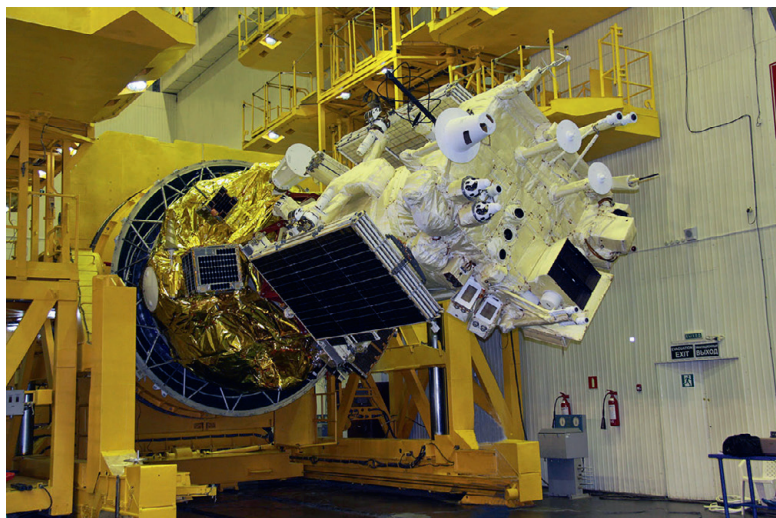
# Исполнено на БИС

Разработанные компанией «Ангстрем» большие интегральные схемы заменят импортные аналоги в метеоспутнике «Метеор-М».

**В** рамках опытно-конструкторских мероприятий по разработке и передаче опытных образцов «Корпорации ВНИИЭМ» — главному разработчику гидрометеорологического космического

аппарата «Метеор-М» — специалисты АО «Ангстрем» (разработчик и производитель одного из самых мощных в России комплексов по созданию и производству полупроводниковых изделий) разработали

комплект микросхем на основе базового матричного кристалла (БМК). В метеоспутнике «Метеор-М» отечественные интегральные схемы (ИС) блоков, отвечающих за изменение температурного режима, заменяют аналоги иностранного производства. Комментарий от Павла Машевича, директора центра микроэлектроники АО «Ангстрем»: «Микросхемы от „Ангстрема“ обладают более высокой стойкостью к факторам космического пространства и к воздействию тяжелых заряженных частиц по сравнению с иностранными аналогами, что скажется на сроках эксплуатации спутников на орбите». Гидрометеорологический космический аппарат «Метеор-М» осуществляет мониторинг погоды, собирает данные для определения температуры, передает данные глобальных и локальных изображений облачности, поверхности Земли, ледового и снежного покрова в видимом диапазоне. Первый запуск космического аппарата «Метеор-М» с ангстремскими БИС запланирован на декабрь.



➤ Космический аппарат «Метеор-М» предназначен для глобального наблюдения атмосферы и подстилающей поверхности Земли и получает гидрометеорологическую и гелиогеофизическую информацию в планетарном масштабе.

НОВОСТИ AMD

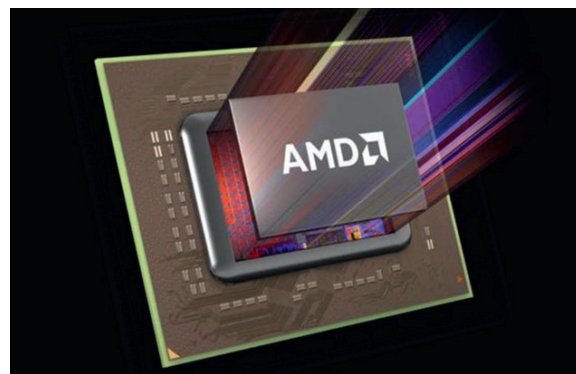
# Самый сложный переход

Техпроцесс 7 нм требует новых инструментов САПР и изменений архитектуры.

**Г**лава AMD Марк Пейпермастер [Mark Papermaster] назвал процесс перехода на техпроцесс 7 нм «самым сложным из всех предшествующих». В интервью EE Times он подтвердил, что второе и третье поколение процессоров Zen будут производиться по техпроцессу 7 нм, которому предсказал «очень долгую жизнь», подобно процессу 28 нм.

Помимо необходимости новых инструментов САПР и архитектурных изменений, при проектировании по нормам 7 нм потребовалось изменить способы подключения транзисторов и наладить более глубокие отношения с производственными партнерами. К 2019 г. на выпускающих предприятиях будет внедрена EUV-литография; при этом отпадает необходимость в использовании четырех фотошаблонов при изготовлении критически важных слоев (достаточно будет одного-двух),

что существенно уменьшает использование дорогостоящих масок и снижает затраты. Другим средством снижения себестоимости производства при переходе на техпроцесс 7 нм станет метод упаковки нескольких чипов в один корпус; AMD рассматривается к «2.5-D» — вертикальной конструкции со сквозными металлическими соединениями типа TSVs, где процессоры и память объединяются в стеки на одной подложке, однако пока это очень дорогой метод. Apple использует Integrated Fan Out Wafer-level Package — тип упаковки, объединяющий мобильные процессоры с памятью в залитую герметиком монолитную



➤ СТО AMD Марк Пейпермастер рассчитывает на долгую жизнь 7-нм техпроцесса.

**Потребовалось изменить способы подключения транзисторов.**

конструкцию; эта технология, получившая название «2.1-D», пока еще не пригодна для более мощных десктопных и серверных процессоров, однако через 2-3 года всё может измениться.

# Глобально шифруем данные

Сверхбыстрый IBM Z совершил наибольший прорыв в технологии мейнфреймов за более чем 10 лет.

**Н**овый мейнфрейм IBM Z способен выполнять свыше 12 млрд операций в день; в систему внедрен также «прорывной» механизм шифрования, позволяющий шифровать все данные транзакций, связанные с любым приложением, облачным сервисом или базой данных.

Основанная на фирменной CMOS-технологии, система IBM Z работает на «самом быстром на рынке микропроцессоре» с тактовой частотой 5,2 ГГц, поддерживает 32 ТБ памяти и имеет новую масштабируемую системную структуру, обеспечивающую 35-процентное увеличение производительности для традиционных рабочих нагрузок по сравнению с предыдущим поколением IBM z13. После внедрения для платформы решений на базе Linux и открытого ПО защита IBM Z значительно

расширилась за счет самых передовых технологий шифрования и защиты ключей; криптографические возможности системы распространяются на любые данные, сети, внешние устройства или целые приложения, такие как IBM Cloud Blockchain, без каких-либо изменений приложения и соглашения об уровне сервиса.

Новые возможности IBM Z в области шифрования предназначены для нейтрализации глобальной эпидемии утечек данных — одного из главных факторов воздействия киберпреступности на мировую

**> IBM Z работает в 18 раз быстрее фрагментированных систем x86, а стоимость обработки составляет 5% от аналогичных расходов для решений на базе x86.**



экономику, ущерб от которого к 2022 г. может составить \$8 трлн. Из более чем 9 млрд записей данных, утерянных или похищенных с 2013 г., зашифрованы были лишь 4%, что делает подавляющее большинство таких данных уязвимыми как для организованных киберпреступников, так и для государственных субъектов и сотрудников, злоупотребляющих доступом к конфиденциальной информации.

**Предназначено для нейтрализации эпидемии утечек данных.**

## ГОНКИ ЧИПМЕЙКЕРОВ

# Purley против Eрус

Intel выпустила линейку серверных процессоров Xeon на основе принципиально новой архитектуры.

**I**ntel представила новую серию серверных процессоров Purley Xeon на основе новой архитектуры Skylake-SP (Scalable Platform), имеющие до 28 процессорных ядер на сокет и поддержку до 1,5 ТБ ОЗУ на сокет (вместо стандартных 768 ГБ); производитель заявляет о повышении производительности в 1,65 раза по сравнению с предыдущим поколением серверных CPU на архитектуре Broadwell. Несколькими неделями ранее AMD представила свою линейку серверных процессоров Eрус на основе архитектуры Zen, с предложением до 32 ядер на чип; Intel утверждает, что производительность ее топового процессора Xeon Scalable на 28 % выше, чем у AMD Eрус 7601. В Сети появились первые результаты по сравнению производительности в тесте Cinebench R15: AMD EPYC (32 ядра, 64 потока) набрал 4000 баллов, Intel Xeon Platinum 8180 (28 ядер, 56 потоков) — 4200 баллов.

Mesh, принципиально новая топология архитектуры процессоров Xeon Scalable, обеспечит «фундаментальное увеличение» производительности: в отличие

от предыдущей топологии Ring, Mesh объединяет отдельные ядра в 3D-структуру, что увеличивает число путей для передачи сигнала и уменьшает преодолеваемые расстояния. Новые процессоры Intel ориентированы на тяжелые расчеты: облачные вычисления, автономные транспортные средства, 5G, искусственный интеллект (для которого от применения Xeon Scalable прогнозируется более чем двукратное увеличение производительности). По данным Intel, новые процессоры Xeon SP в 3,1 раза повышают эффективность криптографических вычислений; опция Intel Key Protection Technology (KTP) обеспечивает защищенное хранение и передачу ключей шифрования внутри платформы: доступ к ключам и управление ими осуществляется только владельцем.

**Новые процессоры Intel ориентированы на тяжелые расчеты.**

HIGHEST	ADVANCED	EFFICIENT	ENTRY
UP TO 28 CORES	UP TO 22 CORES	UP TO 16 CORES	UP TO 10 CORES
UP TO 4 & 8 SOCKET SUPPORT WITH UP TO 3 UPI LINKS	2 & 4 SOCKET SUPPORT	2 & 4 SOCKET SUPPORT	2 & 4 SOCKET SUPPORT
UP TO 1.5 TB TOP-LINE MEMORY CHANNEL BANDWIDTH	UP TO 3 UPLINKS	UP TO 3 UPLINKS	UP TO 3 UPLINKS
HIGHEST ACCELERATOR THROUGHPUT	ADVANCED RELIABILITY, AVAILABILITY AND SERVICEABILITY	EFFICIENT MODERATE TASKS	ENTRY ENTRY PERFORMANCE, PRICE OPTIMIZED FOR LIGHT WORKLOADS
MAINSTREAM	EFFICIENT	EFFICIENT	ENTRY

**> Platinum, Gold, Silver и Bronze различаются между собой не только количеством ядер и набором технологий, но и поддерживаемым объемом ОЗУ, устойчивостью к температурным нагрузкам, сроками работоспособности.**

Модели Bronze, Silver, Gold и Platinum, составившие новую линейку Xeon Processor Scalable Family, обеспечивают различные уровни производительности и варианты интеграции. Такие клиенты Intel, как Google Cloud, AWS и AT&T, уже получили около 500 тыс. штук новых процессоров. Intel Purley будут установлены и в Apple iMac Pro, выпуск которого запланирован на декабрь.



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!**



ИССЛЕДОВАНИЯ

# У дата-центров две беды

Расход электроэнергии и проблемы землепользования подталкивают Сингапур к «зеленым» вертикальным ЦОД.

**Н**ациональный ИКТ-регулятор Сингапура IMDA заключил соглашение о сотрудничестве с Huawei и Kerrel по «изучению технической возможности» строительства высотных (более 20 этажей) «зеленых» зданий для Центров обработки данных. IMDA, в рамках программы Green Data Centre Programme, на два года обеспечит исследованиям поддержку (в т. ч. экономическую), Huawei предоставит технологии и технические решения, Kerrel будет строить и эксплуатировать объекты. Предполагается, что высотное строительство позволит сократить энергопотребление и улучшить коэффициент PUE сингапурских ЦОДов на 10–20%. Концепция «тропического ЦОД» означает его эффективную работу при температурах до 38° Цельсия и влажности 90% и более,

снижение энергопотребления до 40% и сокращение выбросов CO<sub>2</sub>. По данным Forrestier и Broadgroup, в Сингапуре расположены 50% мощностей ЦОДов Юго-Восточной Азии, которые в 2015 г. потребляли 9% электроэнергии страны, а к 2020 г. это будет 12%. Типовой ЦОД на 20 МВт требует здесь столько же энергии, как и 60 тыс. домохозяйств; для его размещения нужен 1 га (площадь 3–4 жилых домов) и здание из 6–8 этажей. Поэтому мощности местных ЦОДов ограничены 20–25 МВт, или 5 тыс. серверных стоек (для сравнения, крупнейший дата-центр мира в Неваде имеет мощность 130 МВт, занимает 13 га и поддерживает до 26 тыс. стоек). Лучшие в своем классе многопользовательские ЦОД Сингапура имеют минимальный PUE 1.44, а многоэтажный ЦОД в Неваде — 1.185.

НАКОПИТЕЛИ

# «Викинги» в дата-центре

Компания Viking Technology выпустила «самый емкий в мире» SSD на 50 ТБ.

**К**омпания Viking Technology (подразделение Sanmina Corporation, США) представила на корпоративном рынке новую серию накопителей Ultra High-Capacity (UHC) Silo Solid State Drive, состоящую из двух SSD емкостью 25 и 50 ТБ (последнее — новый мировой рекорд среди высокоемких SSD). Особое внимание уделялось энергоэффективности: энергопотребление при простое составляет менее 10 Вт, а при активной работе — всего 16 Вт. В Viking Technology увеличили показатель общей емкости хранилища на стойку при уменьшении потребления электроэнергии на 1 ТБ. Новые Viking UHC-Silo SSD, в стандартном форм-факторе 3,5" и с интерфейсом SAS на 6 Гб/с, не для потребительского рынка, а в первую очередь, для ЦОДов: экономия электроэнергии, пространства и затрат на охлаждение до 80% на 1 ТБ. Дуэт новых SSD использует флэш-память Planar MLC NAND и обеспечивает скорости последовательного чтения/записи 500 МБ/с и 350 МБ/с соответственно, произвольного чтения/записи — до 60 тыс.



IOPS и 10 тыс. IOPS. Показатели по современным меркам достаточно скромные, поскольку интерфейс NVMe уже обеспечивает скорости свыше 1 Гб/с, однако Viking Technology делает упор на надежность и долговечность своих накопителей: компания утверждает, что ее SSD способны выдерживать ежедневную полную перезапись диска в течение пяти лет.

«Сегодня не существует более емкого SSD-решения, чем UHC-Silo SSD, — говорит президент компании Хамид Шокргозар [Hamid Shokrgozar]. — Поскольку пространство и охлаждение — критические параметры для современных ЦОД, наши преимущества меняют правила игры». **LXF**

**» Новый накопитель от Viking Technology позволит ЦОДам на 80% снизить затраты на охлаждение.**

Новости короткой строкой

**»** Президент РФ утвердил изменения в ФЗ «Об информации...», обязывающие владельцев средств анонимизации обеспечить блокировку ресурсов из реестра запрещенных в РФ.  
Источник: [ria.ru](http://ria.ru)

**»** Плагин Paragon NTFS USB Android от российской компании Paragon Software Group дает мгновенный доступ к видеофайлам 4K и HD на внешнем диске через Android TV или Android TV Box.  
Источник: [www.paragon.ru](http://www.paragon.ru)

**»** Объединяя идентичные XML-блоки, LibreOffice 5.4 уменьшил на 50% размер документов ODF (ODT), на 90% — OOXML (DOCX), по сравнению с MS Office.  
Источник: [blog.documentfoundation.org](http://blog.documentfoundation.org)

**»** Министр связи РФ сообщил, что в 2018 г. в Москве могут пройти первые испытания сетей связи 5G на беспилотном транспорте.  
Источник: [minsvyaz.ru](http://minsvyaz.ru)

**»** Организация USB 3.0 Promoter Group, разработчик USB-стандарта, представила черновик спецификации USB 3.2. Выход финальной версии планируется на сентябрь.  
Источник: [www.techpowerup.com](http://www.techpowerup.com)

**»** Adobe объявила Flash устаревшей технологией; обновление Flash Player прекратится в конце 2020 г.  
Источник: [blogs.adobe.com](http://blogs.adobe.com)

**»** В протоколе Kerberos выявлена существующая 21 год уязвимость [bypass bug], названная Orpheus Lyre [Лира Орфея]; она позволяет обойти криптографическую аутентификацию, как Орфей, проскользнувший мимо Церберы.  
Источник: [www.orpheus-lyre.info](http://www.orpheus-lyre.info)

**»** Билайн предложил сервис VEON, с функциями мультимедиа-сервиса, платформы общения и маркетинга; для клиентов «Билайна» трафик внутри VEON не тарифицируется.  
Источник: [moskva.beeline.ru](http://moskva.beeline.ru)

**»** Jide Technology сворачивает свои проекты для конечных потребителей, в т. ч. совместную с проектом Android-x86 разработку окружения Remix OS.  
Источник: [www.jide.com](http://www.jide.com)

**»** Проверена совместимость «1С: Предприятие 8» с СУБД Postgres Pro, что позволит внедрять чисто российские решения по автоматизации управления и бухгалтерского учета.  
Источник: [postgrespro.ru](http://postgrespro.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**

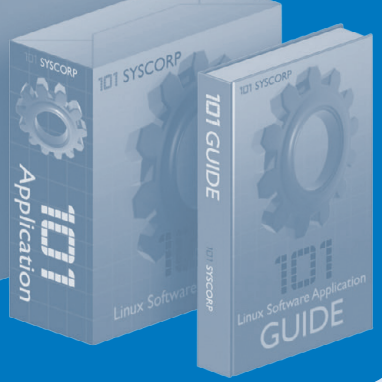
**Атрибутика**

## 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.

## Быть root'ом в KDE

До недавнего времени в KDE легко запускались приложения графического режима с привилегиями администратора — и через специальные утилиты вроде *kdesu* и *kdesudo*, и прямо из файлового, скажем, менеджера. И так было в KDE, начиная как минимум с «двойки». А вот в «пятерке» давеча вдруг пропало. И из *Dolphin* исчезли опции Open as root и Edit as root, и те же *Dolphin* или *Kate* отказались запускаться по команде *kdesudo kate*. Сначала это обнаружилось в «референсных» сборках проекта KDE Neon, а потом и многие другие дистрибутивы поднялись. Ибо оказалось это генеральной линией партии разработчиков KDE, вызванной к жизни их неуклонной заботой о пользователе. Вот и стали они «причинять ему добро налево и направо, нанося пользу и подвергая ласкам», запрещая совершать действия, народу не нужные.

Благо, майнтейнеры не всех дистрибутивов за генеральной линией последовали. Так, создатели дистрибутива Maui, хоть и основывались на «референсном» Неон, сохранили и утилиту *kdesudo*, и плагин Root Actions Servicemenu, позволяющий получать права администратора непосредственно из пользовательского сеанса *Dolphin*. А затем «патченные» версии *Dolphin* и *Kate* с *KWrite*, которые можно было запустить от имени root, появились и в ряде других дистрибутивов — сначала в Antergos и openSUSE, а нынче, вероятно, и в более иных.

Однако Maui в этот ряд попал одним из первых — и потому о нем в следующей колонке.  
[alv@posix.ru](mailto:alv@posix.ru)

## Сегодня мы рассматриваем:

**Slackel 7.0 Openbox** .... 14  
Скрестив супер-надежный Slackware с супер-лаконичным Salix, разработчик получил гибрид непонятно на чей спрос — но легкий и быстрый.

**OBRevenge 2017.05** ..... 15  
В этом дистрибутиве вы по умолчанию получаете голый минимум ПО, но населить его приложениями несложно.

**Skype для Linux Beta** ..... 16  
*Skype* для Linux — скорее web-приложение, чем полноценное настольное. Увы, Microsoft не предоставляет

никаких гарантий, что оно будет развиваться дальше.

**AMD Ryzen 5 1500X** .... 17  
Хотя сам по себе дешевый для своих характеристик, этот процессор потребует инвестиций в материнскую плату и ОЗУ. И тем не менее его время еще придет.

**Sapphire Radeon RX 580 Nitro+** ..... 18  
Графический процессор с разумным соотношением цена–качество увеличил производительность игр на 10%. А глядя на него, этого трудно было ожидать...

**Kentucky Route Zero** ..... 19  
Игра не забивает вам голову философией, а скорее завораживает. В ее медитативном мире вы сотворяете своего героя таким, каков он есть — в зависимости от своих выборов.

**Lords of Xulima** ..... 20  
История в меру пафосная, довольно предсказуемая, но добротная. Здесь нет абсолютно хороших и абсолютно плохих персонажей — у каждого своя правда... Совершаемые ошибки можно компенсировать своим временем и аккуратностью.

### OBRevenge 2017.05



› В отличие от большинства коллег, в OBRevenge по умолчанию не слишком много инструментов.

### Lords of Xulima



› Увы, от драконов остались только кости, а иначе сколько бы можно было на них заработать опыта...

## Сравнение: VPN-сервисы с. 24

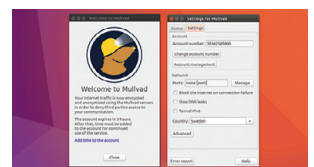
### Buffered VPN



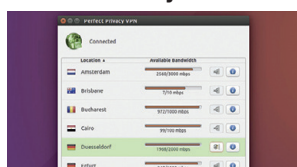
### IVPN



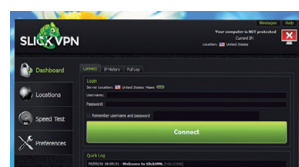
### Mullvad



### Perfect Privacy



### SlickVPN



То и дело всплывает новая история о том, как за всем миром следят и осуществляют мониторинг онлайн. Виртуальные частные сети способны помочь этому горю. И их много, хороших и разных.

# Slackel 7.0 Openbox

Предыдущий релиз не слишком впечатлил **Шашанка Шарму**, и он очень внимателен к этому выпуску: не изменилось ли что-нибудь...

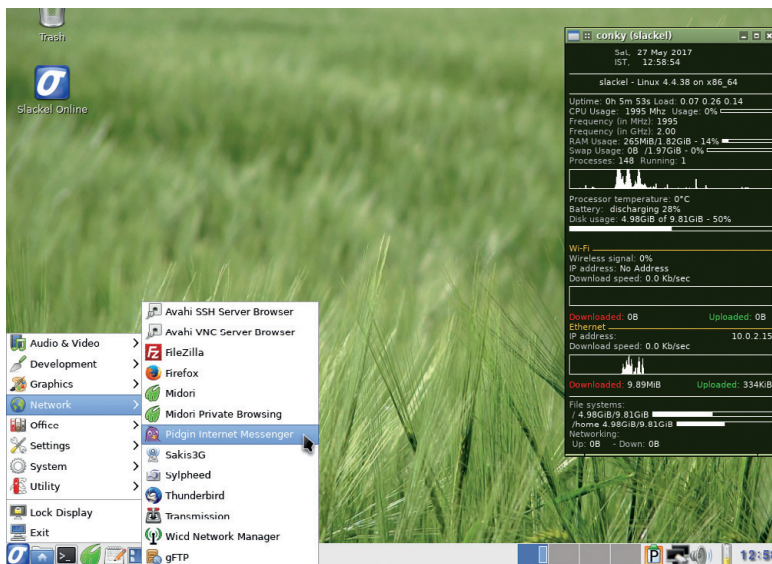
## Вкратце

» Этот дистрибутив основан на текущей ветке Slackware и Salix, который сам по себе является производным от Slackware. Выбор рабочего окружения *Openbox* делает дистрибутив идеальным для маломощных машин, а в последнем выпуске есть новый графический установщик.

**П**роба новой версии дистрибутива, который вы ранее оценили неблагоприятно, вызывает странное чувство. Опасения в очередной раз разочароваться смешиваются с потребностью найти какие-то сдвиги с предыдущего выпуска. К счастью, Slackel 7.0 многому научился на прошлых огрехах. Свежая версия, основанная на Salix и текущей ветке Slackware, является значительным улучшением версии 6.0 (см. Обзоры **LXF184**).

Для начала, дистрибутив теперь поставляется с графическим установщиком. Он минималистичен: всё, что у вас есть — это окно для ввода соответствующих данных: имя пользователя, пароль, пароль root. Кнопка Partitions в левом верхнем углу запускает *GParted*, чтобы помочь вам освободить место для дистрибутива. Можно также выбрать один из трех режимов установки с помощью переключателей внизу: Basic — для минималистичной системы с графическим интерфейсом и всего с несколькими приложениями; Core — для консольной системы с выбором устанавливаемых пакетов; и Full, чтобы установить сразу всё предлагаемое. В зависимости от выбранного режима установка занимает от трех до пяти минут. Этот минимальный установщик также предлагает выбор загрузчика *Grub* либо *Lilo*, но не задает докучливых вопросов о раскладках клавиатуры и часовых поясах, сразу устанавливая GMT.

Выбрав *Openbox* вместо полноценной среды рабочего стола, предыдущая версия также придерживалась легковесных приложений — например, браузера *Midori* вместо более популярных альтернатив вроде



» После прошлого релиза дистрибутив улучшился, но и слегка забуксовал.

*Chrome* или *Opera*. Но последняя версия 7.0, в дополнение к *Midori* поставляется с *Firefox*, полным комплектом *LibreOffice*, *Thunderbird*, *Gimp* и многими другими повседневными интернет- и мультимедиа-приложениями.

## Работа продолжается

Хотя Slackel теперь богаче функциями, он страдает от проблем с производительностью, в основном из-за своего смешанного наследия. Slackware — один из самых стабильных и надежных дистрибутивов Linux, и вполне естественно ожидать того же от его производных. Salix, с другой стороны, стремится добавить простоту в работе поверх стабильности Slackware.

Slackel поставляется с новейшим ПО из текущей ветки Slackware и рядом инструментов из Salix OS, таких как *codec-installer* и менеджер пакетов *Gslapt*. В отличие от прошлого выпуска, поведение *Gslapt* неустойчиво: не раз он неожиданно падал во время наших тестов. *Sourcery*, другой графический инструмент, заимствованный у Salix OS, отказывается запускаться из-за отсутствия модулей. Да еще *Grub* распознает дистрибутив как Slackel 14.2, а не 7.0.

В отличие от большинства других дистрибутивов Linux, со штатным персоналом, работающим над внешним видом и художественным оформлением, здесь единственный разработчик дистрибутива оставляет те же обои, что и в предыдущем выпуске. Хотя это и мелочь, но наводит

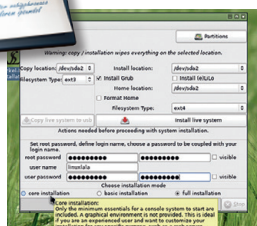
на мысль о том, что дистрибутив выпущен без учета возможных прорех в защите.

Возможно, дистрибутив чересчур обидел, гоняясь за простотой в работе Salix OS. Да, Slackel проделал большой путь в надежде стать желанным выбором для новичков, но в нынешней экосистеме Linux, которая опирается на простые в использовании дистрибутивы плавающего релиза, потребность в нем, по-видимому, невелика.

Однако, если вы начинающий или даже ветеран в Linux и желаете легкий и быстрый дистрибутив с рядом полезных повседневных приложений, способный работать с различными устройствами Wi-Fi и Bluetooth, то Slackel как раз для вас. **LXF**

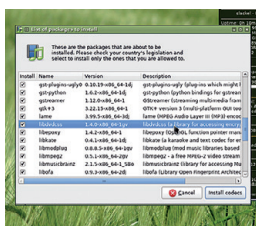


## Свойства новскидку



### Графический установщик

Однооконный установщик охватывает все основные элементы установки Linux.



### Установщик кодеков

Slackel заимствовал у Salix OS инструмент *codec-installer* для установки мультимедиа-кодеков.

## LINUX FORMAT Вердикт

### Slackel 7.0 Openbox

Разработчик: Димитрис Земос  
[Dimitris Tzemos]  
Сайт: [www.slackel.gr](http://www.slackel.gr)  
Лицензия: GPL и другие

Функциональность	9/10
Производительность	7/10
Удобство в работе	7/10
Документация	6/10

» Проект, «заточенный» на желания пользователей и прилагающий к этому все усилия.

## Рейтинг 7/10



# OBRevenge 2017.05

Дистрибутив с любопытным названием — еще одна ветка Arch Linux. Вволю его оттестировав, **Шашанк Шарма** так и не смог объяснить его название...

## Вкратце

» Live-устанавливаемый дистрибутив рабочего стола на базе Arch Linux, оснащенный *Openbox*. Поставляется лишь с минимумом приложений, но может похвастаться несколькими пользовательскими инструментами, включая установщик ПО, который может вмиг установить все популярные приложения. Контрольная панель пользователя и многие настройки *Openbox* задумывались для быстрой и скоростной работы, даже на старых машинах.

**O**BRevenge [можно истолковать как «месть *Openbox*»] — новый дистрибутив на базе Arch Linux, который поставляется как Live-устанавливаемый 64-битный ISO с менеджером окон *Openbox*. Если вы любите более продвинутые рабочие столы, проект также производит *MateRevenge* и *KRevenge*, оснащенные *Mate* и *KDE* соответственно. Но здесь мы рассмотрим исключительно *Openbox*.

Со столь легким менеджером окна, обширными хранилищами ПО, стабильностью и гибкостью Arch Linux, дистрибутив может вдохнуть новую жизнь в слабые компьютеры, которым иначе осталось бы служить малопрактичными пресс-папье.

Инсталляция под управлением *Calamares* довольно проста: надо лишь выбрать раскладку клавиатуры и часовой пояс и предоставить имя пользователя/пароль. К тому же дистрибутив поставляется с *GParted*, чтобы помочь вам выделить на жестком диске место для OBRevenge. Установщик позволяет выполнять множество операций и поддерживает многие файловые системы.

При входе вы обнаруживаете уникальную картину системного монитора *Conky*, которая показывает CPU, сетевой трафик и потребление памяти. Также в дистрибутиве есть быстрый и легкий поиск *Synapse* и пускатель [launcher], пригодный для быстрого поиска файлов, будь то изображения, аудио, видео или документы, и даже запуска приложений или поиска в Интернет.

## Беритесь за настройку

В отличие от большинства коллег, в OBRevenge по умолчанию не слишком много инструментов. На самом деле, нет почтового



» С момента выхода OBRevenge уже прислал кучу исправлений и улучшений. Обязательно установите их для доступа к новым функциям.

клиента, браузера, офисных и мультимедийных приложений. Для заполнения своей установки послужит инструмент *Software Install*. Однако нет средства поиска пакетов, и не предложены пакеты для более узких категорий. Например, чат и клиенты IRC, как и приложения сетевого мониторинга, все размещены под вкладкой Интернет.

Уже через пару недель после релиза проект выпустил более 600 новых обновлений. При первой установке обновлений надо выполнить `raspm -Syu` в командной строке, так как установка обновлений с помощью графического инструмента создает ошибки неразрешимых конфликтов пакетов. При запуске *raspm* этот умный инструмент управления пакетами автоматически предлагает заменить конфликтующие пакеты на альтернативы от сообщества или из добавочного репозитория, а затем производит обновление.

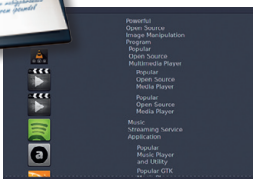
Дистрибутив предоставляет ассортимент собственных инструментов, со множеством полезных функций. Так, утилита *Autostart* пригодна для задания приложений, вызываемых при входе в систему. Панель управления обеспечивает быстрый доступ ко всем настраиваемым элементам рабочего стола. По умолчанию из вкладки Настройки можно переключать тему, обои, изменять параметры уведомлений и многое другое. Аналогично, вкладка Система отображает настройки сети и дисплея, тогда как вкладка Программное обеспечение дает быстрый доступ к настройке брандмауэра, установке ПО и обновлению инструментов. Наконец, вкладка OBR-Tools перечисляет все пользовательские инструменты дистрибутива: менеджер ядра, установщик live

USB, драйверы *VirtualBox* для управления гостевыми модулями, драйверы *nVidia*, установщики кодеков и многие другие.

Помимо краткого руководства по началу работы (немногим более, чем список функций), дистрибутив также может похвастаться форумом. Это молодой дистрибутив, поэтому форум еще не заполнен запросами и ответами, но, будучи ответвлением от Arch, OBRevenge не испытывает недостатка в документации и поддержке сообщества.

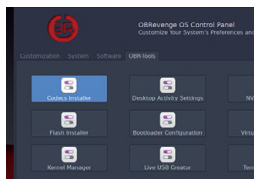
Если отвлечься от ограниченности набора ПО, установленного по умолчанию, и причудливого поведения инструмента графического обновления, рабочий стол OBRevenge безупречен. Хотя предполагается, что это Live-устанавливаемый дистрибутив с со множеством инструментов пользователя, OBRevenge остается верным минималистичной природе Arch. **LXF**

## Свойства навскидку



### Установщик ПО

Обязательно обновите приложение. В версии по умолчанию не показан значок или отметка уже установленных приложений.



### Утилиты пользователя

Пользователи могут установить ПО одним щелчком, без ошарашивания умопомрачительным количеством опций.

## LINUX FORMAT Вердикт

### OBRevenge 2017.05

**Разработчик:** OBRevenge Team  
**Сайт:** <http://obrevenge.weebly.com>  
**Лицензия:** Различные

<b>Функциональность</b>	9/10
<b>Производительность</b>	8/10
<b>Удобство в работе</b>	10/10
<b>Документация</b>	8/10

» Здесь многое и для новых, и для опытных пользователей Linux, стремящихся выйти за пределы Gnome и KDE.

**Рейтинг 9/10**

# Skype для Linux Beta

Нейт Дрейк любопытствует, выполнила ли компания Microsoft свое обещание по выпуску Skype для Linux.

## Вкратце

» В прошлом году Skype выпустил Alpha-версию программы для Linux, а затем, в марте 2017 г., версию Beta 5.0. Сейчас программа находится в версии 5.1.0.1, хотя по сути это хваленый интерфейс для web-приложения Skype. Новые функции включают вызов обычных телефонов, а также видеозвоны других Skype-платформ.

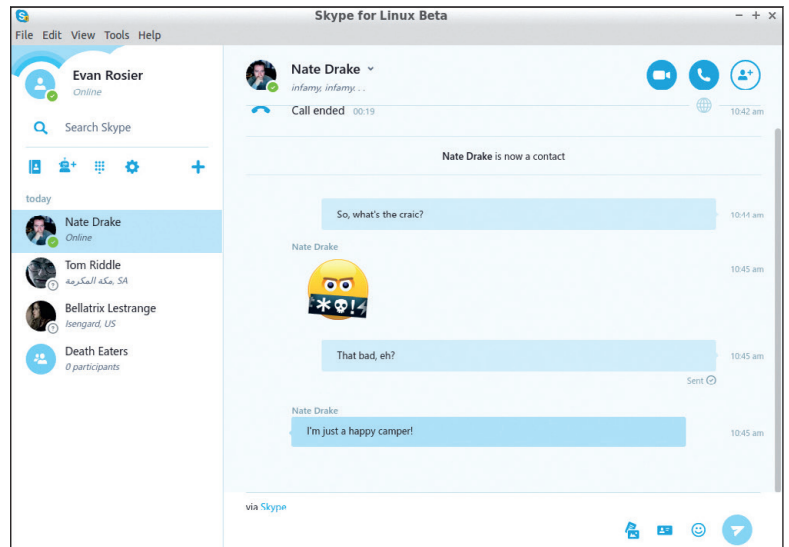
С о времен последнего обзора Skype для Linux Alpha в LXF216, была выпущена Beta-версия 5.1.0.1. По собственным словам Microsoft, в ней отражены «усилия по переходу Skype от одно-ранговых узлов к современной, мобильно-дружелюбной облачной архитектуре».

На человеческом языке это означает, что Skype для Linux — скорее web-приложение, чем полноценное настольное. Пройдите на сайт Skype для или Deb или RPM, в зависимости от ваших пристрастий в Linux, и начинайте работу.

Сейчас Skype для Linux Beta вдобавок поддерживает индивидуальные звонки пользователей на таких платформах, как Windows и Android. Этой функции не было в Alpha-версии. Для наших сравнительных тестов мы делали видеозвонки из Ирландии к пользователю в Таиланде, загружая Skype на Windows XP SP3, и убедились, что у Skype для Linux и Skype для Android нет разницы в качестве.

Пользователи Ubuntu также будут рады услышать, что при запуске Skype Unity теперь отмечает количество непрочитанных разговоров. Правда, поскольку Canonical объявила о планах убрать Unity в пользу Gnome, непонятно, как эта функция будет поддерживаться дальше. Другое очень значимое дополнение — то, что у списка контактов Skype для Linux теперь есть статусы Away [Отсутствует] и Do Not Disturb [Не беспокоить].

Совместный доступ к экрану поддерживается в ограниченной степени. Пользователи ОС Windows и Mac, работающие



» Сейчас Skype поддерживает передачу смайлов, чаты, фотографии, а также видеозвонки. Чтобы делать звонки на стационарные телефоны, пополните свой Skype-кредит.

с относительно свежей версией Skype, могут поделиться своими экранами с вашей Linux-машиной, хотя досадно, что процесс не работает в обратную сторону.

## Skype-кредит

Самые яркие и многообещающие улучшения включают поддержку использования вашего Skype-кредита для звонка на мобильные и стационарные телефоны. Просто нажимаете на панель набора номера, а затем жмете Add Skype Credit [Добавить кредит Skype]. Экран набора также любезно сообщит, что Skype для Web (Beta-версия) не осуществляет экстренных вызовов. Это недвусмысленное напоминание, что, как и Alpha, Skype для Linux — просто интерфейс web-приложения, доступного на <http://web.skype.com>.

Вызов обычных телефонов теперь поддерживается, потому что функция добавлена в web-приложение. Аналогично, опция видеозвонов другим пользователям Skype, использование YouTube для просмотра видео или предложение на выбор смайлов поставляется в контексте web-приложения. Отправка коротеньких видеосообщений в чате временно не поддерживается в web-версии.

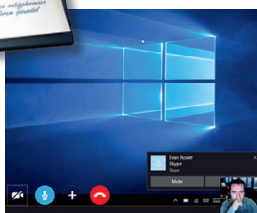
Учитывая, что пользователям Linux, в основном, охота реже шевелить руками, чем пользователям Windows или Mac, раз уж дополнительные функции в наличии есть, нет нужды специально готовить

бинарники для загрузки в ОС, когда можно пройти по web-ссылке. При тестировании единственное заметное различие, которое мы обнаружили, заключается в том, что версия браузера Skype не работает с таким расширением, как uBlock Origin, и что загружаемая версия для Linux имеет лучшую поддержку уведомлений.

Ранее компания Microsoft прекратила поддержку Linux, а значит, нет гарантии, что Skype для Linux будет развиваться дальше Beta-стадии. Но если Microsoft сдержит обещания по развитию своих web-приложений, то пользователи Linux смогут звонить, болтать и обмениваться сообщениями с остальными. LXF

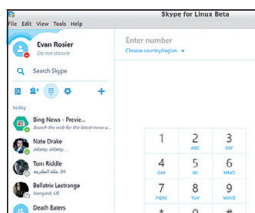


## Свойства новскидку



### Совместный доступ к экрану

Во время разговора пользователи Windows и Mac теперь могут делиться своими рабочим столом с пользователями Linux.



### Мобильные и стационарные телефоны

Для звонков на обычные телефонные номера добавьте Skype-кредит. Для оплаты звонков обратитесь к сайту Skype.

## LINUX FORMAT Вердикт

### Skype для Linux Beta

Разработчик: Microsoft

Сайт: [www.skype.com/en](http://www.skype.com/en)

Лицензия: Ограничена, не подлежит переуступке

Функциональность 7/10

Производительность 8/10

Удобство в работе 8/10

Документация 7/10

» Skype для Linux сидит на двух стульях. Не удовлетворит ни случайных, ни продвинутых линуксоидов.

Рейтинг 7/10

# AMD Ryzen 5 1500X

Хватит ли жалких четырех ядер и восьми потоков для счастья Джереми Лэйрда с экстремальным процессором на уме? Вдобавок это тоже чип AMD!

## Спецификация

- » Разъем AM4
- » Разрядность 64 бит
- » Тактовая частота 3,5 ГГц
- » Турбо 3,7 ГГц
- » Ядер 4
- » Потоков 8
- » Техпроцесс 14 нм FinFET
- » Кэш L1 384 кБ, L2 2 МБ, L3 16 МБ
- » Память DDR4-2666 (64 МБ макс.)
- » Каналов 2
- » PCIe 16 + 4 + 4
- » Тепловыделение 65 Вт
- » Виртуализация AMD-V, AMD-Vi

**Б**ольшее количество ядер, больше потоков и больше производительности за меньшие деньги. Это убойное предложение от нового процессора AMD Ryzen. Внезапно большинство настольных процессоров Intel оказались неоправданно дорогими и неэффективными.

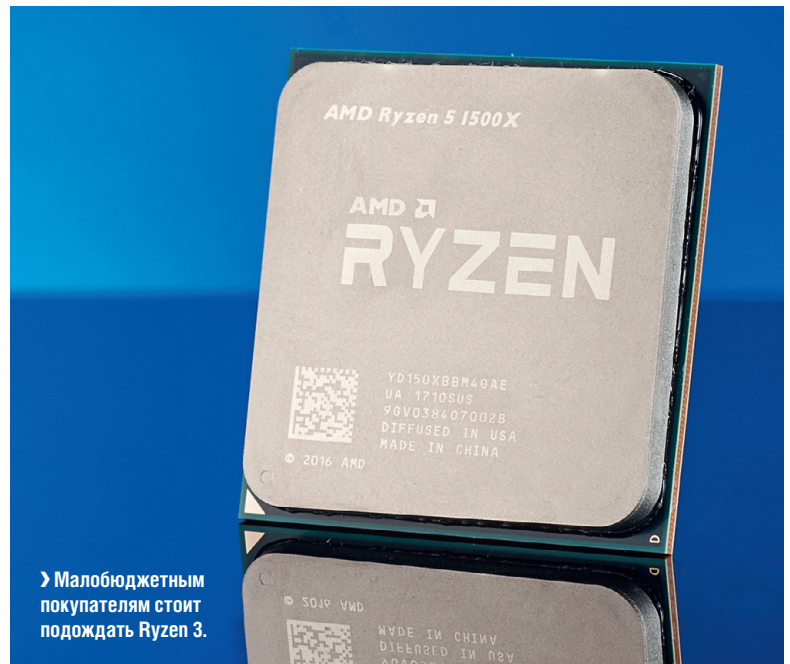
Но распространяется ли такой взгляд на новый Ryzen 5 1500X с его четырьмя ядрами и восемью потоками? В конце концов, у Intel есть множество всяческих четырехъядерных чипов на выбор. И как бы несомненно ни был хорош Ryzen, Intel по-прежнему имеет преимущество по удельной мощности на ядро.

Сперва 1500X выглядит конкурентоспособным, если не вовсе нокаутирующим вариантом шести- и восьмиядерных моделей. £175 примерно соответствует Intel Core i5-7400. Это четырехъядерная модель с четырьмя потоками и базовой частотой 3 ГГц и 3,5 ГГц частотой турбо. Чипы Ryzen напрямую не сопоставимы с точки зрения тактовой частоты, но при базовой частоте 3,5 ГГц и турбо 3,7 ГГц, 1500X кое-что имеет для компенсации небольшой недостачи на такт Ryzen в сравнении с лучшими Intel. Ryzen также полностью разблокирован, что теоретически допускает еще большее преимущество в частоте. Intel, напротив, блокирует большинство своих чипов, закрывая дорогие модели серии К.

Можно, конечно, возразить, что если уж у вас Intel, надо быть сумасшедшим, чтобы не взять Core i5-7500, всего на £10 дороже, с базовой частотой 3,4 ГГц и турбо 3,8 ГГц. И, как мы увидим, именно такой расчет в конечном итоге губит 1500X. Но мы немного забежали вперед...

Как и у других чипов Ryzen, которые мы перепробовали, многопоточность — его главный козырь. В тестах, сильно зависящих от потоков, он побивает гораздо более дорогой Intel Core i5-7600K, не говоря уже о более дешевых моделях Core i5. 1500X быстрее, чем 7600K и на рендеринге, пусть и с довольно небольшим отрывом, хотя они более или менее равны в кодировании видео в x265. Другими словами, он лучше процессоров Core i5-7400 и 7500 той же ценовой категории во всем, где уместна многопоточность.

Исключением, как всегда, является игра. Многие игры обычно зависят от GPU в деталях и разрешении, что, вероятно, существенно для заядлых геймеров. Но когда



» Малобюджетным покупателям стоит подождать Ryzen 3.

игра ограничена процессором, что имеет место даже при высоких разрешениях в некоторых случаях, чипы Intel по-прежнему в лидерах, хотя стоит отметить, что этот четырехъядерный Ryzen обеспечил такую же частоту кадров в наших тестах, как и шестиядерный 1600X.

## Ахиллесова пята

Со временем, когда разработчики игр обдумают нюансы новой архитектуры Zen AMD, этот разрыв может сократиться. Пока, правда, для Ryzen игры — очевидное слабое место, и для 1500X, и для любой другой модели.

Почти везде Ryzen чертовски быстр. 1500X также впечатлил в относительном смысле в наших тестах разгона. Мы просто получили стабильные 4,1 ГГц, или 400 МГц разгона. Не то чтобы очень зрелищно, но полезное 10-процентное ускорение нам пригодится.

Резонный вопрос: стоит ли экономия £25 по сравнению с шестиядерным Ryzen 1600 и £65 — с немного более быстрым 1600X? С последним вы получаете на 50% больше ядер, и несколько большую тактовую частоту. Главный момент здесь то, что с любым процессором Ryzen вам понадобится новая материнская плата и оперативная память. Это означает, что минимальная сумма инвестиции составит около

£500. В таком контексте, не будет ли доплата £25 или £65 за шестиядерную модель и гораздо более высокую производительность безусловно лучшим выбором?

С учетом всего этого, Ryzen — не альтернатива Intel, которая выбивает почву из-под 1500X, а декларация ценности шестиядерных моделей. Через несколько лет, когда у некоторых из нас будут системы с новым гнездом AM4 AMD, расчет стоимости будет выглядеть иначе. Но на данный момент, с учетом общей стоимости перехода на Ryzen, этот конкретный чип 1500X является навязчивой рекламой. **LXF**

## LINUX FORMAT Вердикт

### AMD Ryzen 5 1500X

Разработчик: AMD  
Сайт: [www.amd.com/ryzen](http://www.amd.com/ryzen)  
Цена: £175

Функциональность	8/10
Производительность	9/10
Удобство в работе	8/10
Оправданность цены	7/10

» Явно быстрее, чем предлагает Intel за эту цену, но более быстрые модели AMD еще лучше за чуть большие деньги.

**Рейтинг 8/10**

# Sapphire Radeon RX 580 Nitro+

«Зачему, ну зачем, AMD!» — причитает **Зак Стори**, тестируя последнюю разработку графического процессора.

## Спецификация

- » Графический процессор Polaris
- » Техпроцесс 14 нм FinFET
- » Вентилей 5,7 млрд
- » Потоков 2304
- » Блоков текстур 144
- » Блоков блендинга 32
- » Тактовая частота 1,34 ГГц
- » Тактовая частота при разгоне 1,411 ГГц
- » Память 8 ГБ GDDR5
- » Тепловыделение 185 Вт
- » Разъемы DisplayPort 1.4, HDMI 2.0b, DVI-D



» Мечтаете о 10-процентном повышении производительности на 1080p? Тогда эта карта для вас.

**Д**айте сразу к делу. Это не новая архитектура. На самом деле в Radeon RX 580 вообще очень мало нового. Выпуск этого нового GPU прошел без фанфар и, откровенно сказать, на полном основании: это не новая технология Vega AMD.

Коротко говоря, переход от RX 480 к 580 предлагает не более чем небольшой разгон базовой карты. Собственно, можно было бы достичь тех же результатов, задавшись целью повысить мощность до 110 процентов и чуть увеличить тактовую частоту. Тогда чего ради это делалось?

Хороший вопрос. В общем-то, мы можем только догадываться, что это связано с улучшением производственного процесса. Возможно, в серии RX 500 GloFo использует технологию 14 нм LPU Samsung, повышая энергоэффективность и тактовую частоту в работе. Чем и объясняются дополнительные 30 Вт рассеяния тепла (TDP) на плате и чуть более высокая частота.

Кроме того, это привлекающая нас система обозначений. Традиционно в каждой новой серии карт есть одна или две смены обозначений. Например, 380 становится 470, 370 становится 460, и т. д. Снижение цены и повышение энергоэффективности чрезвычайно радуют потребителя. Это потому, что карты с хорошей

производительностью, отличавшиеся запредельной стоимостью, внезапно становятся более доступными.

А вот 580 пропустил такую смену. RX 480 перенесен на 580. Оно было бы и разумно, если AMD собиралась начать новую схему обозначений или хотя бы выпустить два новых флагмана с этими переобозначениями; но это вовсе не так. И что же происходит? Если уж переименовывать эти две карты, то они должны были бы стать 485 и 475. Особенно, когда исключаются какие-либо упоминания вариантов.

Но довольно брюзжать. Как же это выглядит с точки зрения производительности? Итак, мы сравнили наш образец RX 580 — предварительно разогнанный вариант Sapphire Nitro+ — с имевшимся у нас RX 480. Как правило, в играх на 1080p мы наблюдали увеличение на 10 процентов. Время от времени мы видели скачки, например, с 34 кадр/с до 40. Впечатляюще — для, по сути, просто слегка перетряхнутого GPU.

## Кое-что приятное

Кому предназначена эта карта? Коротко — всем, кто еще не обновился с серии AMD R9 300 или с нижнего конца серии GTX 900 Nvidia. Это перефразирование уже хорошо оцененной графической архитектуры,

которая задает традицию отношения цена—производительность в 2017 г. Фактически, разумное такое отношение — само по себе приятность. Карта не совсем спихнула Nvidia GTX 1060 3GB с вершины 1080p, но близка к этому. И с такой добавочной производительностью? Заманчиво, мягко говоря. Соедините его с приличным монитором FreeSync 1080p, и вы станете победителем. Но предупреждаем, что в какой-то момент 2017 г. появится навороченный Vega GPU следующего поколения. **LXF**

## LINUX FORMAT Вердикт

### Radeon RX 580 Nitro+

Разработчик: Sapphire  
Сайт: [www.sapphiretech.com](http://www.sapphiretech.com)  
Цена: \$ 250

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

» Нового здесь не видно, но стало экономически выгодно и дает полноценную производительность игры на 1080p для всех.

**Рейтинг 8/10**

# Kentucky Route Zero

«Когда же оно кончится?» недоумевает **Филлиппа Уорр**, доигрывая четвертый из пяти эпизодов этого магического реализма с элементами театральности.

## Спецификация

- » ОС SteamOS, Ubuntu
- » Процессор 1 ГГц
- » ОЗУ 512 МБ
- » GPU OpenGL 3.0+
- » Диск 250 МБ

Окунитесь в потусторонний мир, где продавец-развозчик антиквариата ищет адрес, которого, судя по всему, в природе нет. Единственный возможный путь туда — такая же псевдосуществующая магистраль: Route Zero [Маршрут Ноль]. Далее вас ждет похожее на сон путешествие по ночной пригородной местности, выполненной в псевдовекторном стиле, под завораживающий аккомпанемент пространственной музыки.

Головоломка как таковых здесь нет, за исключением понимания смысла происходящего или загадок, предлагаемых вашими случайными спутниками. Персонажи здесь ускользают, словно призраки; радио в заброшенной церкви издает хоровую музыку; поворот на шоссе обозначен горящим деревом, а глубину забытой шахты оглашает звон старого громкоговорителя, словно напоминание о минувшей катастрофе.

Услышав применительно к *Kentucky Route Zero* такое описание, как «магический реализм», можно вообразить, что вас ждет какое-то путешествие туда-не-знаю-куда за тем-не-знаю-чем. И хотя отчасти так и есть, создатели игры сделали так, что всё здесь тонко, продумано и со смыслом.

Игра не забивает вам голову философией, а скорее завораживает вас с того самого момента, как грузовик Конвэя [Copway] останавливается на заправке. Текстовые действия лишены традиционного минимализма языка укажи-и-щелкни, ради



» В *Kentucky Route Zero* всё не то, чем кажется. Но в этом что-то есть.

большей интуитивности. Внутригровой компьютер, активируемый нажатием символа, а не выбором синтаксиса «используйте x и z», не просто банально включается — скорее, он «выходит из забвения» и требует в качестве пароля фрагмент белого стиха, который вы сами выстраиваете из последовательности вариантов.

Дополнением к текстовым действиям являются анимированные визуальные переходы, переводящие вас из одного пространства в другое. Так, попытка настроить неисправный телевизор приводит к тому, что стена позади него разрушается, открывая ночной пейзаж, который также изображен на маленьком поврежденном ящике. А отдаленные звуки музыки или треск сверчков завлекают вас всё дальше в этот странный, озаренный луной мир.

В полной мере медитативный мир *Kentucky Route Zero* раскрывается к середине первого эпизода. Ориентируясь по отзвукам громкоговорителя, Конвэй исследует пространство заброшенной шахты. Игрок выбирает варианты диалога, а Конвэй бормочет в микрофон, и независимо от того, что вы выбираете, в темноте раздаётся стон. Выбор не влияет на развитие игры и на последующие события. Он просто помогает вам сотворить Конвэя таким, каким вы его видите.

Пожалуй, именно то, насколько игра позволяет игроку погрузиться в себя, является главным триумфом *Kentucky Route Zero*. Но обилие поэтической образности порой сбивает с толку и даже кажется бессмысленным, далеким от причинных связей.

Местами темп повествования не согласуется с темпом игры — в моменты, когда физическое движение намеренно замедляется, испытывая ваше терпение, но ощущение замороженности не теряется, и ощущение разлада скоро отступает.

*Kentucky Route Zero* ведет вас по заданному маршруту, мало кем населенному и почти не оставляющему пространства для маневра. Однако в том, как объясняются эти ограничения и какие шаги в игре оказываются самыми важными, присутствует некая красота.

**Прим. редактора:** Этот оригинальный обзор был опубликован в январе 2013 г. на основе Эпизода I. Эпизод IV вышел в июле 2016 г., а Эпизод V еще ожидается, и в этом, как и в самой игре, есть своя театральная интрига... **LXF**

## LINUX FORMAT Вердикт

### Kentucky Route Zero

**Разработчик:** Cardboard Computer  
**Сайт:** [KentuckyRouteZero.com](http://KentuckyRouteZero.com)  
**Цена:** \$20

Сюжет	8/10
Графика	10/10
Увлекательность	7/10
Оправданность цены	7/10

» Прекрасная в своей неспешности игра, влекущая как томительным ожиданием каждого релиза, так и дизайном.

**Рейтинг 9/10**



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

# Lords of Xulima

По заключительным титрам подозрение **Евгения Балдина** подтвердилось: «Все боги — гады».

## Вкратце

» Классическая ролевая игра старой школы в современной упаковке. Казуальщикам проходить мимо.

**Е**ще один удачный проект с Kickstarter. Небольшая студия испанских разработчиков Numantian Games в 2014 г. скромно запросила всего \$10000 (получив в 3,5 раза больше) на создание RPG в стиле «старой школы». Школы еще более старой, чем бессмертные *Baldur Gates/Arcanum/Fallout*, но от этого не менее увлекательной, зато не требующей рекордных бюджетов. Независимые разработчики выполнили свои обещания без особых задержек, включая порт GNU/Linux, не ссылаясь, как некоторые (кивок в сторону Larian Studios), на то, что игровой движок писался вместе с игрой, а не взят уже готовый.

Как и во многих других ролевках, всё начинается с создания команды приключенцев. Вам навязывается главный протагонист — исследователь [Explorer] Гален. Именно с ним все разговаривают, умоляют его или проклинают, а остальные партии играют лишь роль его мускулов, важность которых раскрывается только в бою. Но так как вокруг пошагового боя тут всё и вертится, то к выбору партии нужно подходить очень ответственно. Здесь в бой идут одни люди, хотя на материке Шулима существуют и другие разумные расы. Выбирать можно из девяти классов. Кроме



» Вся игра строится вокруг боев. Ошибок противник не прощает, поэтому к каждому бою нужно подходить хорошо экипированным.

классических Бойца, Варвара, Мага, Клирика, Паладина, Барда и Вора, на выбор есть весьма удачный Заклинатель богов [Divine Summoner] и многими нелюбимый Магический боец [Arcane Soldier]. Кроме класса, для каждого сопартийца надо выбрать бога. Для бойцов первой линии стоит сделать выбор в пользу огненного бога гнева Valvet — усиление ловкости даст ранний доступ к двуручному оружию. Покровительство Леди диких животных Raznet пригодится Паладину или Барду, так как для наложения защитных заклинаний нужна возможность сделать ход пораньше. Если нет уверенности в выборе, можно присмотреться к богине ветра Taliet, ведь скорости много не бывает.

Я прошел игру в условно классическом составе. Первая линия — Гален, Паладин, Варвар, а вторая линия — Маг, Вор и Клирик. В порядке эксперимента можно попробовать в первую линию поставить Клирика, Галена, Варвара и Вора, а во вторую — Мага и Заклинателя (упор на аватары Raznet и Golot с заменой на Valvet в сложных боях).

Есть множество подходящих вариантов развития партии и еще больше способов ее угробить. Враги, особенно если их уровень выше уровня партии, ошибок не прощают и убивают неудачников весьма эффективно. Поэтому до начала выбора партии нужно либо начитать советов, либо смириться с тем, что первая попытка выйдет гарантированным комом и основной будет вторая или даже третья. Хотя некоторое количество ошибок можно компенсировать своим временем и аккуратностью. Например, не рваться сразу

поубивать злобных принцев, а для начала перебить всю их охрану. При досрочной победе же над боссом подчиненные трусливо сбегут и оставят вас без ценного опыта.

Материк Шулима велик — в смысле, меньше чем за 100 часов реального времени его обойти сложновато, хотя и интересно. История в меру пафосная, довольно предсказуемая, но добротная. Абсолютно хороших и абсолютно плохих персонажей здесь, в сущности, и нет. У каждого из них своя правда.

Игра очевидно требует еще некоторой полировки/стабильности/ускорения загрузки сохраненных игр. Идет работа с модами, так что игра еще долго будет актуальной, но уже хочется продолжения — в особенности потому, что финал остается открытым и к богам у людей остались вопросы. **LXF**

» Обязательный главный герой — исследователь, ибо исследовать есть что. Делать это постоянно мешают враги более высокого уровня.



## Свойства новскидку



### Лиха беда начало

Lords of Xulima — предположительно первая вступительная часть большой саги о взаимоотношениях людей и богов.



### Как прекрасен этот мир

Партия высаживается в большом открытом мире, гулять по которому придется не меньше 100 часов реального времени.

## LINUX FORMAT Вердикт

### Lords of Xulima

Разработчик: Numantian Games  
Сайт: [www.lordsofxulima.com](http://www.lordsofxulima.com)  
Цена: 419 руб. в Steam или gog

Сюжет	7/10
Графика	8/10
Увлекательность	7/10
Оправданность цены	8/10

» Пошаговые бои, бои и еще раз бои. В промежутке между боями можно полюбоваться окрестностями.

## Рейтинг 8/10

# 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!**



# Мобильные НОВОСТИ

К ЮБИЛЕЮ

## «Куриный» телефон

KFC выпускает... смартфон! KFC Huawei Commemorative Edition.

Приятная новость для почитателей полковника Сандерса и любителей жареных цыплят, и недорогих китайских смартфонов: в ноябре 1987 г. международная сеть экспресс-кафе KFC — бывшая Kentucky Fried Chicken [Жареный цыпленок из Кентукки], вторая по торговому обороту после McDonald's сеть кафе в мире — открыла свой первый «ресторан» в Китае, неподалеку от площади Тяньаньмэнь в Пекине. В ознаменование 30-летней годовщины этого события компания совместно с Huawei выпустила фирменный смартфон KFC Huawei Commemorative Edition: оформленный в стиле KFC металлический ярко-красный корпус с выполненными лазерной насечкой портретом полковника Сандерса на задней крышке и надпись

«1987», и предустановленные приложения Kentucky Super APP и K-Music (последнее позволяет пользователям создавать и делиться плей-листами в ресторане KFC).

Технически KFC Huawei Commemorative Edition является копией смартфона Huawei Enjoy 7 Plus. Его характеристики подобны родительской модели: экран диагональю 5,5 дюймов с разрешением 1280×720 пикс, четырехъядерный процессор Qualcomm Snapdragon 425 с тактовой частотой 1,4 ГГц, 3 ГБ ОЗУ, аккумуляторная батарея на 3020 мА·ч, 32 ГБ встроенной памяти с возможностью расширения до 128 ГБ за счет карт microSD, сканер отпечатков пальцев (надо полагать, жиронепроницаемый). Анонс KFC Huawei состоялся в китайской социальной сети Weibo;

► Видео на [youtu.be/id\\_l8IXzENk](https://youtu.be/id_l8IXzENk) демонстрирует продвижение обоих брендов с их скромного начала в 1987 г.

продажи модели, выпущенной ограниченной партией в 5 тыс. экземпляров, начались 13 июля китайским ритейлером Tmall по заявленной цене 1099 юаней (около £125 или \$162).



NOKIA

## Новый флагман

В топовые смартфоны Nokia возвращается оптика Zeiss.

После успешного дебюта Android-устройств среднего ценового диапазона Nokia 3, 5 и 6, компания HMD Global готовит к выпуску флагманскую модель Nokia 8 с оптикой Zeiss (в начале июля HMD и Zeiss заключили соглашение, призванное вернуть оптику Zeiss в телефоны Nokia). В середине июля модель Nokia 8 была продемонстрирована на официальном сайте производителя, а затем в Интернете появились ее реальные фотографии.

Презентация Nokia 8 в Лондоне, ранее запланированная на 31 июля, была перенесена на 16 августа. Телефон оснащен 5,3-дюймовым дисплеем с разрешением QHD (2560×1440 пикс) и восьмиядерным процессором Snapdragon 835, и будет поставляться с ОС Android 7.1.1 Nougat, двойной основной камерой с двумя сенсорами по 13 Мпикс, а комплектоваться до 6 ГБ ОЗУ и 64 ГБ встроенной памяти.

Процессор Snapdragon 835 позволяет отнести Nokia 8 к флагманскому классу, однако наличие на дисплее верхней и нижней рамок удивляет: на телефоне такого уровня более уместным был бы безрамочный полноэкранный дисплей. На нижней панели, между емкостными кнопками навигации, присутствует аппаратная кнопка Home, а также сканер отпечатков пальцев. Смартфоны Nokia компания HMD предоставляет ежемесячные обновления безопасности от Google.

Nokia 8 превосходит по производительности Nokia 6, работающую на Snapdragon 430, и при этом смотрится значительно дороже. Предлагаемые цветовые вариации корпуса — синий, сталь, золото/синий и золото/медь.

Nokia 6 продается в США с июля по цене \$229; предполагаемая розничная цена Nokia 8 составит €589 (\$675).



► Флагманская Nokia 8 выполнена в стиле, уже знакомом по другим Android-смартфонам компании.

Источник: [www.theinquirer.net](http://www.theinquirer.net); [www.zdnet.com](http://www.zdnet.com)



## НЕОБЫЧНЫЕ УСТРОЙСТВА

# Образ твой, мучительный и зыбкий...

Компания Red Digital Cinema готовит к выпуску смартфон с поддержкой «голографического контента».

**К**омпания Red Digital Cinema Camera Co., известная своими профессиональными камерами HD, которые использовались для съемок таких фильмов, как *Guardians of the Galaxy*, *The Hobbit*, отечественного сериала «Интерны», объявила о подготовке «первой в мире карманной голографической медиа-машины» — смартфона, способного воспроизводить голографический и 3D-контент.

Поставки новинки, названной Hydrogen One, начнутся в I квартале 2018 г. Устройство получит 5,7-дюймовый «голографический дисплей», будет работать под управлением Android, оснащаться новой высокоскоростной шиной данных и «постоянно расширяющейся системой унифицированных модульных компонентов».

Как утверждает Red, благодаря нанотехнологиям Hydrogen One сможет беспреткновенно переключаться между традиционным 2D-контентом, голографическим multiview-контентом, 3D-контентом

и интерактивными играми, в альбомном и в портретном режимах; при этом специально предназначенные очки для просмотра не потребуются.

В перспективе Red выпустит для Hydrogen One модуль, снимающий в разработанном Red собственном голографическом видеоформате “Hydrogen 4-View content” (h4v). Подобно сервису Apple iCloud, видео и фотографии пользователя будут автоматически сохраняться в облачном сервисе компании Red Channel. Компания также создала собственный алгоритм, преобразующий стереозвук в многомерное объемное звучание.

Hydrogen One будет интегрирован в программу Red camera program, что позволит ему работать с линейкой 8K камер компании Scarlet, EPIC и Weapon в качестве управляющего устройства. В комплект поставки Hydrogen One войдут кабель USB-C, зарядное устройство и слот расширения для карт microSD.



► На сайте производителя ([www.red.com/hydrogen](http://www.red.com/hydrogen)) открыт предзаказ на «первую в мире карманную голографическую медиа-машину» Hydrogen One.

Hydrogen One будет выпускаться разблокированным [unlocked], т. е. не привязанным ни к какому конкретному оператору мобильной связи. Предварительно названы цены \$1195 за модель под названием “Aluminum” и \$1595 — за “Titanium” (различий между версиями компания не уточняет; пока не ясно также, отражает ли название модели тип материала, из которого изготовлен корпус).

## НОВИНКИ SAMSUNG

## Приближая 5G

Новые LTE-модемы Samsung первыми в отрасли получают агрегацию шести несущих.

**К**омпания Samsung Electronics объявила о разработке LTE-модемов нового поколения с агрегацией шести несущих (6CA) для мобильных процессоров; новая технология позволяет достичь скорости загрузки до 1,2 Гбит/с.

Ранее в этом году Samsung представила первый в отрасли мобильный модем с агрегацией пяти несущих (5CA), Cat. 16 LTE для процессоров Samsung Exynos 9 Series (8895), обеспечивающий загрузку данных на скорости до 1 Гбит/с.

Благодаря сотрудничеству Samsung с компанией-производителем контрольного и измерительного оборудования для систем коммуникаций Anritsu (Япония), новые LTE-модемы, основанные на технологии Cat. 18 с агрегацией шести несущих, позволяют увеличить скорость загрузки на 20%, до 1,2 Гбит/с. При такой скорости соединения пользователи мобильных устройств

смогут скачать полнометражный фильм в разрешении HD всего за 10 секунд, общаться видеозвонками без задержки связи, проводить онлайн-трансляции с мобильных устройств.

Технология агрегации несущих (CA) заключается в сочетании определенного числа несущих компонентов с частотными

### Растут скорость передачи данных и производительность сети.

полосами различной ширины; благодаря этому растут скорость передачи данных и производительность сети. По сравнению с предыдущей версией технологии 6CA обеспечивают более быструю и стабильную передачу данных.

В дополнение к сказанному, новая технология поддерживает стандарт 4×4 MIMO (множественный вход, множественный выход) и стандарт 256 QAM (квадратурная амплитудная модуляция).

Кроме того, с целью достижения максимальной скорости передачи данных эта технология поддерживает схему eLAA (Enhanced Licensed Assisted Access); благодаря агрегации частот лицензируемого и нелицензируемого спектров как на нисходящем, так и на восходящем соединении, eLAA позволяет операторам связи полностью задействовать все возможности своего оборудования.

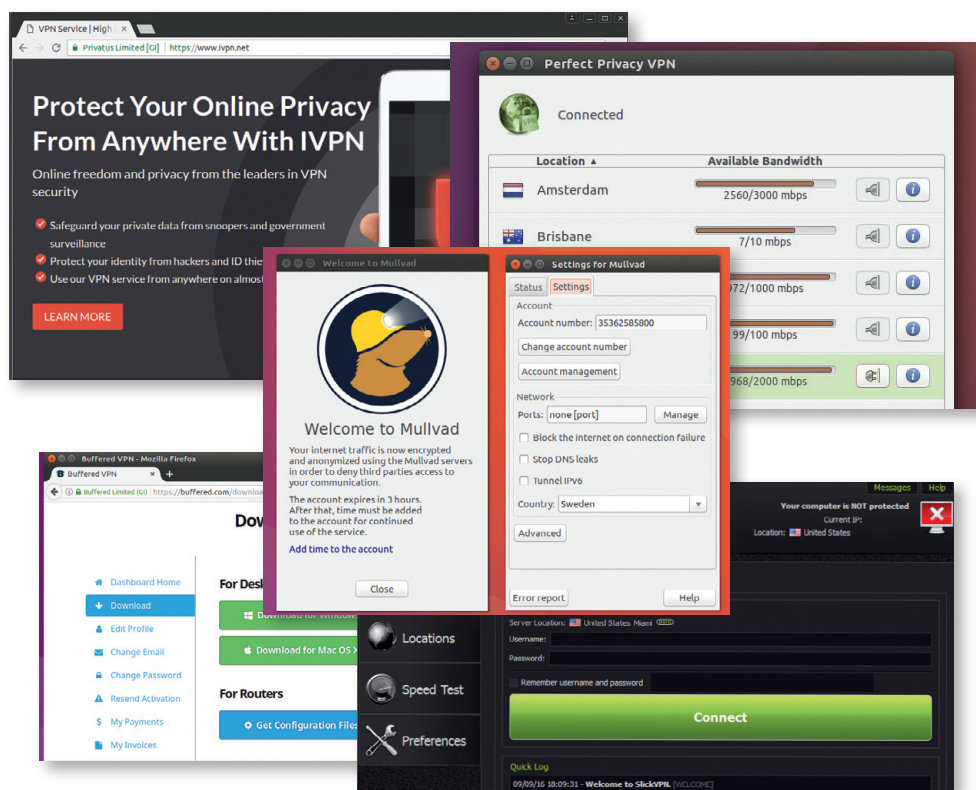
Мобильные процессоры Samsung с LTE-модемами Cat.18, использующими инновационную технологию 6CA, планируется внедрить в массовое производство до конца 2017 г. Об их розничной цене сведений пока не имеется. **LXF**

# Сравнение

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

## VPN-сервисы

Ищете быстрый, многофункциональный VPN, которому можно доверить свою приватность в Сети? Майк Вильямс сравнит пять лучших провайдеров.



### Про наш тест...

Все наши VPN тестировались на одной и той же системе Ubuntu 16.10. Нам было нужно нечто существующее уже некоторое время и имеющее поддержку во всех сервисах. Мы провели сравнительные web-тесты, включая те, что имеются на [speedtest.net](http://speedtest.net) и [testmy.net](http://testmy.net). Тесты мы повторили несколько раз для нескольких серверов, обычно как минимум четырех регионов (Великобритания, Европы, США, Австралии/Азии), затем запустили те же тесты с отключенными VPN, чтобы найти отличия.

Несмотря на то, что некоторые VPN присваивают вам новый IP-адрес, они всё же могут раскрыть некоторые детали вашего настоящего IP-адреса. Мы искали утечки, заходя на [ipleak.net](http://ipleak.net) и [dnsleaktest.com](http://dnsleaktest.com). Кроме того, мы провели немало времени, проверяя основы сервиса каждого приложения. Что собой представляет сайт поддержки, что говорит политика безопасности и есть ли хоть какой-то подвох в малом отпечатке? Мы охватили всё!

**Б**удь то хакеры или вредоносные программы, Google и Microsoft или ЦРУ и АНБ — то и дело всплывает новая история о том, как за всем миром следят и осуществляют мониторинг онлайн. Виртуальные частные сети могут помочь, шифруя web-трафик, защищая его от любопытствующих — среди которых вполне может оказаться и ваш интернет-провайдер — и даже защищая вас в небезопасных точках доступа к Wi-Fi. Они скрывают вашу личность, предоставляя вам новый IP-адрес, и могут дать вам доступ к гео-блокированным сайтам.

Найти нужный VPN бывает непросто, особенно если у вас высокие требования.

Для большинства потребительских VPN превыше всего простота и минимум функций и настроек с базовой поддержкой только самых общих устройств. Мы умышленно избегали таких ориентированных на новичков сервисов, предпочитая компании, способные предложить нечто поинтереснее. Например, у большинства наших провайдеров мощная поддержка Linux, со своими индивидуальными клиентами, подробными

руководствами и предоставлением помощи в случае проблем. Мы также выбирали наших провайдеров на основе тех функций, которые они предлагают, и того, легко ли их можно поднастроить и перенастроить. И, наконец, мы осмотрелись вне технологии — глядя на такие характеристики, как политика конфиденциальности; что помогло нам выделить провайдеров, которым вы вероятнее всего доверите свой трафик.

**То и дело всплывает новая история о том, как следят и осуществляют мониторинг онлайн.**

### Наша подборка

- » Buffered VPN
- » IVPN
- » Mullvad
- » Perfect Privacy
- » SlickVPN

# Функции

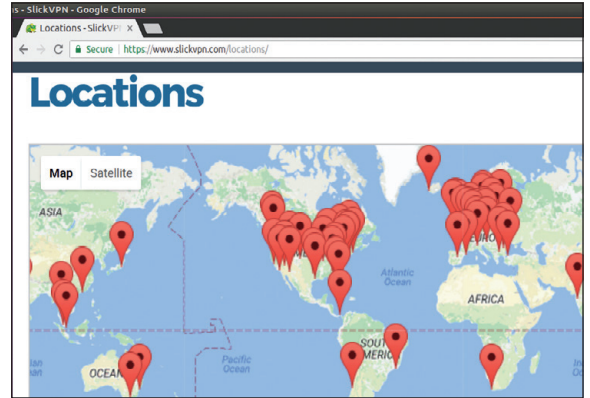
## Что вы получите за свои деньги?

**V**PN кажутся довольно простыми, по крайней мере, в принципе: соединиться с сервером, получить адрес в новом месте и затем работать, как обычно. Велик соблазн предположить, что все они одинаковы и ваш выбор зависит от цены, однако в реальности нужно принимать во внимание много чего еще.

Немаловажно число потенциальных серверов для соединения, а также их географический охват. SlickVPN предлагает местоположение аж в 142 городах по всему миру. Buffered VPN имеет серверы в 37 странах, Mullvad и Perfect Privacy охватывают более 20, а IVPN предлагает выбор всего лишь из 12. Это легкая победа для SlickVPN, широко раскинувшего свои сети, однако подумайте также о требуемом вам местоположении. Если вас устроит несколько мест в Европе и США, вам подойдет любой из этих VPN. Рассмотрите также число клиентов, предлагаемых VPN. Они упрощают установку и заодно могут предложить дополнительные функции. Наличие индивидуальных клиентов означает солидность провайдера VPN — то, что это не просто случайный ухарь, перепродающий чужие

сети из своего домашнего офиса; такое бывает чаще, чем вы думаете. По настольным клиентам вся наша тестовая группа показала результаты выше среднего. Buffered VPN и Mullvad имеют приложения для Linux, Windows и Mac; Perfect Privacy охватывает Linux и Windows; а вот SlickVPN и IVPN предлагают загрузки для Windows и Mac — но не всё потеряно. Вы можете настроить сервис вручную через OpenVPN или использовать любой другой клиент по своему усмотрению, в том числе на устройствах Android и iOS. И каждый VPN предлагает руководства.

Иногда требуется соединиться с VPN через несколько устройств одновременно — скажем, через телефон, планшет и ПК. Mullvad и IVPN разрешают до трех одновременных подключений — вполне достаточно для большинства; зато SlickVPN и Buffered VPN разрешают до пяти, а у Perfect Privacy вообще нет ограничений. Однако помните, что большинство VPN ограничивают свой план одним пользователем, и хотя они допускают несколько подключений, но если все устройства будут заняты круглосуточно, вы, вероятно, получите жалобу.



» Неужели вам и правда нужен узел в Пакистане? Ищите VPN с серверами в ближних к себе странах.

Для некоторых пользователей ключевой функцией является одноранговое соединение (P2P), и у всех наших VPN есть некий уровень поддержки торрентов. IVPN и Perfect Privacy разрешают P2P на выборочных серверах, а SlickVPN, Mullvad и Buffered VPN поддерживают его повсеместно. Если вы более заинтересованы в разблокировке Netflix или похожих сайтов, это намного труднее оценить. Любой VPN может проработать одну неделю и отказать на следующую, поскольку сайт его заблокирует. Единственный эффективный подход — самостоятельно попробовать сервисы, хотя и тогда нет никакой гарантии длительной разблокировки сайта.

### Вердикт

**Buffered**  
★★★★★  
**Mullvad**  
★★★★★  
**Perfect Privacy**  
★★★★★  
**SlickVPN**  
★★★★★  
**IVPN**  
★★★★★

» Полнофункциональный сервис Buffered VPN работает почти везде.

# Политика конфиденциальности

## Можно ли верить обещаниям «без регистрации»?

**V**PN помешает посторонним перехватывать ваш web-трафик, но при этом вы доверяете VPN-провайдеру немалый объем вашей личной информации. Стоит немного покопаться в тексте мелким шрифтом, чтобы выяснить, кому можно ее доверить, а кому нет.

SlickVPN ясно заявляет, что «не отслеживает действий пользователей, зарегистрированных в сервисе SlickVPN». Звучит хорошо, но ведь все утверждают нечто подобное...

Perfect Privacy поступает лучше: его прямодушная политика объясняет, что

компания не записывает ваш логин или информацию по вашему VPN-трафику. Сохраняются только имя пользователя, адрес электронной почты и срок действия. Оплату за сервис позволяется осуществлять в биткойнах.

Mullvad тоже обещает «не отслеживать деятельность», но это не всё. Вы можете зарегистрироваться без указания адреса электронной почты и, опционально, оплатить в биткойнах, чтобы компания никогда не узнала, кто вы.

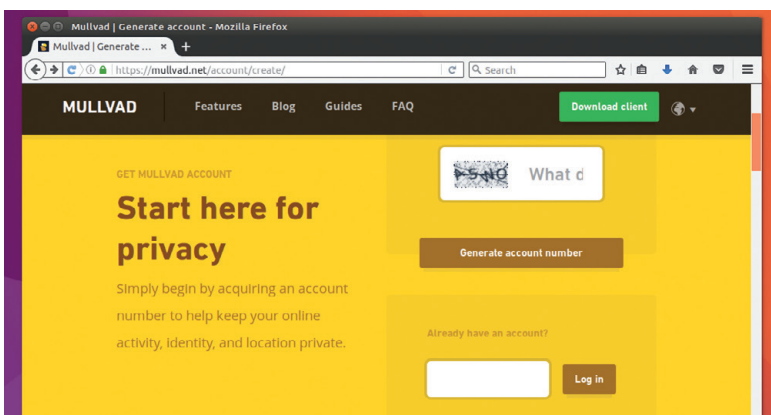
IVPN просит предоставить адрес электронной почты, но не личную информацию. Его политика конфиденциальности тоже легко читаема.

Политика конфиденциальности Buffered VPN столь же легко читается, но толкование более широкое. Компания не записывает информацию о том, что вы делаете онлайн, но может сохранять данные об использовании полосы и прочую информацию по сессии, потенциально способную вас раскрыть.

### Вердикт

**Mullvad**  
★★★★★  
**IVPN**  
★★★★★  
**Perfect Privacy**  
★★★★★  
**Buffered**  
★★★★★  
**SlickVPN**  
★★★★★

» Mullvad вообще не просит никакой личной информации.



» Это полная форма регистрации Mullvad. Анонимнее и быть нельзя.

# Начало работы

Насколько легко настроить и использовать сервис?

**V**PN бывают непростыми в настройке, и обычно провайдеры не очень-то стремятся помочь пользователям Linux. Поэтому мы сознательно выбрали более дружелюбные к Linux сервисы.

Мы тщательно ознакомились с каждым провайдером, оценивая такую информацию, как процесс

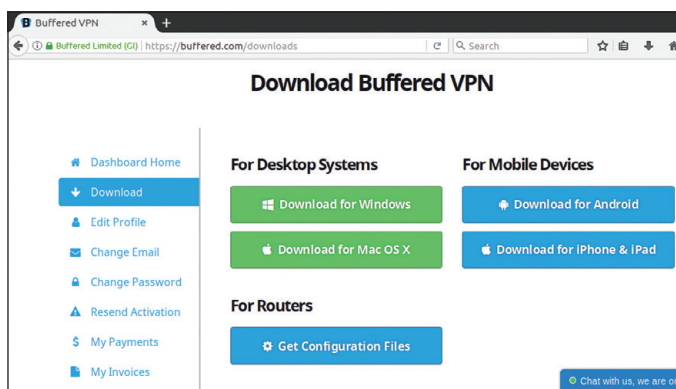
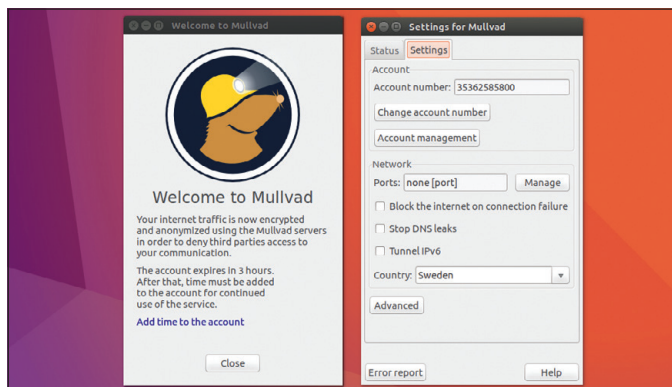
регистрации, установку и все специальные настройки. Мы также рассмотрели все предлагаемые клиенты, их руководства по настройке, и опробовали их в реальных ситуациях, чтобы проверить их удобство в работе.

Те, кто поднаторел в VPN, могут не особенно беспокоиться о проблемах с использованием,

особенно если они подумывают установить нечто стандартное типа OpenVPN. Но даже и в такой ситуации не лишены вероятности шансы, что вам понадобится совет по настройке; поэтому мы дополнительно оценили страницы и руководства «Начало работы» провайдеров на предмет качества и подробностей.

## Mullvad ★★★★★

Mullvad — идеальный сервис для начинающих пользователей VPN. Его настройка сводится к заполнению CAPTCHA на сайте, чтобы получить номер учетной записи, установке и вводу этого номера в клиент Linux — вот и всё. Секунд за 60 мы зарегистрировались, скачали, установили и запустили клиент, и нам выделили новый шведский IP-адрес. У клиента также есть функции для продвинутых пользователей. Он показывает ваш IP-адрес, сервер, порт и протокол заранее. Имеется переадресация (проброс) портов, защита от утечек DNS, поддержка IPv6, полный набор расширенных настроек и весьма подробное онлайн-руководство для всех, кто в нем нуждается. Есть небольшие проблемы с удобством в использовании. Другие клиенты позволяют изменять IP-адрес одним щелчком, а здесь у вас уходит шесть. Производительность тоже немного ниже средней. Но всё же Mullvad — это простой способ начать.



## Buffered VPN ★★★★★

Buffered VPN стоит отметить за создание своего клиента VPN, однако компания не сделала его простым в работе. Его нет в изначальном списке загрузки. Есть клиент Linux в разделе Beta, но без инструкций по настройке; инструкции по старому клиенту Legacy очень краткие. На сайте имеется длинный список руководств по настройке, но они коротки и часто заставляют о многом догадываться самим. Одно из руководств по установке говорит, что вы должны зайти на [buffered.com](https://buffered.com) и скачать «требуемый файл настройки» — ни ссылок, ни имени файла, ни подсказки, где искать: догадывайтесь сами. Опытных пользователей это надолго не задержит; начинающие тоже в итоге осваивают основы Linux, с некоторой помощью от поддержки, и когда система будет настроена, пользоваться ей будет легко. Но она явно могла бы быть намного проще, а вялый темп разработки клиентов намекает, что в ближайшее время вряд ли что-то изменится.

# Цены и оплата

Кто предлагает лучшую сделку и дается ли проба до приобретения?

**С**равнить цены VPN сложнее, чем может показаться. У некоторых сервисов имеются бесплатные планы, другие предлагают ежемесячные выплаты, а иногда можно сэкономить, подписавшись на более долгий срок. Если вы ищете простой ежегодный план, SlickVPN предлагает вам наилучшие условия при оплате £3,20 (\$4) в месяц. Buffered VPN стоит вдвое дороже — £6,60 (\$8,25), IVPN почти столько же — £6,66 (\$8,33), а Perfect Privacy, самый дорогой из всех — £8,84 (\$11,05) в месяц при годовой подписке.

Если вам нужен VPN лишь изредка, есть смысл платить ежемесячно. Mullvad выигрывает, предлагая всего £4,24 (\$5,30); плата за один месяц SlickVPN составляет £8 (\$10), Buffered VPN — £10,40 (\$13), Perfect Privacy — £11,46 (\$16,86), а IVPN стоит £12 (\$15) за месяц. Удобно платить в биткойнах, поскольку это позволяет не сообщать провайдеру личной информации; SlickVPN, IVPN, Perfect Privacy и Mullvad принимают биткойны, но Buffered VPN — нет.

Бесплатные пробные периоды всегда нас радуют, однако в нашей группе таковой

предлагает только Mullvad. Период ужасно короткий — всего три часа; но этого достаточно, чтобы подтвердить возможность подключения.

У других сервисов нет пробного периода, зато они гарантируют вам возврат средств. SlickVPN и Buffered VPN — самые щедрые, они предлагают 30-дневный гарантированный возврат средств без всякого риска; следующий за ними — IVPN, с семидневным периодом; у Perfect Privacy тоже семидневный период, но с большим количеством условий, чем у других.

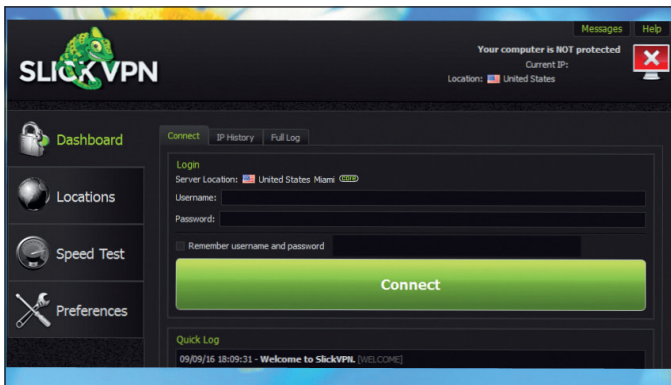
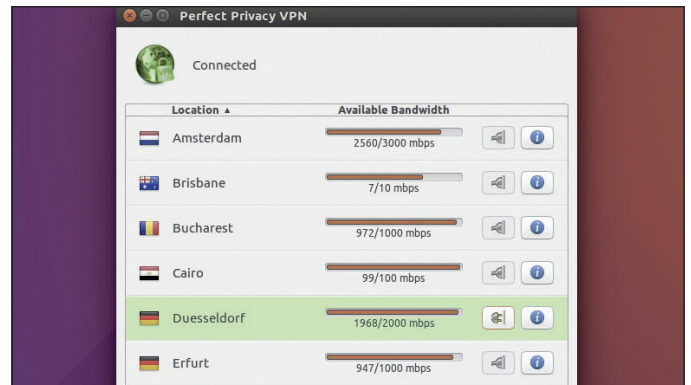
## Вердикт

- Mullvad ★★★★★
- SlickVPN ★★★★★
- Buffered ★★★★★
- IVPN ★★★★★
- Perfect Privacy ★★★★★

» Помесячная оплата может быть разумным вариантом, но он не единственный.

## Perfect Privacy ★★★★★

Отлично разработанный сайт Perfect Privacy прекрасно справляется с помощью в начале работы. Попробуйте его: перейдите на [www.perfect-privacy.com/howto](http://www.perfect-privacy.com/howto), и он определит вашу ОС, предложит вам ссылку для скачивания нужного ПО, покажет пошаговое руководство, подсказки по настройке и много чего еще. Нам практически не понадобилась вся эта поддержка, поскольку клиент скачался, установился и настроился на нашем тестовом компьютере за считанные мгновения. Клиенты Perfect Privacy отображают большое количество информации, включая страны и города, где расположены ваши серверы, и их текущую нагрузку. Интерфейс не столь интуитивен, как у некоторых его конкурентов, но как только вы поймете основы работы, это не будет проблемой. Решив пойти дальше, вы оцените множество настроек, которые обеспечивают невероятный уровень управляемости сервиса.



## SlickVPN ★★★★★

Отсутствие в SlickVPN клиента Linux означают необходимость задуматься о настройке. Сайт всё же предлагает несколько руководств по установке, охватывая OpenVPN на *NetworkManager*, командную строку и настройку PPTP. Руководства подробнее, чем в Buffered VPN, хотя и не столь просты. Клиент Windows, который можно запустить через *Wine*, работает хорошо. Он и выглядит хорошо, отображает серверы в списке или на карте, проводит тесты на скорость, чтобы найти самые быстрые серверы, и имеет ряд удобных функций и настроек как бонус для более опытных пользователей. Подключитесь прямо в Linux, и у вас будет доступ к обычным настройкам OpenVPN через *NetworkManager* или к любому другому OpenVPN-совместимому клиенту, который они захотят установить. У новичков в сети будет не так много опций, поскольку сайт SlickVPN не предлагает особой помощи сверх начальной установки.

## IVPN ★★★★★

IVPN, как и SlickVPN, не имеет клиента Linux, но компания очень старается помочь вам через руководства по настройке с помощью *NetworkManager* или терминала, на диво подробные, и они далеко выходят за рамки списка команд «введите это, введите то», часто встречающегося в других местах. Мы попробовали клиент Windows, потому что вы, вероятно, будете использовать свой VPN дома. Он предлагает что-то каждому, с подключением наведи-и-щелкни, если вам это надо, но также и с более расширенными настройками по защите конфиденциальности для более опытных пользователей. IVPN во время тестирования показал себя надежным, и у нас вообще не было проблем с удобством в использовании. Но если вам повезет меньше, компания предлагает массу подробных руководств по решению проблем, вам в помощь. Единственная мелкая загвоздка — поддержка Linux менее объемна, чем утверждает сайт.



# Поддержка и документация

Уф, ох... похоже, вы подключились. Что теперь?

**Н**астройте VPN, и он, как правило, будет работать, а вам не надо будет его подправлять. Но, как обычно происходит в сети, если уж проблемы возникли, то они сложные, а значит, важно, чтоб ваш VPN имел систему поддержки. У Buffered VPN большое количество статей в Сети, включая статьи по настройке системы для Linux, но им явно не хватает подробности. Если у вас возникли проблемы, с техподдержкой можно связаться круглосуточно по электронной почте, и иногда доступен чат live. База сетевых знаний

Mullvad тоже предлагает очень краткие статьи, но охватывает куда больше областей. Так, вместо общего параграфа по установке Linux вы получаете детальные сведения по разным дистрибутивам. Есть руководства по темам для специалистов, и имеется поддержка по электронной почте. SlickVPN предлагает хорошо представленный ряд руководств по настройке для многих ОС и роутеров. Советы по решению проблем только базовые, а помощь доступна только через общий адрес электронной почты, причем ответ дается «в течение рабочего

дня». У Perfect Privacy самый дружелюбный к новичкам подход, со скриншотами по каждому шагу установки и руководствами по настройке. Компания также предлагает поддержку по электронной почте, через форум и даже *TeamViewer*. IVPN дает даже больше информации, чем Perfect Privacy, но она — для опытных пользователей и предполагает наличие достаточно обширных знаний. Тем, кто знает, что делает, есть с чем ознакомиться, а если нет — свяжитесь с техподдержкой по электронной почте, и ответ должны прибыть в течение дня.

## Вердикт

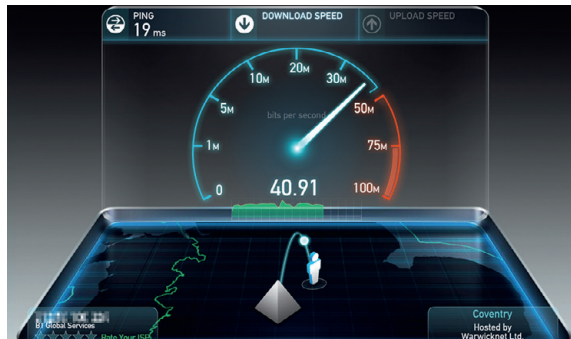
IVPN  
★★★★★  
Perfect Privacy  
★★★★★  
Mullvad  
★★★★★  
SlickVPN  
★★★★★  
Buffered  
★★★★★

» До регистрации загляните на страницу техподдержки VPN; она может вам пригодиться.

# Производительность

Как поведет себя ваш VPN в тестах реальной жизни?

**З**агляните на сайт любого провайдера VPN, и обычно там найдется масса громких утверждений и рекламы. Однако в реальности бывает совсем иначе, так что стоит проверить их лично, на своих тестах. Скорость Mullvad была ниже средней. Сайты сравнительных тестов, например, [speedtest.net](http://speedtest.net), в большинстве мест дали нам как минимум 10 Мб/с, что очень неплохо для базового просмотра и даже потокового вещания HD, но P2P и массивные скачивания могут занять продолжительное время. SlickVPN в наших тестах получил больше очков, особенно на близкие расстояния (из Британии в Британию), где скорость была почти нормальной. На больших расстояниях производительность значительно снижалась и сильно различалась на разных серверах, однако минимальная скорость составляла более 15 Мб/с. Buffered VPN был намного лучше. Небольшие расстояния до ближайшего сервера в Британии дали увеличение скорости вдвое по сравнению с Mullvad, скорость из Великобритании в США также впечатляла, и малые



Мы проверяли скорость скачивания для всех основных серверов.

периоды ожидания способствовали работе в Сети без задержек. Как и следовало ожидать, на более удаленных серверах производительность падала. IVPN в большинстве ситуаций превзошел Buffered по производительности, хотя разница была незначительна. IVPN оказался крепким всесторонним провайдером и выдал хорошие результаты на всех серверах. Большинство наших участников развили неплохую скорость, но Perfect Privacy стал настоящим королем производительности. Локальные

подключения дали около 40 Мб/с, и даже дальние, из Британии в Австралию достигли минимум 15 Мб/с. Наши тесты VPN не ограничились скоростью. Мы не поленились проверить каждый сервис с помощью [ipleak.net](http://ipleak.net) и прочих сайтов, пытаемся найти утечки конфиденциальности, способные раскрыть вашу подлинную личность, и копали весьма глубоко, но наши тестируемые VPN показали хорошие результаты, блокируя DNS-, WebRTC- и прочие утечки и защищая нашу конфиденциальность постоянно.

**Вердикт**

Perfect Privacy  
★★★★★

Buffered  
★★★★★

IVPN  
★★★★★

SlickVPN  
★★★★★

Mullvad  
★★★☆☆

» Даже самым медленным VPN хватает скорости для базовой работы в Сети.

# Расширенные функции

Нужен ли вам VPN, выходящий за рамки основ?

**П**одписка на любой VPN дает вам одни и те же основные функции: возможность безопасно перенаправлять свой интернет-трафик через один из серверов провайдера. В большинстве ситуаций этого должно быть и достаточно, но некоторые компании идут дальше, добавляя еще функции для защиты вашей конфиденциальности и обеспечения безопасности онлайн.

Обычные VPN направляют ваш трафик через любой выбранный вами сервер, например. Это дает высокую степень защиты, но если кто-то увидит, когда именно вы зашли на VPN и какой сервер был вам выделен (некоторые провайдеры хранят такую

информацию о сессии), то есть шансы, что этот кто-то сумеет связать ваш трафик с какими-то вашими действиями онлайн. Некоторые провайдеры предлагают дополнительный уровень безопасности, направляя трафик через несколько серверов.

У IVPN имеется функция “multihop [перескоки]”, когда вы можете подключиться, например, к серверу в США, а затем выйти через сервер в Амстердаме, что сильно затруднит отслеживание вашей деятельности в Сети. В Perfect Privacy есть похожая функция, под названием “cascading [каскадирование]”, а в SlickVPN — Hydra tech направит ваш трафик через случайно выбранные серверы.

Клиент Buffered VPN не столь умелый. В нем есть интересные и необычные функции, но ничего сногшибательного с технической точки зрения.

Mullvad тоже кажется весьма заурядным сервисом, по крайней мере — сперва; но не дайте себя провести — впереди вас ждет много интересного. Mullvad умеет работать с туннелированием IPv6, блокировками и защитой от утечек, и даже с IPv6 через IPv4. Просмотр спецификаций Mullvad показывает, что там еще много чего хорошего, в том числе черновая поддержка WireGuard, нового протокола VPN, который, вероятно, встроит в ядро Linux в 2017 г.: это явный признак провайдера, который знает, что делает. Mullvad, вероятно, станет нашим выбором для экспертов, но Perfect Privacy не очень сильно от него отстает. Среди расширенных функций — настраиваемое аварийное отключение, переадресация портов, защита от утечек и интересные скрытые режимы. Есть также и функции для новичков, в том числе опциональный модуль TaskStop, который блокирует доступ к отслеживанию IP-адресов.



SlickVPN Hydra направляет весь сетевой трафик через разные серверы VPN.

**Вердикт**

Mullvad  
★★★★★

Perfect Privacy  
★★★★★

IVPN  
★★★★★

SlickVPN  
★★★★★

Buffered  
★★★☆☆

» Направление вашего трафика через несколько серверов дает больше конфиденциальности.

И лучшим сервисом VPN становится...

# Вердикт

**П**ровайдеры VPN часто с виду очень похожи, и велик соблазн выбрать сервис, основываясь на простейших сведениях, например, о цене и количестве местоположений. Но это не лучшая стратегия. Достойных учета факторов куда больше, и даже самые очевидные цифры трудно интерпретировать. Например, избытие местоположений не сыграет для вас большой роли, если вам незначим их использовать, а низкая цена может оказаться менее привлекательной, если за нее вы получите медленные серверы без всякой поддержки. Мы постарались пореалистичнее взглянуть на наших провайдеров, подвергнув их более глубокой проверке: читая условия, напечатанные мелким шрифтом, регистрируясь, устанавливая сервисы и проверяя их в реальном мире.

Buffered VPN, может быть, и оказался в низу нашего списка, но это неплохой сервис. Вы получаете серверы в 37 местах, что выше среднего показателя; все они с поддержкой P2P, и скорость тоже лучше, чем

у многих конкурентов. Поддержка на сайте не самая лучшая, и политика конфиденциальности могла бы быть более внятной, но если у вас есть опыт, вы получите всё, что вам нужно. SlickVPN тоже показывает хорошие результаты в некоторых ключевых областях, предлагая больше местоположений (140+), чем остальные, при самой низкой цене в нашей группе. Нам показалось, что он не преуспел в других тестовых областях, но если ваши приоритеты — цена и количество местоположений, может стать хорошим выбором. IVPN стоит почти вдвое дороже, чем SlickVPN, но нам показалось, что его производительность в реальном мире выводит сервис вперед. Он был быстрее, надежнее и логичнее, и на его сайте поддержки было чуть больше подробностей на случай проблем. У него четкая политика конфиденциальности, и нам понравилась возможность регистрации с указанием только адреса электронной почты. Mullvad выводит анонимность на новый уровень, позволяя вам создать учетную



запись и тестировать ее в течение трех часов, вообще не указав личной или финансовой информации. Низкая цена за месяц и сильные функции OpenVPN тоже являются важными плюсами, однако несколько разочаровывающая производительность отбрасывает его назад.

Perfect Privacy здесь самый дорогостоящий сервис, однако превосходные результаты во всех остальных областях выводят его в победители. Он самый простой в настройке, у него есть клиент Linux, он обеспечил самую высокую скорость в наших тестах и предлагает хорошую смесь функций для начинающих (блокировка отслеживания) с настройками экспертного уровня.

► Победитель может быть только один, но борьба была серьезной.

**I Perfect Privacy** ★★★★★  
 Сайт: [www.perfect-privacy.com](http://www.perfect-privacy.com) Цена: £ 8,84 в месяц (оплата за год)  
 » Быстрый VPN, который сумел позаботиться обо всех аспектах сервиса.

**IV SlickVPN** ★★★★★★  
 Сайт: [www.slickvpn.com](http://www.slickvpn.com) Цена: £ 3,20 в месяц (оплата за год)  
 » Вы получаете размещение в 142 городах при низкой годичной плате.

**II Mullvad** ★★★★★★  
 Сайт: [www.mullvad.net](http://www.mullvad.net) Цена: £ 4,24 в месяц  
 » Мгновенная регистрация, не требуется никаких личных данных.

**V Buffered VPN** ★★★★★★  
 Сайт: [buffered.com](http://buffered.com) Цена: £ 6,60 в месяц (оплата за год)  
 » Предлагает несложный клиент Linux, и скорость у него выше средней.

**III IVPN** ★★★★★★  
 Сайт: [www.ivpn.net](http://www.ivpn.net) Цена: £ 6,66 в месяц (оплата за год)  
 » Весьма надежный, стабильный и простой в использовании сервис.

**Обратная связь**  
 А каков ваш опыт с VPN, о которых мы рассказали? Может, мы упустили яркого претендента? Сообщите нам на [lxf.letters@futurenet.com](mailto:lxf.letters@futurenet.com).

## Рассмотрите также...

**М**ы старались выбрать VPN, предлагающие чуть больше, чем обычные сервисы, но если вас устраивает базовая конфиденциальность, то популярных альтернатив немало. IPVanish ([www.ipvanish.com](http://www.ipvanish.com)) заявляет, что он «быстрейший в мире VPN» с 500+ серверами на 60+ серверах. Он имеет клиенты для Windows,

Mac и мобильные; но пользователям Linux придется настраивать систему вручную. VPN Unlimited ([www.vpnunlimitedapp.com](http://www.vpnunlimitedapp.com)) предлагает клиент Linux и недорогие понедельные планы. У PureVPN ([www.purevpn.com](http://www.purevpn.com)) тоже есть клиент Linux — и сервера в 140+ странах, а NordVPN ([www.nordvpn.com](http://www.nordvpn.com)) умеет отправлять трафик в Tor для обеспечения

лишнего уровня безопасности. Не стоит презирать бесплатные VPN. У многих сильно ограничена полоса пропускания — 500 МБ в месяц, а то и меньше — а Windscribe щедро предлагает 10 ГБ. Мест всего 8, в основном в Британии, США и Европе, и скорость средняя, но он лучше иных коммерческих продуктов, а среди бесплатных — один из лучших. **LXF**

# Уход от Windows

Если вас ослепляет сияние Linux OS,  
Ник Пирс предложит вам ответы  
на все наиболее общие вопросы о Linux!

Ставим  
Linux!  
См. стр. 46

**К**аким бы знакомым ни было всё в Windows, вы не можете перенести всё, что знаете, и начать использовать новую ОС точно так же, как раньше, и это особенно актуально, когда речь идет о Linux, даже в его нынешнем виде, дружелюбно к пользователю.

Конечно, кое-что перенести можно, но всё равно имеются фундаментальные различия в работе Linux, которые следует понимать.

В нашем руководстве мы возьмем за основу Ubuntu 16.04 LTS ([www.ubuntu.com/download](http://www.ubuntu.com/download)), на котором базируется множество ОС Linux, включая те, которые вы найдете на LXF DVD, и вы увидите, что одна из главных прелестей мира Linux — это богатство выбора.

Мы начнем с изучения файловой системы Linux (а файлом является всё, даже ваши папки

и аппаратные устройства), и рассмотрим, как она работает с пользователями и разрешениями. Мы вскользь затронем драйверы (в Linux вам нечасто придется о них беспокоиться), и затем перейдем к поиску, установке, обновлению и обще-

## Вы получите необходимую базу для перехода с Windows на Linux.

му управлению программами, поскольку всё это несколько отличается.

Мы также познакомим вас с *Терминалом*, который предоставляет доступ к оболочке, лежащей в основе всей работы Linux. Мы объясним основы использования *Терминала* и перечислим 10 практических способов его применения, от копирования

файлов и папок до получения информации об аппаратной части вашего ПК.

Как только вы освоитесь, мы пройдем немного дальше и объясним, что происходит при запуске компьютера с Linux. Понять, что и почему происходит в Linux, крайне важно. В Windows процесс загрузки спрятан за непрозрачным барьером, а в Linux вы можете изучить его и настроить с огромной точностью.

Преподав всю эту теорию, мы проведем вас по процессу установки дистрибутива Linux неразрушающим способом; даже если вы не вполне готовы покончить с Windows, вы сможете запускать их параллельно. Вы получите необходимую базу для перехода с Windows на Linux. Не будем же терять времени впустую — как-никак, нас уже заждались целая операционная система!



# Навигация по файловой системе

Не бойтесь управления файлами в Linux — по сути, оно весьма логично.

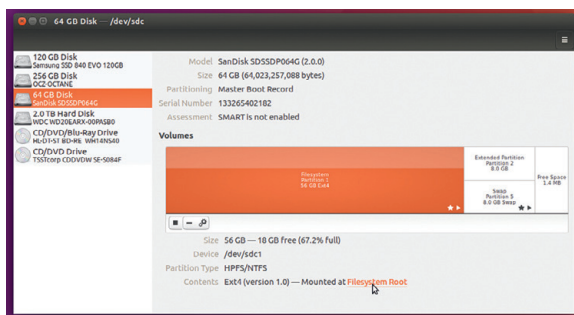
**Ф**айловая система Linux (ext2, ext3 и ext4) отличается от Windows. Главное отличие в том, что диски и разделы рассматриваются не как определенные буквы дисков, а как файлы внутри основной файловой системы.

Откройте Dash и введите 'disks', чтобы открыть утилиту *Disks*, которая поможет вам визуализировать, как всё работает. Выберите диск в левой части меню и загляните в меню *Disks*. Вы увидите ссылку на свой диск, которая выглядит примерно так: `/dev/sda`. Это ссылка на папку устройств `dev` (devices), внутри которой каждому физическому диску соответствует уникальный файл, имя которого начинается с `sd` (т.е. устройство хранения — storage device), за чем следует уникальная буква от `a` до `z`. Логическая схема здесь такова: `/dev/sda` обозначает диск, где установлена ОС (у нас — Ubuntu), а за ним следуют диски `sdb`, `sdc` и далее, в порядке следования сперва внутренних дисков, а затем внешних носителей. Данная логика касается и разделов диска, т.е. вместо присвоения разделам (или томам) отдельных букв дисков, Linux идентифицирует каждый из них посредством отдельного файла, давая им имена `sda1`, `sda2` и т.д. Разделы представлены гистограммой в *Disks*.

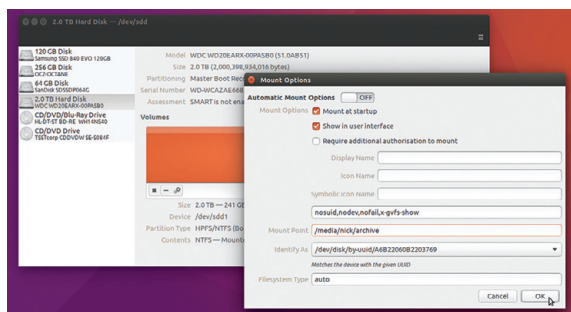
Нумерация не всегда логична. Если ваш диск форматировался, например, через схему разбиения на разделы MBR (Master Boot Record), вы увидите, что разделу подкачки [swap] присвоено `sda5`, а не `sda2`. Не беспокойтесь: всё будет работать, как надо.

## Оценка дисков

Диски не просто встроены в основную файловую систему как отдельные файлы — они также смонтированы в специальные папки тоже внутри файловой системы, где и можно найти их содержимое.



Утилита *Disks* визуализирует способ, которым Linux определяет ваши диски и разделы.



С помощью утилиты *Disks* настройте тома на автоматическое монтирование при запуске Ubuntu.

Ваш главный системный диск (обычно `/dev/sda1`) монтируется в корневую директорию (`/`) файловой системы. Другие внутренние и внешние тома не монтируются по умолчанию, а раздел подкачки не монтируется вообще. Простейший способ доступа к файлам и папкам — через менеджер файлов. Соответствующая программа Ubuntu по умолчанию, *Files*, основана на инструменте *Nautilus* и похожа на *File Explorer* [Проводник] в Windows. Откройте ее, нажав на значок с картой в Dash, и увидите в левой панели навигации список имеющихся томов. Нажмите на требуемый том, чтобы его смонтировать, затем на *Computer* в панели навигации, для перехода в корневую директорию, и просмотрите `/media/<yourname>` для поиска папок, указывающих на каждый физически смонтированный диск. Можно смонтировать основные папки на отдельных томах — так, многие варианты Linux создают в процессе установки отдельный раздел для папки `/home`, где хранятся все настройки вашего пользователя и данные, включая такие папки, как *Documents* (но Ubuntu, один из немногих, этого не делает).

## Монтирование при запуске

Пожелав, чтобы Ubuntu при запуске монтировал заданные разделы, откройте утилиту *Disks* (а также *Parted* или *Gparted*). Выберите свой диск слева, затем выделите том, выбрав его на гистограмме. Нажмите на кнопку *Settings* и выберите *Edit Mount Options* [Редактировать Опции Монтирования]. Переключите переключатель *Automatic Mount Options* [Автоматические Опции Монтирования] в *On* и обязательно отметьте две первых опции, затем измените поле *Mount Point* [Точка Монтирования] на `/media/<user>/<label>`, заменив `user` на свое имя пользователя, а `label` — на метку тома. Нажмите *OK* и перезагрузитесь, чтобы проверить, монтируются ли диски автоматически.

»

## Экстренное сообщение: Linux — не Windows!

Рабочие столы большинства дистрибутивов Linux умело скрывают тот факт, что Linux и Windows — разные операционные системы. Конечно, есть такая мелочь, как файловые системы, но Linux намного жестче в плане безопасности, чем Windows. В некоторых областях Windows ликвидировал разрыв — например, по части разрешений на файлы и папки; но в остальном Linux оставляет его далеко позади. Например, в Linux нельзя выполнять административные задачи — скажем, установить программу — не предоставив свой пароль пользователя.

Именно это внимание к деталям и делает Linux гораздо менее уязвимым для вредоносных программ и хакеров.

Несмотря на графическую внешность, Linux остается операционной системой, которая лучше всего работает из оболочки. У Ubuntu хороший интерфейс, но, как вы увидите, не стоит сопротивляться использованию командной строки из Терминала.

Ubuntu является более открытой, поддерживаемой сообществом средой, чем Windows. В нем имеются бесплатные (и с открытым кодом) программы,

которые позволяют снизить стоимость компьютерных работ, но большая часть поддержки находится онлайн, на форумах. То, как Linux работает с программами — через централизованные репозитории и менеджеры пакетов — еще одно, к чему вам придется привыкнуть.

И, наконец, совместимость оборудования уже не является такой проблемой, как раньше. Хотя некоторые устройства не работают, но большинство работает, и многим из них не требуется дополнительной настройки.

# О правах на файлы

Пользователи, суперпользователи и права на файлы... узнайте, как Linux повышает безопасность и контролирует доступ к файлам и папкам.

**Л**юбая стоящая файловая система применяет к файлам ограничения в виде разрешений, ограничения доступа по пользователям и группам. Windows отчасти делает это в своей файловой системе NTFS, но это не заменяет подход Linux.

В Linux всё представлено в виде файла, включая папки и аппаратные устройства. И затем файловая система ext применяет к этим файлам специальные разрешения, чтобы определить, как получить к ним доступ и кому дозволено его получить. Эти ограничения сводятся к трем основным уровням доступа: r (read), w (write) и x (execute). Вы можете просмотреть разрешения файлов из Терминала командой `ls -l`: в ее выводе рядом с каждым файлом вы увидите записи вроде, например, `rwX` (полный доступ) или `r--` (только чтение). Разрешения `rwX` применяются также и к папкам, и всё несколько осложняется тем фактом, что определенные разрешения — например, на удаление файла — размещаются в родительской папке, а не в самом файле. Поэтому файлы можно просматривать (r), редактировать (w) и исполнять (x), если это программа, но если родительская папка не имеет разрешений w, то вы не можете создавать (или удалять) файлы в этой папке. Подобным же образом, нельзя просматривать файлы — даже с разрешением `g` — если у папки нет разрешений x.

Безопасность для Linux исключительно важна, поэтому разрешения не применяются напрямую к каждому файлу и папке. Вместо этого они применяются к трем категориям пользователей: владельцу (пользователю, который создал файл), заданной группе пользователей и прочим (всем остальным).

Вторая категория относится к одной пользовательской группе, которой были выданы определенные права, и открывает вопрос о пользователях и группах. Хотя кажется, что они применимы к определенному пользователю (часто к тому, который является владельцем папки), эти разрешения применяются к группе пользователей. Когда вы создаете нового пользователя, заодно создается группа с тем же именем, и ваш пользователь добавляется к этой группе; на нее-то и ссылается Linux. Можно также добавлять пользователей к нескольким группам, позволяя опытным пользователям создавать группы и добавлять к ним пользователей, и эти пользователи получают уровень доступа к выбранному файлу или папке на основании прав группы, к которой они относятся.

## Задание разрешений

Разрешения/права задаются при создании файла, и владелец файла также считается пользователем или группой по умолчанию для этого файла. Обратите внимание, что если вы создаете файл в режиме с повышенными правами (например, через `sudo` в Терминале), то владельцем будет суперпользователь-root, а не вы. У владельца обычно имеется право полного доступа к созданному файлу или папке, тогда как все остальные обычно имеют более ограниченные права относительно файлов, причем их права относительно папок заблокированы. По умолчанию все пользователи имеют полную собственность, доступ и контроль за своей персональной папкой **Home** и ее содержимым, а другим пользователям доступ к ней заблокирован. Вне домашней папки доступ более ограничен — определенные папки доступны, но большинство — или только для чтения, или доступ к ним полностью запрещен, то есть может быть получен только через учетную запись `root`. Если вы — владелец файла или папки, то можете изменить разрешения через менеджер файлов *Nautilus*. Щелкните правой кнопкой по папке или файлу и выберите вкладку **Properties > Permissions** [Свойства > Разрешения]. Отсюда вы можете изменять права владельцев, групп и прочих и вносить изменения в то, какая группа пользователей имеет специальный доступ к данному объекту. Нажмите на выпадающее меню **Group** [Группа], и в списке появится куча незнакомых имен — это системные пользователи, созданные для выполнения определенных задач без нарушения безопасности, и их лучше не трогать.

➤ Используйте `ls -l` в Терминале, чтобы просмотреть разрешения для файлов и папок в текущей директории.

```
nick@nick-pc:~$ ls -l
total 31500
drwxrwxr-x 3 nick nick 4096 Feb 28 08:42 apps
-rw-rw-r-- 1 nick nick 905 Dec 29 16:26 dan-retroshare
drwxr-xr-x 2 nick nick 4096 Feb 28 08:42 Desktop
drwxr-xr-x 5 nick nick 4096 Sep 15 17:42 Documents
drwxr-xr-x 4 nick nick 4096 Feb 23 16:39 Downloads
-rw-rw-r-- 1 nick nick 905 Dec 30 14:41 edc
-rw-r--r-- 1 nick nick 8980 Mar 14 2016 examples.desktop
drwxr-xr-x 3 nick nick 4096 Mar 15 2016 Music
-rw-rw-r-- 1 nick nick 897 Dec 30 14:43 nickp
drwxr-xr-x 2 nick nick 4096 Oct 11 15:56 Pictures
drwxr-xr-x 2 nick nick 4096 Mar 14 2016 Public
drwxrwxr-x 6 nick nick 4096 Jun 17 2016 quicklisp
drwxrwxr-x 4 nick nick 4096 Jan 3 09:44 Recovered
```

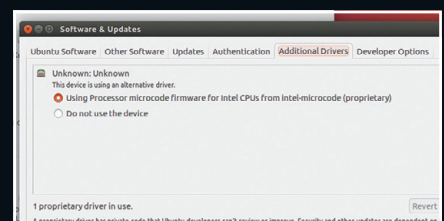
## Драйверы стали проще

Одна из множества сильных сторон Linux заключается в том, что к оборудованию применяется подход «это просто работает». Когда вы устанавливаете ОС Linux, вам не приходится судорожно отыскивать ключевые драйверы. Всё должно работать сразу.

Ubuntu отлично поладит с вашей видеокартой или чипсетом, хотя если вы планируете использовать свой ПК для игр, вам, возможно, придется прибегнуть к сторонним альтернативам. Пользователи NVIDIA и AMD должны открыть **System Settings** [Настройки системы] и перейти в **Software & Updates > Additional Drivers** [Программы и обновления > Дополнительные драйверы]. Немного погодя выберите самую свежую версию, отмеченную как «проприетарная, проверенная [proprietary, tested]» и нажмите

**Apply** [Применить]; подождите и после завершения перезагрузитесь.

Если вы планируете добавить принтер, то перейдите в **System Settings > Printers** [Настройки системы > Принтеры] и нажмите **Add** [Добавить], чтобы посмотреть, не определится ли он автоматически; затем следуйте инструкциям мастера, чтобы узнать, включены ли драйверы. Если нет, загляните на сайт производителя, чтобы выяснить, есть ли драйвер для Linux. Если это Deb-пакет (.deb), то установка проста; если нет, вам, возможно, придется поискать программу установки .sh — проверьте, чтобы файл был исполняемым (отметьте **Allow executing file as a program** [Разрешить исполнение файла в качестве программы]) во вкладке **Permissions** [Разрешения]



➤ Узнайте, какие графические драйверы использует Ubuntu, через инструмент **Software & Updates**.

свойств файла в *Nautilus*), затем откройте Терминал, перейдите в директорию файла и запустите так:

```
$. /filename.sh
```

# Установка программ

Есть масса способов установки программ в Linux, как в *Терминале*, так и вне его. Какой способ больше подойдет вам?

**М**ногие программы Linux хранятся в так называемых репозиториях. Это онлайн-каналы, которые объединяют программы одного типа, созданные для определенных версий этого дистрибутива, например, Ubuntu (поэтому репозитории Ubuntu 16.04 отличаются от репозитория Ubuntu 14.04 или 16.10). Если говорить об Ubuntu, то здесь есть четыре основных канала для каждой отдельной версии: **Main** [Основной], **Restricted** [Ограниченный], **Universe** [Универсальный] и **Multiverse**.

Репозиторий **Main** содержит программы с открытым кодом, которые можно перераспространять и которые поддерживаются и постоянно обновляются командой Ubuntu. **Universe** содержит свободные программы и программы с открытым кодом, обновления для которых предоставляет сообщество. **Restricted** размещает проприетарные (с закрытым кодом) инструменты и драйверы, необходимые для поддержки Ubuntu на обычном оборудовании, а **Multiverse** содержит программы, которые не являются ни свободными, ни поддерживаемыми.

Доступ к этим репозиториям можно получить через менеджеры пакетов: например, *Software Centre* дает вам удобный централизованный пункт для установки и автоматического обновления пакетов.

## Вне репозитория

Большую часть нужных вам программ можно найти в этих четырех основных репозиториях. Если «кругозор» *Software Centre* покажется вам несколько ограниченным, попробуйте более универсальный (и немного более расширенный) менеджер пакетов, введя в поиске «Synaptic», чтобы установить *Synaptic Package Manager*. *Synaptic* обеспечит более тщательный поиск по репозиториям, помогая вам найти нужные пакеты, и дает более всесторонний (возможно, даже слишком) список программ по умолчанию.

Не все программы доступны через репозитории по умолчанию — иногда они хранятся в своих собственных репозиториях, которые вы можете добавить в свой менеджер пакетов одним из двух способов: или через **Settings > Software & Updates > Other Software** [Настройки > Программы и Обновления > Другие Программы], или через *Терминал* (о нем речь пойдет далее). После установки система будет сообщать вам обо всех обновлениях, выпущенных производителями программ.



» *Ubuntu Software Centre* предоставляет универсальное средство для установки и управления программами.

Другие программы можно скачать отдельно в виде пакетов, которые работают так же, как программы установки в Windows. В них есть всё, что нужно программе для успешной работы — не только сами программы, но и ссылки на зависимости, которые вам предложат установить, если их еще нет в вашей системе.

Эти файлы часто имеют расширение **.deb**. Сохраните его в вашу папку **Downloads**, затем дважды щелкните по файлу — в этот момент всю работу должен взять на себя ваш менеджер пакетов. Обратите внимание, что хотя ваш менеджер пакетов регистрирует установку, он не сможет определить наличие обновлений — это зависит от программы или ее разработчика.

## Установка из Терминала

Познакомьтесь с огромным выбором инструментов *apt*, и вы увидите, что по большей части наилучшим способом установки программных пакетов является *Терминал*. Начните с команды `$ sudo apt-get update`. Она находит самые свежие списки пакетов (включая обновленные версии) из всех установленных репозиториях.

Следующие две команды устанавливают и удаляют программы:

```
$ sudo apt-get install <package>
```

```
$ sudo apt-get remove <package>
```

А эта команда обновляет все установленные программы:

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

Через *Терминал* также можно добавлять репозитории. Многие сторонние репозитории находятся на <https://launchpad.net>, и они добавляются (если вы доверяете источнику) такой командой:

```
$ sudo add-apt-repository ppa:<имя репозитория>
```

»

## Шагните в Терминал...

Хотя стандартный рабочий стол Linux, как, например, Ubuntu Unity или Gnome, предлагает для выполнения повседневных задач знакомую среду «наведи и щелкни», всё же он далек от той мощи и универсальности, которую предлагает «оболочка» Linux, лежащая в основе среда командной строки. И благодаря *Терминалу* вы можете взаимодействовать напрямую с оболочкой, не покидая рабочего стола.

Стандартная оболочка Linux использует язык под названием BASH для запуска определенных команд с помощью серии инструментов. Каждая команда следует одной и той же базовой структуре:

```
$ имя_утилиты <команда> -опция
```

Сервисная часть команды является инструментом, который вы хотите запустить — например, `cd` для смены директории или `apt-get` для запуска инструмента установки программ. Некоторые утилиты можно запускать самостоятельно — введите, например, `ls`, и вы получите основной список всех файлов и папок в директории.

Добавив к имени команду, вы сможете указать, чего именно хотите от утилиты:

```
$ apt-get install <имя_программы>
```

Это дает указание инструменту программных пакетов установить указанную программу.

И, наконец, раздел `-опция` (она же — «ключ», «параметр» или «флаг, флажок» команды) — это место, где вы указываете одно или более предпочтений. Перед каждой опцией должен стоять один (-) или два (--) дефиса.

Например, чтобы получить более подробную информацию о содержании текущей папки, введите команду

```
$ ls -l
```

Флаг `-l` обеспечит вам дополнительные сведения, например, о разрешениях на файлы и владельцах этих файлов, а также даты последнего изменения и объем файлов в байтах.

# Топ-10 системных подсказок

Употребляйте Терминал с пользой при помощи этой подборки команд.

**О**т использования *Терминала* до исследования файловой системы и выяснения подробной информации по статусу вашего оборудования, знакомство с этими командами совершенно необходимо.

## 1 Подсказка

Есть три способа получить подсказку в *Терминале*:

Знаете имя инструмента? Используйте флаг `--help`:

```
$ ls --help
```

Еще одной полезной командой для получения помощи является инструмент `whatis`, который предлагает вам краткое описание полностью сформулированной команды:

```
$ whatis apt-get install vlc
```

Он опишет инструмент `apt-get`, аргумент установки, и что такое пакет `vlc`. Однако помните, что `whatis` игнорирует все флаги.

И, наконец, *Терминал* также дает доступ к полноценным материалам онлайн через утилиту `man`. Начните с `man intro`, чтобы прочитать длинное и подробное введение в сам Терминал, и `man man`, чтобы получить рекомендации о навигации по справочнику. И, наконец, приложите имя определенного инструмента — например, `man apt-get` — для получения подробного описания этого инструмента, написанного программистами.

## 2 Навигация по файловой системе

Откройте *Терминал*, и вы начнете со своей домашней папки. Используйте команду `ls -l` для подтверждения. Если бы вы хотели

```
nick@nick-ubuntu:~$ apt-get --help
apt 1.0.1ubuntu2 for amd64 compiled on Jan 12 2016 20:13:58
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 libc6 not libc6.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
```

При необходимости Терминал предлагает огромное количество документации и подсказок.

просмотреть все скрытые файлы и папки (это файлы, имена которых начинаются с точки — например, `.hidden.sh`), используйте `ls -a`. Вы также можете просматривать содержимое подпапок в текущей папке, с помощью `ls --recursive`.

Команда `cd` позволяет быстро перемещаться по директориям. Для перемещения в подпапку внутри текущей папки используйте `cd <имя_подпапки>` (помните, что в Linux для имен папок и файлов важен регистр). Чтобы вернуться на уровень назад, используйте `cd ..` или `cd ../..` для перехода на два уровня вверх. Для перехода в корневую директорию послужит `cd /`, а для перехода в свою директорию `/Home` введите `cd ~`. Введите `cd /путь/к/папке`, чтобы напрямую перейти в другую папку, используя ее точный путь (или `cd ~/Documents`, если она внутри вашей папки `Home`). И, наконец, введите `cd` — для возврата к предыдущей директории, где вы были. Если путь к вашей папке содержит пробелы — например, папка называется `Star Trek`, надо заключить весь путь в одиночные кавычки: `$ cd '~/Videos/Star Trek'`. Очень удобно, что можно нажимать на клавишу `Tab` для автодополнения имен файлов.

## 3 Копирование, перемещение и удаление

Вам надо работать с файлами в *Терминале*? Начните со знакомства с инструментами `cp` (сору — копирование) и `mv` (move — перемещение), у которых одинаковый синтаксис:

```
$ cp -i исходный_пункт пункт_назначение
```

И исходная, и конечная точки могут быть полными путями к папке, или просто используйте имя файла, если вы уже перешли в папку, содержащую исходный файл; а флаг `-i` обеспечит получение уведомления при попытке переписать существующий файл с тем же именем. Например:

```
$ cp -i invoice.odt ~/Documents/Backup
```

Если вы хотите переименовать скопированный файл, просто присоедините новое имя после пункта назначения: например, `~/Documents/Backup/invoicebackup.odt`. Подобным же образом, `cp invoice.odt invoicebackup.odt` создаст переименованный дубликат файла в той же папке. Подставьте `mv` для перемещения или переименования файла. Желая скопировать папку, используйте флаг `-r` для отображения рекурсии — это означает, что папка и ее содержимое будут скопированы или перемещены.

## 9 Получение системной информации

*Терминал* предлагает множество инструментов для выяснения подробностей о вашем оборудовании. Начните со следующей команды для получения сведений о вашем оборудовании в виде таблицы (список оборудования):

```
$ sudo lshw -short
```

Можно также вывести полноценный подробный отчет для просмотра в браузере:

```
$ sudo lshw -html > sysinfo.html
```

Среди похожих инструментов — `lscpu` (list processor information — список информации о процессоре), `lsblk` (list block storage — список блочного хранения), `lspci` (list internal hardware — список внутреннего оборудования) и `lsusb` (устройства USB). Все они работают немного по-разному — например, `sudo lspci -v` предоставляет

удобную сводку, тогда как `lsusb` работает лучше всего, если сначала вы укажете устройство, информацию о котором хотите получить, через `sudo lsusb`, а затем используете следующую команду, основанную на шине и номере устройства:

```
$ sudo lsusb -D /dev/bus/usb/00x/00y
```

И последний инструмент — `dmidecode` — может предоставить полезную информацию через флаг `-t` и ключевые слова, включая `bios`, `baseboard` (motherboard) [материнская плата], `processor` [процессор], `memory` [оперативная память] и `chassis` [шасси] — например, `sudo dmidecode -t memory`. Помните, однако, что поскольку используемые им таблицы DMI не всегда точны, не стоит удивляться, если что-то не будет согласовываться.

```
lshw: /dev/bus/usb/001/002
id:          0000:00
description: 1.1 device
physical id: 4
```

Используйте `lshw` для создания файла HTML со сводкой по оборудованию вашего ПК.

Удалите отдельные файлы с помощью `rm`, а пустые папки можно удалять с помощью команды `rmdir` (добавьте флаг `-r` для удаления папки и всего существующего содержимого):

```
$ rmdir -r ~/Documents/Backup
```

И, наконец, создайте новые папки с помощью команды `mkdir` имя\_папки (опять же, поддерживаются полные пути) и используйте `touch` имя\_файла для создания нового пустого файла — например, `touch config.sys`. При создании папок с пробелами в названии не забудьте заключить их в одиночные кавычки или использовать обратный слэш (\) перед этим самым пробелом, например: `mkdir Star\ Trek`.

## 4 Смена разрешений

*Nautilus* предлагает несколько инструментов для настройки разрешений, но быстрее будет использовать инструмент `chmod`, со следующим синтаксисом:

```
$ chmod 777 file
```

Естественно, `file` — это имя файла, что может означать также папку. Если вы хотите настроить всё содержимое папки — включая поддиректории — на те же разрешения, что у самой папки, добавьте флаг `-R`, вот так:

```
$ chmod -R 777 folder
```

Ключевой элемент здесь — число `777` (никогда не используйте его для системных файлов), которое относится к устанавливаемым вами разрешениям. `7` означает `gwx` (т.е. полный доступ), и первая цифра относится к владельцу файла, вторая — к группе файла, а третья — ко «всем остальным», как уже говорилось ранее в этой статье.

Стоит также запомнить цифру `6` (`rw-`, или чтение/запись, но не исполнение), `5` (`rx-`, или чтение и исполнение), и `4` (`r--`, или только чтение). Так, присвоив папке или файлу `644`, вы дадите владельцу права чтения и записи, но ограничите всех остальных только чтением. Установите `775`, и вы предоставите владельцу и указанной группе полный доступ, а все остальные смогут читать и исполнять файл. Прежде чем приступить к выдаче разрешений, прочитайте <https://help.ubuntu.com/community/FilePermissions>.

## 5 Смена пользователя и владельца

Желая изменить владельца или группу выбранного файла или папки, используйте команду `chown`:

```
$ chown имя_пользователя имя_файла
```

Можно использовать `chgrp` для смены группы файла с помощью того же синтаксиса (`chgrp` название\_группы имя\_файла), или `chown` для смены и пользователя и группы одной командой, вот так:

```
$ chown имя_владельца:название_группы имя_файла
```

## 6 Запуск нескольких команд

Есть три способа запустить две или более команд вместе в одной строке (что позволит вам пойти попить чайку, пока ваш ПК будет выкалывать). Если вы хотите, чтобы вторая команда сработала

```
nick@nick-ubuntu:~$ cd Documents
nick@nick-ubuntu:~/Documents$ ls
Doctor Who
nick@nick-ubuntu:~/Documents$ rmdir Doctor\ Who
nick@nick-ubuntu:~/Documents$ ls
nick@nick-ubuntu:~/Documents$ mkdir Doctor Who
nick@nick-ubuntu:~/Documents$ ls
Doctor Who
nick@nick-ubuntu:~/Documents$ rmdir Doctor Who
nick@nick-ubuntu:~/Documents$ ls
nick@nick-ubuntu:~/Documents$ mkdir 'Doctor Who'
nick@nick-ubuntu:~/Documents$ ls
Doctor Who
nick@nick-ubuntu:~/Documents$ █
```

только при успешном выполнении первой, используйте аргумент `&&`, таким образом:

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

Если вам надо, чтобы вторая команда сработала, только если первую выполнить не удалось, поставьте `||` вместо `&&`. И, наконец, используйте точку с запятой (`;`) для запуска второй команды независимо от того, что произойдет с первой:

```
$ sudo apt-get update ; sudo apt-get remove vlc
```

## 7 Редактирование в папо

Одним из популярных способов использования *Терминала* является редактирование файлов настройки. Лучшим инструментом для этого является `nano`, который работает, как обычный текстовый редактор. Запустите его с помощью

```
$ sudo nano /path/filename
```

Вы увидите, что *Терминал* превратился в интерфейс `nano` — имена файлов и пути перечислены вверху, а список основных команд показан внизу — символ `^` относится к клавише `[Ctrl]`.

Под содержание вашего файла отводится основная область интерфейса `nano` — здесь нет заверствывания текста, поэтому ищите справа символы `$`, указывающие на то, что строка не соответствует размеру (измените размер окна или нажмите клавишу `[End]` для перехода в конец строки).

Используйте `[Home]/[End]` для перехода в начало и конец строки, или `[PgUp]/[PgDn]` для перемещения по странице. Нажмите `[Alt] + [↓]` для перехода вниз документа, и `[Alt] + [↑]` для возврата наверх. Используйте `[Ctrl] + [W]` для поиска по файлу, а `[Alt] + [W]` нажимайте для поиска дополнительных совпадений.

## 8 Запись на диск USB

А если вам необходимо создать загрузаемый диск USB из скачанного вами файла ISO? На помощь придет инструмент `dd`, но мы советуем использовать его с осторожностью — при операции копирования следующая команда затрет целевой диск, поэтому убедитесь, что вы правильно указали диск перед началом с помощью `sudo fdisk -l`:

```
$ sudo dd if=filename.iso of=/dev/usb status=progress
```

Естественно, надо заменить `/dev/usb` на должный диск (скажем, `/dev/sdc`).

» Будьте внимательны при создании папок с пробелами в имени — мы покажем, как это делается правильно.

»

# 10 Повысить привилегии доступа через Sudo

Открывая *Терминал*, вы входите в оболочку через свою учетную запись пользователя. Некоторые команды требуют повышенных прав доступа в качестве суперпользователя-`root` (ищите ошибку "Permission denied [доступ запрещен]"), и этого можно достичь, поставив перед командой `sudo`, вот так:

```
$ sudo apt-get install vlc
```

Вам предложат ввести пароль для вашей учетной записи, и теперь команда будет запускаться — теперь вы сможете использовать команду `sudo` снова

во время сессии *Терминала* без необходимости вводить свой пароль. Кстати, если вы забудете указать `sudo` перед командой и она откажется выполняться, просто введите `sudo █` и нажмите `[Enter]`, чтобы повторить команду уже с приставкой `sudo`.

Если вы хотите запустить приложение рабочего стола с правами `root` (например, менеджер файлов *Nautilus*), используйте инструмент `gksu` (сначала установите его):

```
$ sudo apt-get update && sudo apt-get install gksu
```

```
$ gksu nautilus
```

*Nautilus* откроется в обычном виде, но с повышенными правами доступа. Закончив, закройте окно приложения, затем нажмите `Ctrl + C` в *Терминале*, что прервет текущее исполнение программы и вернет вас в командную строку.

Некоторые дистрибутивы позволяют с помощью инструмента `su` войти в командную строку от имени `root`, но Ubuntu блокирует его, и неспроста. Пользуйтесь вместо этого инструментом `sudo`.

# Запуск Linux

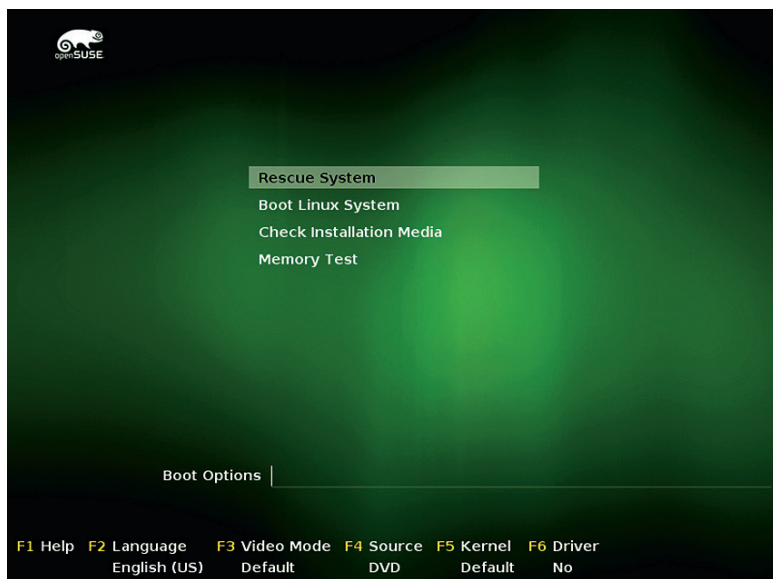
А хотите узнать, что происходит, когда вы загружаете свой Linux-ПК? Нейл Ботвик анализирует процесс загрузки, шаг за шагом.

**Н**ет, это не очередная статья для начинающих о том, как приступить к работе в Linux. Она о том, как Linux запускается. Часть прелести Linux в том, что вы как пользователь имеете полный контроль над всей системой, если вы этого хотите. Многим пользователям это кажется каким-то волшебством, где вместо дыма или зеркал основную работу скрывает экран заставки. Мы рассмотрим весь процесс, начиная с того момента, когда вы нажимаете кнопку питания, и до появления рабочего стола. Его можно разбить на несколько стадий:

- » POST » Программа загрузки
- » Загрузка ядра
- » Начальный гат-диск
- » Инициализация устройств
- » Init » Запуск сервисов и уровней запуска
- » Менеджер отображения
- » Запуск рабочего стола

Мы рассмотрим все стадии по очереди, выясняя, что происходит и какой у вас есть выбор. Как и со многими аспектами Linux, с каждой частью процесса существуют альтернативные способы работы.

Давайте начнем с обзора того, что входит в каждую стадию.



» Меню загрузки *Grub* — наше разделено на темы, но многие являются просто белым текстом на черном фоне. Загрузчик нужен на старых, до-UEFI, системах.

POST — это Power On Self Test, процедура, которая проводится оборудованием компьютера и включает проверку памяти, поиск подключенных дисков и т.д. Именно на этой стадии вы можете придержать клавишу входа в меню настройки оборудования или вызвать всплывающее меню для выбора устройства загрузки.

Программа загрузки предлагает меню, где в системах с двойной загрузкой вы выбираете операционную систему и прочие опции загрузки. Иногда меню скрыто и ждет несколько секунд, чтобы загрузиться по умолчанию. Обычно его можно увидеть, если в этот момент нажать на пробел. Затем программа загрузки загружает ядро, передавая ему некоторые опции, например, размещение раздела вашей системы и обычно также загружает начальный гат-диск. Это загружаемый в память образ диска, который содержит программы, необходимые для инициализации вашего оборудования.

Загрузившись, ядро запускает программу под названием *init*, которая начиная с этого момента будет контролировать всё. На этой стадии запускаются сервисы — это фоновые программы, например, демон *sshd* или системный журнал. Большинство дистрибутивов определяют разные уровни запуска [*runlevel*]; это одна из опций меню загрузки, которая определяет, как должен загружаться компьютер. Например, традиционно *runlevel 1* был режимом одного пользователя, 3 — многопользовательским текстовым режимом, а 5 выдавал графический рабочий стол. Менеджер загрузки — это программа, которая просит вас ввести информацию для входа в систему и может разрешить вам выбрать графической среде рабочего стола перед ее загрузкой. И, наконец, загружается ваш рабочий стол, возможно, запуская некоторые программы уровня пользователя и позволяя вам начать использовать компьютер.

## Запуск

О POST вряд ли можно сказать многое. Вам надо прочитать справочник по вашему оборудованию, чтобы узнать об имеющихся опциях. Он загружает базовую прошивку, которая на старых системах называлась BIOS. Вы можете делать всякие умные вещи, например, разогнать свой компьютер — если у вас хватит отваги, или сбросить всё в установки по умолчанию — если не хватит. Стоит заметить, что производители материнских плат предлагают обновления прошивки для исправления ошибок, и эти обновления не менее важны, чем обновления любых программ в вашей операционной системе.

Теперь мы подошли к одному из тех моментов, когда нужно делать выбор. В старых системах BIOS оборудование загружало программы с первых 446 байтов жесткого диска для загрузки компьютера. Затем этот крошечный кусочек кода в Master Boot Record

## Монтирование файловых систем

Одна из первых вещей, которые делает *init* — монтирование вашей файловой системы. При запуске ядро смонтирует вашу файловую систему *root*, но она часто монтируется только для чтения. Монтирование файловых систем управляется содержимым */etc/fstab*, которое перечисляет устройства для монтирования, их точки монтирования и все опции для использования при монтировании. Корневая

файловая система заново монтируется для чтения и записи, затем монтируются другие локальные файловые системы, включая подкачку. Любые файловые системы с включенной опцией *noauto* не монтируются, а сетевые файловые системы оставляются до того момента, когда будет доступна сеть.

Хотя традиционно эта задача решалась *init*, недавно ситуация изменилась. Больше нельзя

гарантировать, что всё необходимое для ранней загрузки находится в корневой файловой системе, в частности, в системах, которые используют отдельную файловую систему для */usr*, поэтому большинство дистрибутивов теперь монтируют локальные файловые системы, или, по крайней мере, критически важные, из *initramfs*, а уж потом передают управление *init*.



» Вот экран менеджера загрузки rEFInd для систем UEFI. Во многих случаях он определит ваши ОС вообще без всякой настройки.

(MBR, загрузочная запись) должен отыскать на жестком диске остальную часть программы загрузки; чаще всего это *Grub*. Загрузчик *Grub*, Grand Unified Boot Loader, предлагает меню, где можно выбрать опции загрузки. Обычно есть опция по умолчанию, которая активируется, если вы не сделали выбор за установленное время. Есть и другие программы загрузки, например, *LILO*, но *Grub* стал стандартом, потому что он работает с разным оборудованием и может автоматически генерировать свои меню, что важно при установке нового дистрибутива, особенно для двойной загрузки.

Файл меню *Grub* находится в `/boot/grub/grub.cfg`. Хотя его можно редактировать вручную, рекомендуется изменить настройки в `/etc/default/grub` и затем запустить `$ sudo grub-mkconfig -o /boot/grub/grub.cfg` для создания нового файла меню. Некоторые дистрибутивы скрывают меню, показывая его только при нажатии на клавишу. Это, как и время ожидания, можно изменить, отредактировав `/etc/default/grub` как описано выше.

*Grub* — это просто загрузчик, так что потом он вызывает то ядро операционной системы, которое вы ему укажете. Однако на современном оборудовании с UEFI в этом уже нет необходимости: при соответствующей настройке ядро может загружаться напрямую. UEFI рассчитывает, что первый раздел вашего диска содержит файловую систему FAT, и ищет там все загружаемые ядра. Загрузится ядро по умолчанию, которое настроено в прошивке, или можно нажать на клавишу для вывода меню со всеми опциями.

Это устраняет необходимость в загрузчике, однако большинство людей всё равно предпочитают иметь менеджер загрузки — это дает им возможность выбора операционной системы или ядра при загрузке. Вы по-прежнему можете использовать для этого *Grub*, но это отчасти перебор. Программа *bootctl* в *Systemd* использует простой файл настройки для создания опций меню, которые появляются в простом текстовом меню. В качестве альтернативы можете использовать *rEFInd* ([www.rodsbooks.com/refind/](http://www.rodsbooks.com/refind/)), который использует симпатичное графическое меню и обычно может создать его автоматически, основываясь на том, что имеется у вас на жестком диске.

## Ядра и ram-диски

Одна из причин использования менеджера загрузки даже при наличии системы UEFI, способной напрямую загрузить ядро,

заключается в передаче параметров ядру. Обычно вы можете просмотреть параметры, используемые для загрузки ядра на работающей системе, с помощью

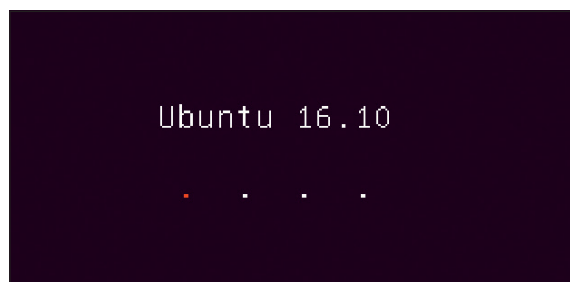
```
$ cat /proc/cmdline
```

Некоторые из этих опций косметические — например, те, которые загружают всплывающий экран; но некоторые критически важны. Например, параметр `root=` сообщает ядру, на каком разделе содержится операционная система — от ядра без остальной части ОС довольно мало прока. Опции ядра также могут определять систему *init* для использования, к чему мы вскоре подойдем, или указывать путь к `initrd`.

Linux имеет своего рода монолитное ядро. Монолитное ядро содержит весь код и драйверы, необходимые для загрузки ОС, в противоположность микроядру, которое содержит только основной код и загружает всё остальное по требованию. Споры о том, что лучше, не затихают уже долгие годы, но в конечном итоге в Linux есть и то, и другое.

Во времена, когда вы сами компилировали ядро, вы включали в него части, необходимые для загрузки системы: драйверы для контроллеров жесткого диска, используемую вами файловую систему, сетевую карту и т. д.

В результате у вас было ядро со всем, что вам требовалось, однако оно могло работать далеко не везде. В дистрибутивах должно быть всё необходимое, но их весьма раздуло бы включение драйверов для всех устройств, поэтому были добавлены загружаемые модули. Это дополнительные части кода драйвера, которые



» Всплывающие экраны скрывают внутреннюю работу процесса загрузки, и если вы хотите знать, что происходит, отключите их.

## Загрузка ОС

Linux — это ядро, тот самый файл в `/boot`. Всё остальное, остальная часть операционной системы GNU/Linux, работает поверх этого ядра. Ядро — сердцевина ОС: оно находится между оборудованием и остальной частью программ, и всё проходит через него, но оно не является полноценной ОС. Так где же именно находится операционная система?

Помните параметр `root=`, который программа загрузки передавала ядру? Там-то и начинается всё остальное — с программы под названием `init`,

которая по умолчанию находится в `/sbin/init` — это первая программа, которую запускает ядро, ID ее процесса — всегда 1, всё остальное является дочерним по отношению к ней. Это один факторов, приводимых в качестве аргумента против `systemd`, что он работает, как `init`, как PID1, и со столь критически важной частью системы не стоит шутить. Хотя `/sbin/init` является настройкой по умолчанию, вы можете предоставить ядру другое размещение, как это иногда делается в `systemd`, и даже можете изменить

это в командной строке программы установки. Если у вас проблемы с вашим компьютером, которые не позволяют ему завершить процесс загрузки, иногда их можно решить, добавив `init=/bin/bash` к опциям ядра в программе загрузки. Это означает, что ядро даже не будет пытаться загрузить `init`, и вместо этого переведет вас в оболочку в качестве `root`. Вот вам еще одна причина стараться не редактировать опции загрузки без пароля на любом компьютере, который не находится за закрытыми дверями.

загружаются по требованию; ваш компьютер загрузит драйвер для вашей сетевой карты и не будет трогать остальных, что займет очень небольшое место на жестком диске.

Это оставляет вопрос о том, как монтировать ваш жесткий диск, если драйверы хранятся в виде модулей на этом самом диске? Вот так и возник начальный `ram-диск` [initial ram disk]. Это файл, содержащий образ диска со всем необходимым. Он хранится в той же директории, что и ядро, и поэтому, как и ядро, вызывается загрузчиком. Ядро монтирует этот диск во временном местоположении, загружает с него необходимые модули, а затем размонтирует диск и переключает корневую директорию на окончательное местоположение. Насколько велик этот `ram-диск`, в разумных пределах неважно, поскольку он загружается в память всего на несколько секунд и потом сразу удаляется. Именно `ram-диск`

## Одно из первых действий ядра — сканирование оборудования и загрузка его драйверов.

заботится обо всем, что следует сделать для получения оставшейся части ОС, даже если она находится на носителе экзотического формата или где-то в сети.

Образ `ram-диска` можно бы встроить в файл ядра, но все стандартные системы, применяемые дистрибутивами, используют для него отдельный файл. Загляните в свою директорию `/boot`, и вы увидите соответствующий файл `initramfs` или `initrd` для каждого ядра. Строго говоря, в настоящее время ядро использует `initramfs`, но термин `initrd` пока не хочет выходить из обращения.

```

[ OK ] Found device qemu-hnrc00jrk 5.
[ OK ] Listening on Load/Save RF Kill Switch Status /dev/rfkill Watch.
Starting Braille Device Support...
Starting TUI Plymouth to Write Out Runtime Data...
Reloading swap /dev/disk/0p-uid/0e4b25e-66ab-4093-9a73-09a72e955245...
Mounting Arbitrary Executable File Formats File System...
Starting Nameserver information manager...
[ OK ] Reached target System Time Synchronized.
Starting Update UTPM about System Boot/Shutdown...
[ OK ] Activated swap /dev/disk/0p-uid/0e4b25e-66ab-4093-9a73-09a72e955245...
Mounted Arbitrary Executable File Formats File System.
[ OK ] Started Braille Device Support.
[ OK ] Started TUI Plymouth to Write Out Runtime Data.
[ OK ] Started Enable support for additional executable binary formats.
[ OK ] Started Nameserver information manager.
[ OK ] Started Update UTPM about System Boot/Shutdown.
Starting Raise network interfaces...
[ OK ] Reached target Swap.
[ OK ] Reached target System Initialization.
[ OK ] Started ACPI Events Check.
[ OK ] Started Daily apt activities.
[ OK ] Listening on UID daemon activation socket.
[ OK ] Listening on Avahi mDNS/DNS-SD Stack Activation Socket.
[ OK ] Listening on CUPS Scheduler.
[ OK ] Listening on D-Bus System Message Bus Socket.
Starting Console System Startup Logging...
[ OK ] Started CUPS Scheduler.
[ OK ] Started Trigger resolvconf update for networkd DNS.
[ OK ] Reached target Paths.
[ OK ] Listening on RFID Listen Socket.
[ OK ] Reached target Sockets.
[ OK ] Reached target Basic System.
    
```

▶ Вот что вы получаете, отключив экран-заставку. Пробегающие по экрану зеленые OK — это очень хороший знак.

Разница — в структуре, лежащей в их основе; но с точки зрения пользователя они работают одинаково.

Одно из самых первых действий ядра — сканирование имеющегося оборудования и загрузка его драйверов. В отличие от других операционных систем, где установка драйверов является отдельным процессом, Linux по умолчанию включает драйверы для большей части оборудования. Вы можете видеть их в сообщениях ядра, отображаемых при запуске команды

```
$ dmesg | less
```

Вы увидите, что на ранней стадии ядро определяет CPU, материнскую плату и память, а затем переходит к поиску остальных устройств. Ко времени начала загрузки операционной системы всё ваше оборудование уже должно быть доступно для нее.

## Запуск сервисов

Если вам нужно лучше представлять, что происходит при загрузке вашего компьютера, избавьтесь от всплывающего экрана при загрузке, который скрывает всю работу. Обычно это делается посредством нажатия на Esc при его появлении, или можете использовать редактор меню загрузки для удаления всех ссылок на экран-заставку при загрузке из опций ядра. Загружаясь, вы увидите строки текста, большая часть которых сообщает, что всё идет хорошо. Это запускаются сервисы.

Сервисы, также называемые демонами — это программы, которые работают в фоновом режиме, и каждая из них выполняет свою задачу. Это могут быть сервисы, подобные NTP (Network Time Protocol), который обеспечивает точность часов вашего компьютера, или `cron`, сервис запуска команды с постоянным интервалом, или системный журнал [system logger], дающий средства или записывающий сообщения о статусе от других программ на вашем компьютере. Эти сервисы работают на большинстве компьютеров; кроме них, есть демоны сервера, web-сервер, серверы электронной почты и т.д.; они используются для предоставления сервисов

```

# cat /etc/fstab
# fstab (Modified)
# Row 24 Col 1
/dev/nvme0n1p1 /boot vfat defaults
/dev/nvme0n1p2 none none SW
LABEL=looon / btrfs noatime
LABEL=looon /usr btrfs noatime,subvol=usr
LABEL=looon /var btrfs noatime,subvol=var
LABEL=looon /home btrfs noatime,subvol=home

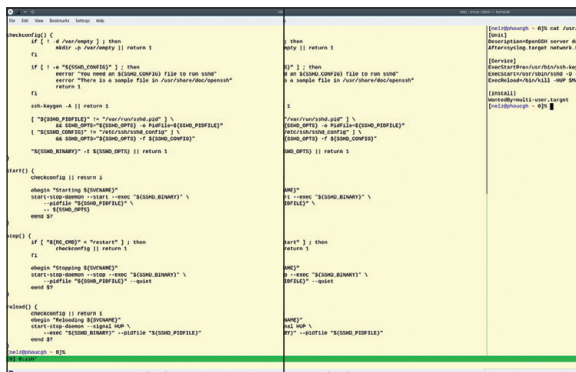
tmpfs /tmp tmpfs noatime,mode=1777,size=

LABEL=quawl /quawl btrfs noatime,subvol=id=0,noou
LABEL=quawl /lxfvdvd btrfs noatime,subvol=lxfvdvd
LABEL=quawl /snapshots/quawl btrfs noatime,subvol=snapshot
LABEL=quawl /mnt/scratch btrfs noatime,subvol=scratch
LABEL=quawl /mnt/duplicity btrfs noatime,subvol=duplicit
LABEL=quawl /mnt/vm btrfs noatime,subvol=vm
LABEL=quawl /containers btrfs noatime,subvol=containers

lunkwill:/mnt/portage /mnt/portage nfs4 defaults
lunkwill:/mnt/backup /mnt/backup nfs4 defaults
lunkwill:/mnt/torrent /mnt/torrent nfs4 defaults
vroomfondel:/storage/recordings /mnt/mythtv/recordings nfs4 defaults,noauto
vroomfondel:/storage/videos /mnt/mythtv/videos nfs4 defaults
    
```

▶ Файл `/etc/fstab` укажет, какие файловые системы монтировать и где. Именно этот — явно чрезмерный, но он лучше выглядит на снимке, чем `fstab` по умолчанию.





» Сравнение традиционного скрипта *init* (слева) с файлом *systemd unit*. В окне показана примерно половина скрипта *init*.

другим компьютерам, которые вы, возможно, не запускаете из рабочего стола.

О работе этих сервисов заботится менеджер сервисов. Это сложная задача: сервисы должны запускаться при загрузке компьютера и останавливаться при выключении или перезагрузке, но это далеко не всё. Есть сервисы, зависящие от других, и порядок их запуска важен. Также следует проверять, что сервисы продолжают работать и что в случае отказа какого-то сервиса будут предприняты меры.

Традиционный менеджер сервисов для Unix и Linux основан на *SysVinit*, ведущего начало со времен Unix. Он использует набор уровней запуска, каждый из которых представляет определенное состояние системы. Стандартные уровни запуска:

- 0 Останов
- 1 Режим одного пользователя
- 2 Многопользовательский режим
- 3 Многопользовательский сетевой режим
- 4 Не используется
- 5 Как уровень 3 плюс менеджер отображения
- 6 Перезагрузка

Сервисы запускаются и останавливаются скриптами, которые хранятся в */etc/init.d*. Папки имеются также для всех уровней запуска, от */etc/rc0.d* до */etc/rc6.d*. Они содержат символические ссылки на скрипты в */etc/init.d*. Поэтому когда вы загружаете компьютер на уровне запуска 3, запускаются все скрипты из */etc/rc3.d*. При запуске они стартуют по порядку и передаются с аргументом "start". При выключении или перезагрузке они отработывают в обратном порядке с аргументом "stop".

А как система узнает правильный порядок запуска сервисов? Это делается по именам скриптов — все они начинаются с цифр, что облегчает задачу определения порядка.

Итак:

- S01syslog-ng
- S02cron
- S03firewalld

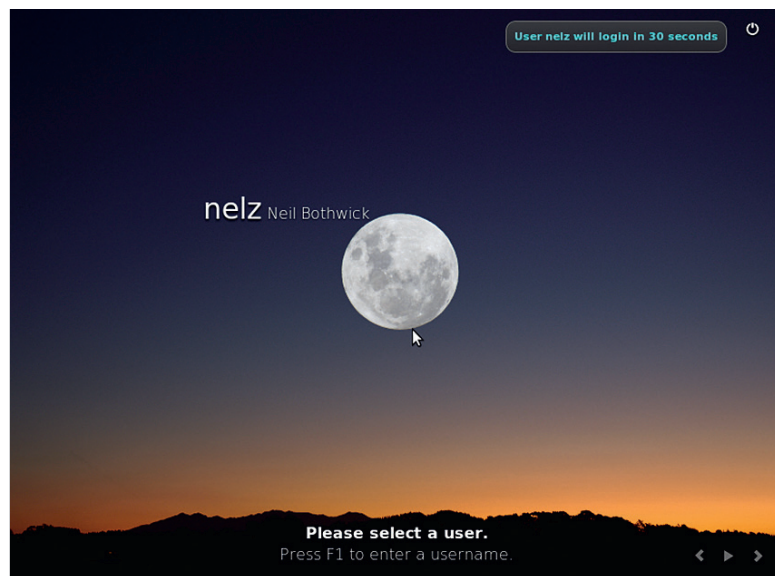
Это означает, что первым запускается системный журнал, после него — демон *cron*, а затем брандмауэр. Скрипты *init* запускаются серийно, поэтому один из них должен завершиться прежде, чем запустится другой. Это довольно дубовая система, способная затормозить процесс загрузки, если у какого-то одного сервиса уходит на запуск много времени, но она используется еще с тех пор, когда Линус Торвалдс был юн, и она надежна.

Обратите внимание, что детали того, как каждый дистрибутив реализует *SysVinit* и его уровни запуска, могут различаться как по размещению скриптов, так и по определению уровней запуска.

## На новенького

Есть ряд альтернативных менеджеров сервисов, таких как *Upstart* (ныне покойный) и довольно привлекательное предложение от Gentoo — *openrc*; но они не обрели особой популярности, за одним лишь исключением. Вызывающий горячие дискуссии (причем под этим также надлежит понимать споры, ссоры, а то и оскорбления) *systemd* пытается устранить некоторые из ограничений *SysVinit*.

Одно из ключевых отличий в том, что *SysVinit* использует для запуска и остановки сервисов скрипты оболочки. Каждый сервис имеет целый свой скрипт для работы, с несколькими последствиями. Немало работы дублируется, даже при том, что некоторые сервисы обеспечивают вспомогательные функции чтения скриптов, чтобы увидеть, что именно делается при перезапуске сервиса — это непростая задача, и между дистрибутивами нет единообразия, поскольку каждый использует собственные скрипты, в соответствии со своим образом действий.



» Менеджер отображения — ваши ворота на рабочий стол. У этого очень милая тема, но всё, что вам здесь нужно — это место для ввода пароля, и потом он исчезнет!

## Смена уровней запуска

Уровень запуска по умолчанию определяется в файле */etc/inittab*, и строка

```
id:5:initdefault:
```

сообщает, что эта система по умолчанию загружается на уровень запуска 5. Это можно изменить, отредактировав файл или добавив уровень запуска в список опций ядра при загрузке.

Если рабочий стол нужен вам не всегда, у вас могут быть отдельные пункты меню загрузчика, чтобы

вы могли выбирать из них. После того, как компьютер загрузился, уровни запуска можно изменить командой *telinit*. Если вы загрузились в текстовую консоль, а теперь вам нужен рабочий стол, вы можете запустить от имени *root*

```
$ telinit 5
```

В *Systemd* нет уровней запуска; их эквивалентами являются *target* [цель]: *multi-user.target* — эквивалент уровня запуска 3, *graphical.target*

соответствует 5. Эквивалентом вышеупомянутой команды *telinit* для переключения из текстового режима в графический будет

```
$ systemctl isolate graphical.target
```

По умолчанию *target* настраивается так:

```
$ systemctl set-default graphical.target
```

и вы можете указать другой *target* в строке программы загрузки с помощью

```
systemd.unit=multiuser.target
```

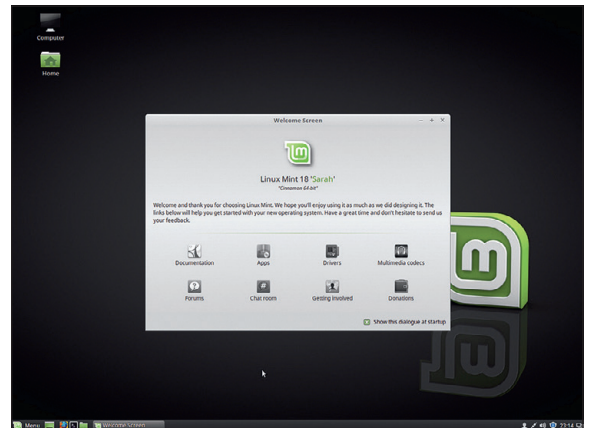
*Systemd*, напротив, использует файлы сервисов, которые просто велят *systemd* что-то делать, вместо того, чтобы делать это самим. Большинство файлов сервисов являют собой всего лишь несколько строк, и для понимания, что именно будет сделано, вполне достаточно бегло на них взглянуть. Поскольку вся реальная работа выполняется самим *systemd*, это также означает, что файлы сервисов стандартизированы, и один и тот же файл обычно годится для большинства дистрибутивов. Еще важнее, что файлы сервисов отражают зависимости, поэтому файл для *web*-сервера заявит, что сначала надо запустить работу в сети, а для брандмауэра может указывать, что он запускается перед началом работы в сети.

Итак, эти три будут запускаться по очереди, и каждый будет ждать своего предшественника; зато другие сервисы могут стартовать, когда хотят. Нет никаких причин откладывать запуск *cron*, пока сеть получает адрес от DHCP, лишь из-за того, что у *cron* большая цифра перед именем. И есть еще одно преимущество запуска программ самим *systemd*: если ему приказать, он проведет их мониторинг. Файл сервиса может сообщить системе, что делать в случае отказа программы, от ее перезапуска (использовать с осторожностью) до отправки вам уведомления.

Конечно, *systemd* делает намного больше, но это уже отдельные компоненты общей инфраструктуры *systemd*, и по большей части тут нет ничего общего с запуском сервисов.

## Запуск сети

Мы рассмотрели запуск сервисов в целом, *SysVinit* или *systemd*, но есть одна группа сервисов, заслуживающая особого внимания: это сервисы, относящиеся к работе в сети. Мы говорим не о сервисах, которые используют сеть, а о тех, которые нужны, чтобы сеть заработала. Linux использует сеть, как и большинство людей в наши дни, даже на отдельном компьютере. В процессе загрузки



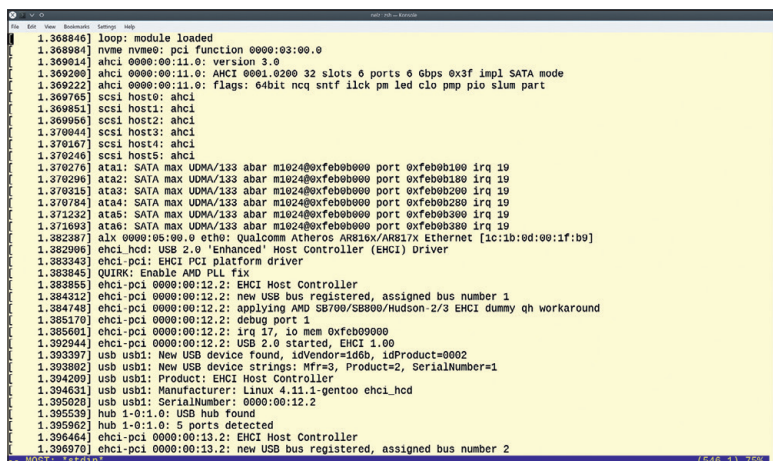
► **Наконец-то! Процесс загрузки почти завершен, и перед нами графический рабочий стол.**

участвуют три пункта, когда запускается сеть. На ранней стадии запускается интерфейс *localhost* (*lo*). Он позволяет сетевым программам общаться на локальном компьютере и не требует практически ничего больше.

Вскоре после этого вызываются внешние сетевые интерфейсы. Помните, что мы говорили о порядке загрузки сервисов? Некоторые сервисы зависят от сетевого интерфейса, поэтому он должен быть доступен перед тем, как они запустятся, и обычно вызывается довольно рано. Если у вас один проводной интерфейс, это очень просто. Как правило, дистрибутивы используют Dynamic Host Config Protocol для получения адреса и прочей необходимой информации по маршрутизации от DHCP-сервера в сети (обычно это ваш модем/роутер) и затем настраивают для вас сеть.

Если вам нужен статический адрес, кое-какие настройки придется сделать самим. Для получения статического адреса можно использовать и DHCP-сервер: большинство роутеров позволяют вам распределять конкретные адреса по MAC-адресам (MAC-адрес — это уникальный аппаратный адрес сетевого интерфейса). В таком случае вы по-прежнему будете использовать DHCP, но каждый раз получать тот же самый адрес. Альтернатива — употребить для задания статического адреса инструмент сетевой настройки вашего дистрибутива, причем адрес должен быть в том же диапазоне (подсети), что и остальная часть вашей сети. В дополнение к компьютерному адресу надо задать адрес шлюза (обычно вашего роутера) и один или более DNS-серверов. Вы можете использовать DNS своего интернет-провайдера — например, Google предоставляет публичный — или использовать адрес своего роутера, который переадресует запросы на внешний сервер.

Беспроводная сеть — дело другое, поскольку она может включать несколько сетей и аутентификационную информацию. Обычно с ней работает такая программа, как *NetworkManager*. В зависимости от настройки, ваш *NetworkManager* может найти известную



► **Команда *dmesg* показывает сообщения на ранней стадии загрузки ядра. Здесь запущены диски, сетевая карта и USB-порты.**

## Перевод компьютера в состояние ожидания и выход из него

Мы говорили о том, что происходит, когда вы делаете «холодный» запуск своего компьютера, и коснулись того, что происходит при выключении, однако это не единственный способ включить и выключить компьютер.

Спящий режим и гибернация — это возможности выключить компьютер, не выключая его. Они сохраняют текущее состояние компьютера и возобновляют его при включении. Это экономит время включения и выключения и также означает, что вы можете

продолжить ровно с того места, где остановились, и все ваши приложения остаются открытыми.

Хотя изначально спящий режим и гибернация предназначались для ноутбуков, они становятся всё более популярными и для настольных ПК. В спящем режиме компьютер входит в режим низкого энергопотребления, когда содержимое памяти сохраняется, но большинство других функций выключаются. Он позволяет быстро перезапуститься, но при этом всё же потребляется некоторое количество энергии,

потому что память продолжает работать. Гибернация перед выключением компьютера сохраняет содержимое памяти на жесткий диск, обычно в раздел подкачки. При перезапуске ядро определяет присутствие данных гибернации и снова загружает их в память.

Сочетание обоих методов означает, что вам вообще не придется выключать и перезагружать свой компьютер, если вам не надо обновлять ядро или добавлять программы.

сеть и соединиться с ней, когда загружается компьютер, или не делать этого, пока вы не войдете на рабочий стол. Последний подход практикуется по умолчанию. Он медленнее, поскольку переговоры по соединению не начинаются, пока не загрузится рабочий стол; зато он безопаснее. Обычно имеется опция глобальной настройки сети, если вы хотите, чтобы *NetworkManager* запускал ее во время запуска сервисов. Мы говорим «обычно», потому что в разных рабочих столах разные интерфейсы пользователя.

## Что такое «менеджер отображения»?

В старые «добрые» времена, когда все CPU были 32-битными и все дистрибутивы умещались на одном CD, вы загружались в текстовую консоль и входили в систему от имени своего пользователя. А потом, если вам была нужна одна из этих новых штук GUI, вы вводили *startx*. Некоторые поступают так до сих пор, однако нормально будет использовать менеджер отображения [Display Manager]. Он запускается в качестве сервиса, как и любой другой, но только из уровня запуска GUI.

Менеджер отображения — это программа, которая открывает графический экран приглашения, хотя продолжает работать в качестве сервиса после входа в систему, готовая обработать следующий запрос на вход. Есть ряд менеджеров отображения для обычного использования, включая *gdm*, *lightdm* и *sddm*. Большинство разрешают некоторую настройку, например, изменение вида экрана приглашения, но так как, по сути, всё, что вам надо сделать — это ввести свою информацию для входа, возможно, времени, потраченного на это, можно найти более эффективное применение. Если вы — единственный пользователь компьютера, то большинство менеджеров отображения можно настроить на автоматический вход при запуске. Впрочем, делая так, задумайтесь о безопасности. Ведь если ваш жесткий диск не зашифрован, любой включивший ваш компьютер войдет в систему без аутентификации.

## Запуск рабочего стола

Мы почти у цели. Программа загрузки загрузила ядро, ядро запустило *init*, и благодаря менеджеру сервисов заработали все ваши фоновые сервисы. Вы вошли через экран менеджера отображения, и что еще делать? Теперь менеджер отображения запустит сессию X и загрузит среду рабочего стола. Вы можете счесть это простым делом, но как раз тут возможны осложнения, поскольку каждая среда рабочего стола делает это по-своему. Рабочий стол загружает обои, обычно панель и одно-два меню, и часто некоторые виджеты. То, что он загружает и как вы его настроите, целиком и полностью зависит от вашего рабочего стола. На [www.freedesktop.org](http://www.freedesktop.org) приводятся некоторые стандартные механизмы, однако разные рабочие столы склонны считать их скорее тонкими намеками, чем настоящими стандартами.

Документация для вашего рабочего стола даст вам больше информации по этому вопросу, или можете покопаться в настройках программы, чтобы найти искомое. Помимо таких очевидных вещей, как обои и стили украшения окон, обычно можно настроить

индивидуальные горячие клавиши клавиатуры, добавить виджеты или значки на рабочий стол и заставить какие-то программы запускаться автоматически при загрузке рабочего стола. Кроме того, бывают дополнительные сервисы, запускаемые вместе с рабочим столом. Рассмотренные нами ранее сервисы запускаются от имени *root* и доступны для всей системы, но могут также быть сервисы, запускаемые для пользователя. Хорошим примером подобных является *PulseAudio*, который обычно запускается не от имени *root*, а вместе с рабочим столом вошедшего в систему пользователя.

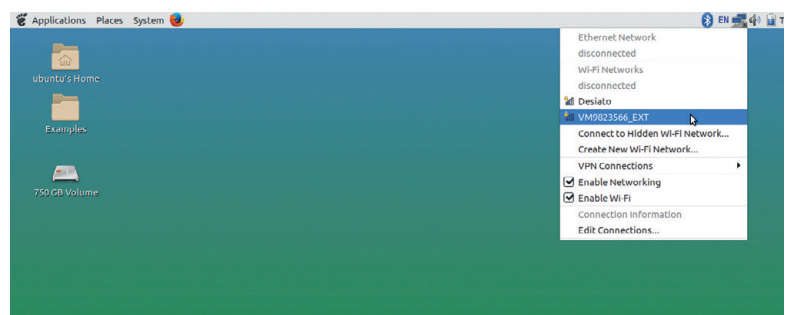
Вот и всё! Вы вошли и готовы задействовать свой компьютер. Еще важнее то, что, как мы надеемся, теперь вы лучше понимаете происходящее.

## Выключение

Мы рассмотрели в некоторых подробностях процесс загрузки; ну, а что происходит, когда вы выключаете или перезагружаете компьютер? В основном, ОС повторяет те же шаги в обратном порядке. Во-первых, запрещаются новые входы в систему, затем рабочий стол перед выходом закрывает открытые приложения и пользовательские сервисы. Менеджер сервисов останавливает работающие сервисы, в основном в обратном порядке, т.е. первым выключая менеджер отображения и в последнюю очередь размонтирует файловые системы.

Некоторые начальные *gpt*-диски оставляют свои скрипты работать в фоне, поэтому когда *init* выходит, они могут отменить всё, что запустили, затем выгружается ядро и экран пустеет. Так же, как и в процессе загрузки, если это происходит без экрана-заставки, вы увидите весь процесс выключения, что может быть полезным, если вдруг он займет больше времени, чем предполагалось. Увидев, что вызвало задержку, вы сможете это исправить, и как минимум будете знать, почему это происходит — заставка и индикаторы состояния дружелюбны к пользователю, только если всё нормально.

Сложно было, правда? В Linux главное — учиться и давать пользователю большие возможности; современные дистрибутивы супер-просты в использовании, однако от вас ничего не скрывают, чтобы вы могли испытывать всё и экспериментировать со всем. Итак, развлекайтесь, наслаждайтесь Linux и растите над собой! **LXF**



▶ С проводными сетями в процессе загрузки разговор короткий, но для соединения с беспроводной сетью, возможно, придется подождать загрузки рабочего стола.

## Советы по решению проблем

Процесс загрузки современной операционной системы определенно сложен, даже если некоторые его детали от вас скрыты. Кроме того, он хорошо проверен и по большей части должен работать нормально. Естественно, выражение «по большей части» обычно относится к другим людям и их компьютерам; так что же вы можете сделать, если у вас возникли проблемы?

Первым делом надо отключить все заставки: тогда вы сможете увидеть, что происходит. Если отказал определенный сервис, можно попробовать загрузиться на уровень запуска 1, однопользовательский режим, и затем отключить этот сервис.

В результате подобного подхода у вас может оказаться не вполне нормально работающий компьютер, но вы сможете исправить проблемы. Если

вы настроили статическую сетевую адресацию, тщательно проверьте свои настройки или временно переключитесь на DHCP.

Не пытайтесь лихорадочно менять настройки в надежде решить проблему, чтобы в конечном итоге обнаружить: причина немоты Интернета в том, что по соседству землекопы рыли канаву и перерезали кабель!

Джо Дамато

# Пакеты на проверку

Джонни Бидвелл встретился с основателем `packagedcloud`, Джо Дамато. Готовьтесь к краху своих иллюзий насчет управления пакетами...



Джо Дамато (Joe Damato) — специалист по низкоуровневой компьютерологии и создатель [packagecloud.io](https://packagecloud.io), сайта, предоставляющего бесплатный и коммерческий хостинг для пакетов данных любого формата, типа и размера: от привычных нам `.deb` и `.rpm` до более экзотичных, вроде репозитория Maven. Сайт поддерживает любые варианты сборки/оркестровки/интеграции систем, включая Chef, Puppet, Jenkins, и TravisCI и другие, и в общем облегчает работу дистрибутивов.

Мы пересеклись с Джо на OSCON Europe 2016 в Лондоне, где он сделал интереснейший доклад под названием «Код как инфраструктура буквально невозможен», рассказав о мифах, связанных с высокоуровневым кодом, и тех невидимых ужасах, которые творятся при выполнении с виду приличного кода. А еще он побаловал нас рассказами о крупных провалах известных пакетных менеджеров и о том, что они не умеют обращаться с GPG.

**Linux Format:** Чем же занимается специалист по низкоуровневым вычислениям?

**Джо Дамато:** До того, как был основан [packagecloud](https://packagecloud.io), я в основном был системным инженером. Работал с драйверами устройства, отладчиками, ядром и тому подобным. На самом деле, я до сих пор много пишу об этом в блоге [packagecloud](https://packagecloud.io). Более ранние записи есть также в моем личном блоге, [timetoblead.com](https://timetoblead.com), но он давненько не обновлялся. Один мой приятель как-то сказал, что раз компьютерная наука — значит, —логия, отсюда и компьютерология. На самом деле, так называется родительская компания [packagecloud](https://packagecloud.io).

**LXF:** Как вы подсели на компьютеры, программирование и тому подобные дурные привычки?

**ДД:** Программированием я стал заниматься где-то в старших классах, может, чуть раньше. Мои родители компьютерами и программами особо не интересовались. Мой отец работал на заводе по переработке мусора в Нью-Джерси; кто-то выкинул старый Apple IIe, и он его, естественно, спас. Тогда я впервые увидел компьютер. Да еще тот, кто выбросил машину, заодно выбросил и учебник по BASIC для детей. Так что потом я какое-то время просто перепечатывал оттуда листинги кода, не вполне понимая, как работает каждая строка — просто радовался, видя результат. В основном это были несложные игры, так что было еще и весело.

**LXF:** У нас до сих пор в журналах приводят листинги кода. И рискну сказать, многие так и копируют их, не особо вникая в смысл. Это, конечно, палка о двух концах, но, может быть, лучше учиться так, чем копипастом из Интернета.

**ТО ЖЕ, ДА НЕ ТО**

**У каждой системы пакетов свои причуды, но у них много общего.**

**ДД:** На каком-то уровне это странно, в наши-то дни. Но многим из тех, кто учится программировать, особенно молодежи, это помогает.

**LXF:** И в конце концов вы решили основать [packagecloud](https://packagecloud.io), сайт, где люди могут создавать собственные хранилища и размещать свои приложения во всевозможных форматах. Зачем вы это сделали?

**ДД:** Я создал [packagecloud](https://packagecloud.io), потому что на многих моих прежних работах мне приходилось предоставлять клиентам программное обеспечение. Как

правило, они покупали нечто вроде агента для работы на их серверах. Проблема в том, что всегда нужен механизм конвейерной сборки. Чтобы вы запускали какие-то изменения, а затем могли скомпилировать этот агент для всех версий CentOS или Ubuntu, или Debian. Эти сборки затем оказываются в репозитории, а клиенты получают к ним доступ и устанавливают. Ну и этот механизм мне приходилось переделывать кучу раз, каждый раз настраивая аутентификацию и отзывая доступ, если клиенты переставали платить.

На третий или четвертый раз я подумал: «Должен же быть способ лучше это сделать», и тогда-то, пострадав от нехватки средств автоматизации, я и создал [packagecloud](https://packagecloud.io).

**LXF:** [Packagecloud](https://packagecloud.io) поддерживает любые виды пакетов — не только `.deb` и `.rpm`, с которыми люди работают на уровне дистрибутива, но и такие

вещи, как Python wheel/egg, RubyGem и архивы Java/Android. Как вам удалось найти универсальные способы работы с ними?

**ДД:** Ну, настолько универсальных способов не существует. Однако хорошая новость в том, что у инструментария для всех этих пакетных систем есть открытый код и непло-

хая документация. Изучив ее, вы поймете, как построить хранилища и как должны генерироваться метаданные. У каждой системы управления пакетами свои причуды, но у них у всех много общего. Как правило, у большинства есть базовые свойства, такие как имена, версии, зависимости и тому подобное. Так что над ними есть хороший слой абстракции, применимый для всех более общих вещей.

**LXF:** Вы только что сделали доклад под названием «Код как инфраструктура буквально

»



невозможен», и там говорилось про жуткие вещи. В том числе о том, что пакетные менеджеры в дистрибутивах толком не проверяют GPG-подписи. Вы не могли бы рассказать поподробнее?

**ДА:** Это было проблемой с *rpm* на более ранних системах CentOS; вроде бы это исправили в CentOS 7 и более поздних версиях CentOS 6. Но какое-то время сам пакет *Yum* не зависел от *rpm*, а значит, те, у кого был установлен *Yum*, не могли проверить подписи GPG в своей системе. По сути, каждый раз, устанавливая пакет, подписанный GPG, вы не могли знать, проверен он, или нет.

**LXF:** Кошмар. Ну, ничего хорошего, тем более, что линуксоиды привыкли гордиться самой идеей управления пакетами, в качестве альтернативы загрузке и установке отдельных двоичных файлов из Интернета. Чем же успокоить бдительных пользователей пакетов, кроме как убедить их делать всё только через *packagecloud*?

**ДА:** По-моему, у вашего вопроса множество аспектов. Масса народу сейчас трудится над проектом под названием The Update Framework (<https://theupdateframework.github.io>), который должен решить кучу проблем с безопасным распространением программного обеспечения и обновлений

у нынешних пакетных систем. Возможно, отчасти дело в том, что многие переизобретают или пытаются переизобрести одно и то же, только применительно к разным инструментам, библиотекам или языкам программирования. Люди продолжают переделывать одни и те же вещи, каждый раз совершая одни и те же ошибки. Отчасти дело в отсутствии исследований или желания просто сесть и подумать, что для этого нужно.

Я считаю, что The Update Framework — по крайней мере, насколько мне известно — является одним из наиболее полных и надежных решений для безопасного распространения программного обеспечения. Я сам был из тех пользователей Linux, кто хвалил процесс управления пакетами, и думал, что с *Apt* и *Yum* всё в порядке. Но затем я прочитал статью — кажется, исследователей из Университета Аризоны, где они проиллюстрировали уязвимости менеджеров пакетов. Прямо читать было жутко, вот я и включил это в доклад.

**LXF:** Вы также затронули вопрос о том, что программируя, скажем, на Python, люди думают, что он позаботится за них обо всех низкоуровневых вещах, но иногда этого не происходит, отсюда все эти неприятные ошибки или узкие места.

Означает ли это, что начинающие программисты должны всегда обращаться к основам основ, если они желают сделать нечто, способное впоследствии расти?

**ДА:** Если вы говорите о том, чтобы начать и построить новый проект, то я определенно всегда за то, чтобы начинать с самых простых вещей, а уже потом задумываться об оптимизациях. Но к моменту, когда вы научитесь писать код, который работает действительно хорошо и очень быстро, и с нужной вам производительностью, вы должны на самом деле досконально знать свою операционную среду. До такой степени, чтобы быть старшим системным программистом, который знает всё, что надо знать о том, как устроена операционная система изнутри.

Я думаю, многие люди много лет жаловались, что язык X — слишком медленный, где X — это Ruby, или Python, или еще какой-нибудь. Но, оказывается, только понимая, как работает виртуальная машина или интерпретатор, и как написать код, который на самом деле совместим с ними, вы сможете приблизиться к совершенству. Нужно очень глубокое понимание того, как это всё стыкуется.

**LXF:** Сейчас все массово переходят на контейнеры, оркестровки и микросервисы. Якобы они имеют всевозможные преимущества, но при этом довольно сложны (я и сам в них не разобрался). Не считаете ли вы, что решив старые проблемы, это добавит новых?

**ДА:** Мое первое настоящее место работы после колледжа было в VMware, где я занимался гипервизором ESX. Виртуализация и контейнеризация — вещи разные, но, надеюсь, то, о чем я говорю, применимо и к тем, и к другим. Вкратце, моя идея насчет виртуализации заключается в том, что с виртуализацией надо быть очень осторожным, поскольку добавляется гораздо больше программ в системный стек, который, вообще говоря, для понимания, отладки и добавления кода труднее всего. В конечном счете, вы накладываете несколько операционных систем друг на друга, и вам приходится разбираться с ошибками не только в гипервизоре, но и в операционных системах хостов, и с их взаимодействием с гипервизором.

Думаю, то же можно сказать и о контейнеризации. Вы добавляете новый код в ту часть стека, которая наиболее сложна и для понимания, и для отладки. Каковы будут риски, зависит от каждого конкретного случая, так что вам как разработчику программного обеспечения, или кем вы там являетесь, надо их оценивать.

Недавно я был на конференции, где, в частности, обсуждалось и то, что многие люди используют контейнеры для разработки, но мало кто использует их в производстве — разумеется, говоря относительно. Большинство мест, в которых я работал, пока не используют их в производстве. Возможно, это придет потом. Однако в любом случае, я постоянно возвращаюсь ко всё той же идее: вы делаете



ставку на самый сложный уровень, и в этом всегда есть свой риск.

**LXF:** Вы говорили об удалении некоторых из этих слоев, но с чего начать? Есть ли уже четкое понимание того, что это за слои?

**ДД:** Я думаю, что ныне существует много интересных исследований по поводу специализированных ядер [unikernel, см. LXF215]. Я понятия не имею, насколько они будут востребованы и есть ли какие-нибудь другие системы с теми же преимуществами, например, для создания производственной системы. Но с теоретической точки зрения они выглядят очень интересно, и, возможно, из этого выйдет какое-нибудь полезное решение, которое позволит нам упростить наши системные стеки.

**LXF:** Не могли бы вы рассказать нашим читателям про баг с образом Vagrant для bento, который вы упоминали в докладе? Если я правильно понял, люди, используя один из образов CentOS, в итоге, получили машину, которая не доверяла Amazon. Так и LXF может оказаться в полете.

**ДД:** Конечно. Был образ виртуальной машины, который, я полагаю, собирался из кучи скриптов. В процессе они обновляли комплект сертификатов CA непосредственно с основного хоста — [curl.haxx.se](http://curl.haxx.se), откуда, как должен знать каждый, поставляется список всех доверенных сертификатов. Проблема была в том, что запустив этот сервис, люди перезапустили скрипт, чтобы восстановить этот пакет, но по какой-либо причине, может быть, ошибка или еще что-то [связано это было с тем, что Mozilla пометила сертификат как слабый, см. <https://blog.chef.io/2015/02/26/bento-box-update-for-centos-and-fedora/>], но CA Amazon Web Services был из него удален. В результате полученный образ виртуальной машины вообще не мог обращаться к AWS или S3. Для многих это стало огромной проблемой, ведь на нарушенные SSL-сертификаты вы не подумаете, и вместо этого начнете отладку своей AWS-библиотеки, кода приложения, вашей сети, и только потом, пролив море слез, узнаете, в чем была причина.

**LXF:** То есть еще раз, это не повлияло на набор CA-сертификатов, установленный на большинстве машин наших читателей?

**ДД:** Верно. Это затронет вас только в том случае, если вы используете именно этот образ или ска-



таковой с официального зеркала. Существуют ли меры предосторожности, чтобы такого больше не повторялось?

**ДД:** Честно говоря, сам не знаю. Думаю, тут стоит попытаться это заново повторить, и причем повторить уже на других пакетных системах. Это из всё той же статьи из Университета Аризоны [см. <https://www2.cs.arizona.edu/stork/package-managersecurity/papers.html>], о которой я упоминал; не знаю, были ли исправлены те недостатки, о которых там говорилось. Но сами эти результаты уже поразительны, поскольку пользователи наивно рассчитывают, что подобные вещи априори безопасны.

Один из примеров, которые я не включал в доклад, о том, как быть людям, которые думают создать, к примеру, сайт, и им надо где-то хранить списки имен пользователей и паролей. Все советы сводятся к тому, чтобы не применять собственное шифрование и не изобретать велосипед. Я бы рекомендовал не делать того же в отношении хранения и распространения программного обеспечения.

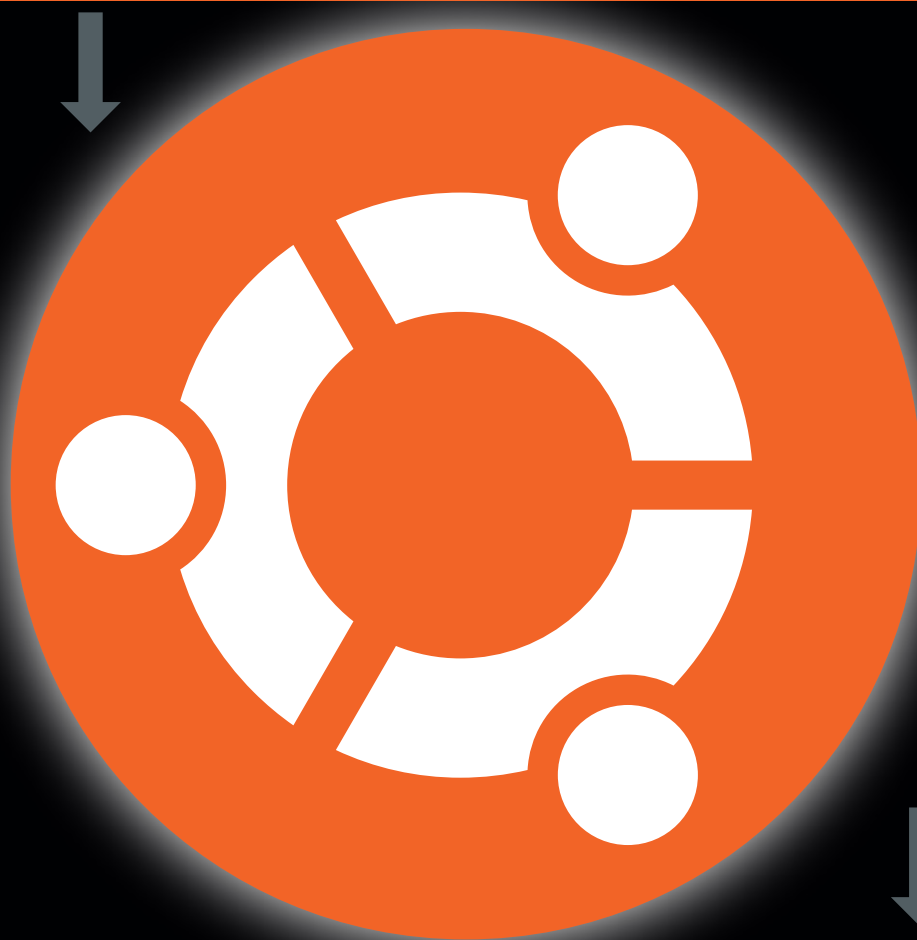
Шифрование и проверку трудно сделать правильно — как показывает история, даже у именитых проектов были сложности: все эти проблемы безопасности, о которых я говорил, повторные атаки, атаки DDoS уже опубликованы. Поэтому, возможно, тем, кто занимается управлением пакетами, стоит приглядеться к TUF или другим исследованиям в этой области, вместо того, чтобы пытаться что-то изобрести самостоятельно. Потому что сделать это правильно очень сложно. **LXF**

## О СЛОЖНОСТЯХ ПРОГРАММ

### Сначала сделайте простейшее, а уже потом думайте про оптимизацию.

чиваете напрямую из <https://curl.haxx.se>. Во всех прочих случаях, имеющийся у вас набор сертификатов, вероятно, меняется не так часто, поэтому шансы на ошибку невелики.

**LXF:** Вы также показали, что можно не только создать вредоносный репозиторий, но и получить



# Ставим Ubuntu

**Ник Пирс откроет всё, что надо знать о безопасной установке Ubuntu на ваш ПК, без уничтожения Windows (а захотите — сами уничтожите).**

**В**ышел Ubuntu 17.04, и это лучший момент, чтобы выбросить Windows и перейти на Linux. В этом кратком — но содержательном — руководстве мы покажем, как безопасно установить Ubuntu (и другие дистрибутивы) вместе с Windows, позволяя вам продолжить использовать обе операционные системы бок о бок на одном компьютере.

Конечно, вы можете решиться пойти до конца и полностью ликвидировать Windows, или же, если у вас есть более старый ПК, установить на него Ubuntu в качестве отдельной ОС (он менее требователен, чем Windows, так что это отличный способ вдохнуть новую жизнь в преданного старого служаку, избавиться от которого вы не готовы). В любом случае, мы покажем вам, как безопас-

но стереть Windows с диска и установить Ubuntu на его место.

Переход на новую операционную систему не так страшен, как кажется непосвященным, но предпринимать меры предосторожности имеет смысл всег-

## Способ вдохнуть новую жизнь в старого служаку, избавиться от которого вы не готовы.

да. Если вы планируете установить Ubuntu наряду с существующей операционной системой, но у вас нет полной резервной копии Windows, перейдите в Settings > Update & security > Backup (Windows 10) и нажмите Go to Backup and Restore (Windows 7). Выберите Create a system image [Создать образ системы] и затем следуйте инструкциям по созданию

резервной копии на внешнем диске диска или сетевом хранилище (опция DVD не очень практична из-за непомерного количества необходимых DVD).

Обзаведясь образом системы, на том же экране создайте еще и диск восстановления системы — полученный загрузочный DVD вы сможете употреблять для восстановления образа вашей системы, если в процессе установки Ubuntu что-то пойдет наперекосяк. После отката вы сможете заново запустить процесс установки, не потеряв никаких данных.

Единственное, что вам понадобится — это пустой DVD или флешка на 2 ГБ (или больше), на которую вы запишете средства установки Ubuntu. Потом продолжите чтение данной статьи, чтобы выяснить, как настроить простую систему с двойной загрузкой Ubuntu/Windows.



**П**режде чем заняться самим процессом установки, выясните: а хватит ли на вашем диске места для Ubuntu? Ubuntu способен уместиться всего лишь на 5 ГБ, но тогда не останется места для приложений или файла подкачки разумного размера, так что в первую очередь надо определить, сколько свободного места на вашем диске с Windows.

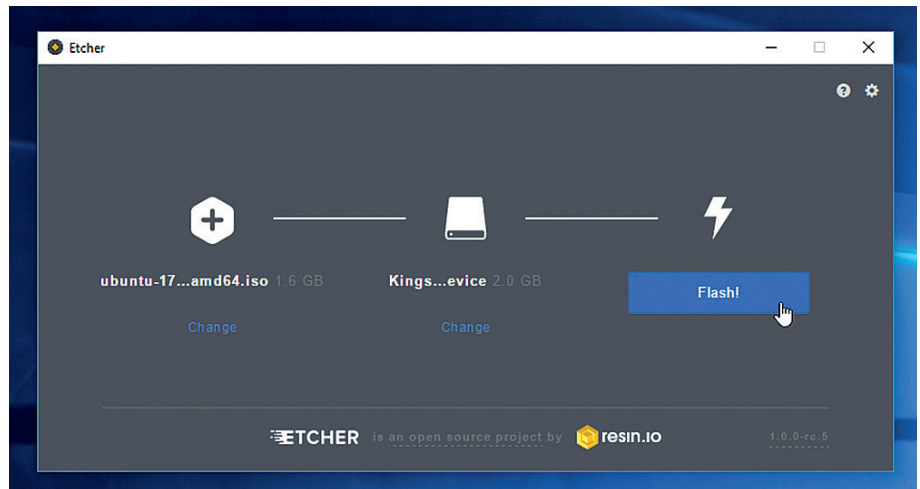
Если кто-то собирается просто порезвиться в мире Ubuntu, то он вполне обойдется разделом на 24 ГБ, который впоследствии можно нарастить. Но если вы планируете переход на постоянной основе, следует отвести намного больше места, особенно если вы собираетесь на вашем новом компьютере с Linux играть в игры.

Этот вопрос, вероятно, будет более серьезным для тех, у кого быстрый SSD-диск — обычно их емкость меньше. Если вы устанавливаете Ubuntu на настольный ПК и в его корпусе есть резерв (запасной отсек для диска вместе с сокетом SATA и кабелем питания), то обдумайте покупку отдельного диска для запуска Ubuntu с него. Вашей основной заботой будет обеспечить установку Ubuntu на правильный диск. Скоро мы вам покажем, как это делается.

## Загружаемые носители

Если вы еще не сделали это, скачайте установочное средство Ubuntu с: [www.ubuntu.com/download/desktop](http://www.ubuntu.com/download/desktop). Оно придет в виде 1,5-ГБ файла ISO, который вы сможете записать на DVD, нажав на него правой кнопкой мыши в Windows и выбрав Burn disc image [Прожечь образ диска]. Если у вас есть лишняя флешка на 2 ГБ (или больше), мы бы посоветовали использовать ее — это ускорит процесс установки. В ином случае, приготовьтесь к долгой загрузке и найдите экземпляр LXF225, на диске которого имеется Ubuntu 17.04.

Вы не можете «записать» образ диска напрямую на USB; вместо этого вам понадобится сторонний инструмент для создания загрузаемого USB-диска, и мы обнаружили, что проще всех в использовании — *Etcher* (<https://etcher.io>). После скачивания и установки запустите инструмент



» *Etcher* — простейший вариант с открытым кодом для создания загрузаемого через USB средства установки из ISO-файлов.

и нажмите на Select [Выбрать], чтобы выбрать свой файл Ubuntu ISO. Он должен автоматически выбрать вашу флешку (нажмите Change [Изменить], если это не так), так что нажмите Flash! [Залить!] и наблюдайте, как творится магия.

## Установка Ubuntu

В наши дни самые современные ПК заменили BIOS на нечто под названием UEFI. UEFI и сложнее, и мощнее, и хотя изначально это разрабатывалось для Windows, Ubuntu мигом адаптировалась к работе с такими функциями, как SecureBoot.

Однако одна функция, под названием FastBoot (также известна как FastStartup или QuickBoot), Ubuntu не поддерживает, так что вам придется перезагрузить свой ПК и найти возможность вызвать утилиту настройки. Обычно надо нажать клавишу F2 или Del. Перейдите в утилиту настройки UEFI и найдите опцию FastBoot — обычно в меню Boot или Windows — отключите ее при необходимости.

Теперь снова перезагрузитесь, подключив свое средство установки Ubuntu, и если оно не определилось автоматически (другими словами, если Windows загрузилась как обычно), то перезагрузитесь

опять, но найдите опцию входа в меню загрузки — например, по F11 или F12. Найдите свой DVD или флешку и выберите ее. Если она дважды появляется в списке, выберите опцию UEFI.

Когда появится основной экран, выберите опцию Install Ubuntu [Установить Ubuntu]. Когда вам это предложат, подключитесь к своей сети Wi-Fi, затем отметьте оба окна — для скачивания обновлений и установки сторонних программ — и нажмите Continue [Продолжить]. После небольшой паузы появится меню Installation Type [Тип установки], и тут начинается самая технарская часть установки.

Если вы намерены установить Ubuntu, то вам надо выбрать опцию Erase disk and install Ubuntu [Очистить диск и установить Ubuntu], но сначала обязательно ознакомьтесь со врезкой внизу. Если нужна двойная загрузка Ubuntu вместе с Windows, вам стоит взглянуть на опцию Install Ubuntu alongside Windows 10 [Установить Ubuntu вместе с Windows 10]. Если она помечена, оставьте ее выбранной и нажмите Continue.

Если же такой опции нет, значит, Ubuntu по какой-то причине не обнаруживает вашу установку »

## Установка Ubuntu на ПК

Если вы намерены стереть Windows с диска и установить себе исключительно Ubuntu, то процесс установки будет лобовым. Если на вашем диске нет ничего ценного, что вам хотелось бы сохранить, то просто сотрите весь диск целиком во время обычного процесса установки, выбрав Erase disk and install Ubuntu [Затереть диск и установить Ubuntu] на экране Installation type, и нажмите Continue.

Перед вами появится экран с сообщением, что вы собираетесь установить Ubuntu на выбранный диск (/dev/sda) и что будет использован весь

диск. Если у вас установлен второй жесткий диск, вы можете также разместить Ubuntu там — повторимся, этот диск будет очищен и полностью отведен под Ubuntu.

А если у вас множество разделов и отдельный том данных, и стерев Windows, вы бы хотели сохранить этот том нетронутым? Это не проблема: нажмите на ярлык Advanced partitioning tool [Расширенный инструмент деления на разделы] на экране Erase disk and install Ubuntu или выберите Something else [Другое] на предыдущем экране.

Здесь вам надо выбрать раздел Windows (определите его по названию и объему) и нажмите '-', чтобы удалить его. Пока вы здесь, мы рекомендуем также удалить все маленькие разделы в начале и в конце диска. Один из них обычно отмечен как Windows и относится к разделу загрузки.

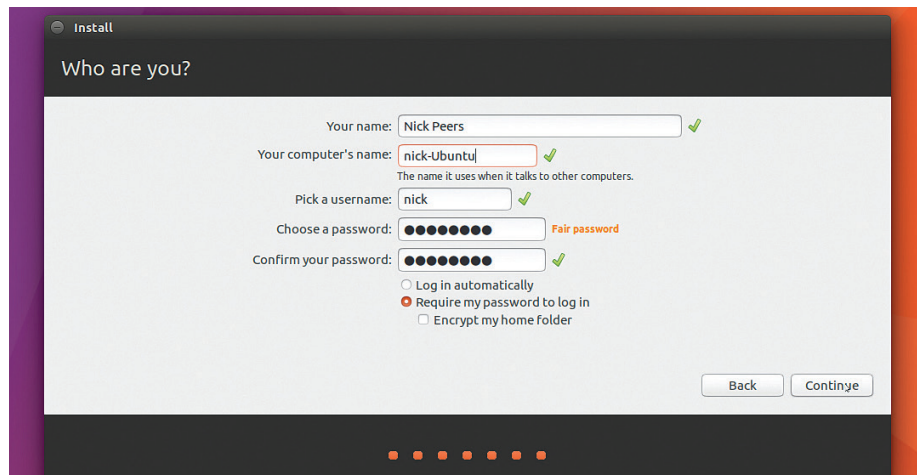
После этого просто выберите свободное место в начале диска и следуйте процессу шага 6 руководства для создания вашего раздела Ubuntu, затем нажмите Install Now, чтобы заменить Windows на Ubuntu.

Windows 10. Но повода для беспокойства нет — вам всего лишь придется чуть больше поработать. Пошаговое руководство (см. стр. 49) расскажет, каким образом вы можете вручную разбить диск на разделы и установить Ubuntu рядом с Windows.

Если опция по установке вместе с Windows появилась, то следующий экран должен графически отобразить ваш жесткий диск, где Ubuntu будет расположен справа — т.е. в конце диска. По умолчанию на него отводится 25 ГБ, если место позволяет, но это можно изменить нажатием и перетаскиванием границы двух разделов, чтобы увеличить или уменьшить объем.

Важный момент: если у вас есть второй жесткий диск, установленный в вашем ПК, возможно, Ubuntu выберет для установки именно его. Взгляните на выпадающее меню выбора диска, чтобы подтвердить (он отобразит пометки — sda, sdb и т.д. — и емкость в ГБ. Ваш диск Windows будет 'sda').

Это может испортить вам песню — например, окажется, что это более медленный жесткий диск или на нем меньше свободного места. Зато если вы установили второй диск специально для Ubuntu,



Ubuntu подведет вас к выбору сильного пароля для защиты вашей учетной записи.

то этот выбор вас порадует. В этом пункте можно попробовать нажать на выпадающее меню, чтобы выбрать другой диск, но в большинстве случаев sda будет недоступен. Тогда нажимайте Back —

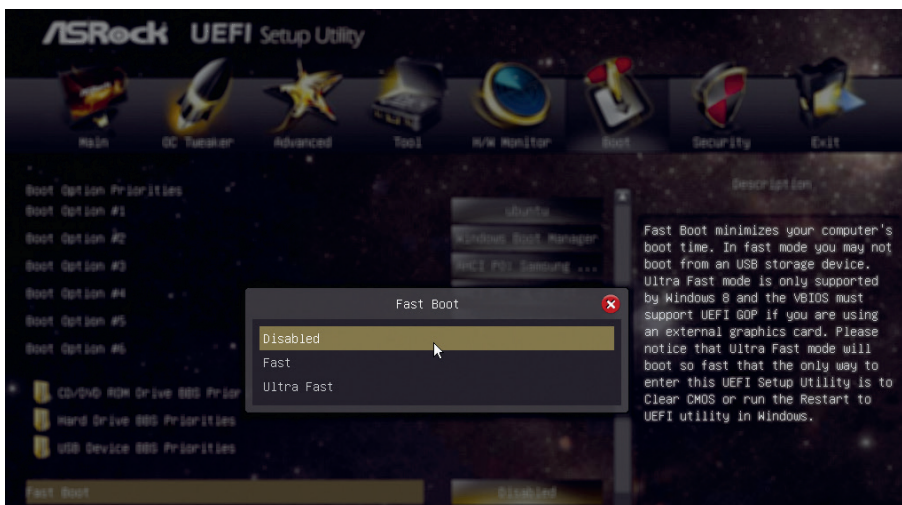
и следуйте пошаговому руководству, чтобы настроить всё вручную.

## Завершение установки

Если вы довольны выбором диска и изменили объем раздела в нужную вам сторону, нажмите Install Now > Continue. Просмотрите изменения и снова нажмите Continue. Оставшаяся часть установки идет обычным путем: вначале выберите свое местоположение, установите свой язык и раскладку клавиатуры. Затем введите свое имя — на его основе Ubuntu сгенерирует имя компьютера (для сетевой идентификации) и имя пользователя, но и то и другое вы можете при необходимости изменить. Вам также будет предложено установить пароль — не пропускайте это — и вы можете выбрать, войти ли прямо на рабочий стол или же проходить через приглашения при каждой загрузке Ubuntu. Нажмите Continue и предоставьте Ubuntu завершить свою работу.

В конце процесса нажмите Restart Now [Перезагрузить сейчас], удалите или отключите свой загрузочный носитель, когда предложат, и нажмите Enter. Если всё идет по плану, вы должны увидеть меню загрузки Grub с Ubuntu, выбранным по умолчанию, и видимым Windows 10. Это позволит вам при необходимости переключаться между операционными системами — если вы столкнетесь с проблемами загрузки, взгляните во врезку внизу про решение проблем. **LXF**

## Вы можете вручную разбить диск на разделы и установить Ubuntu рядом с Windows...



Отключите FastBoot в вашей утилите настройки UEFI, или Ubuntu не установится.

## Решение проблем с двойной загрузкой

При установке Ubuntu появляется менеджер загрузки под названием Grub, который должен обнаружить Windows и позволить вам переключаться между Ubuntu и Windows при каждом запуске. Иногда это не работает — особенно если вы принудительно разбили свой диск вручную — и Windows не обнаруживается. Вот что можно сделать, чтобы попытаться вернуть Windows.

Сперва загрузитесь с вашего средства восстановления Windows и выберите опцию починки ПК. Это вернет назад Windows, но удалит Grub. Затем

загрузитесь с вашего средства установки Linux и выберите Try Ubuntu, когда предложат. Теперь откройте окно терминала (нажмите Ctrl+Alt+T) и введите следующее:

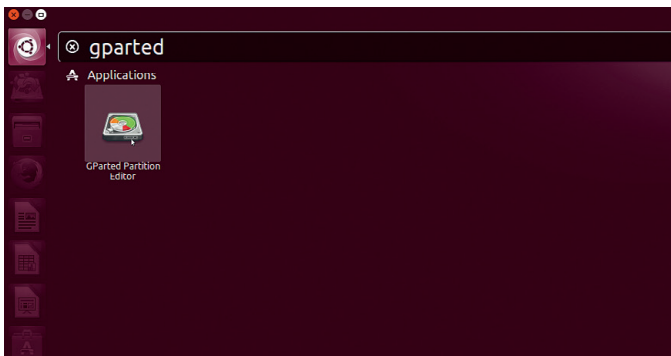
```
$ sudo add-apt-repository ppa:yannubuntu/boot-repair
$ sudo apt-get update
$ sudo apt-get install -y boot-repair && bootrepair
```

Затем откройте Boot Repair из Dash и выберите рекомендуемое восстановление. Процесс включает команды копирования (Ctrl+C) и вставки (нажатие правой кнопкой мыши и выбор Paste) в терминале.

Другие горячие клавиши, которые вам понадобятся — Tab и клавиши со стрелками для перемещения между опциями, пробел для выбора опций и Enter для продолжения.

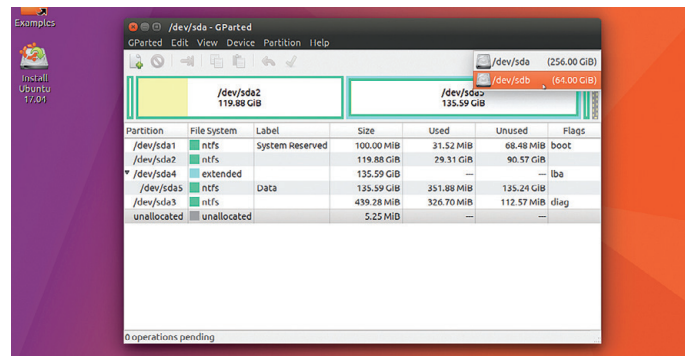
Теперь, после завершения, перезагрузка должна сработать. Если Grub есть, но опции Windows нет, возможно, он добавился в меню загрузки UEFI. Нажмите клавишу, чтобы перенести его в загрузку, и, возможно, вы обнаружите опции для переключения между Windows и Ubuntu, которые сможете использовать.

## Ручное разбиение жесткого диска на разделы



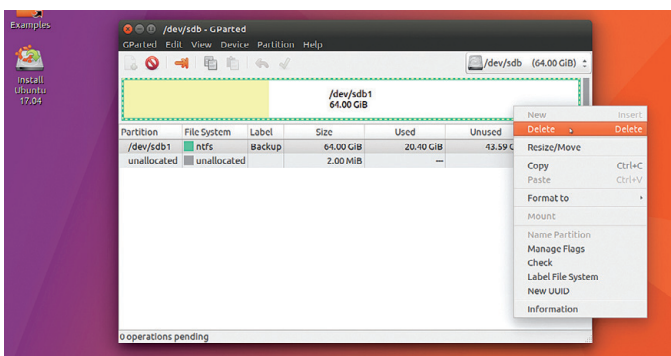
### 1 Загрузитесь на рабочий стол Ubuntu live

Загрузитесь с вашего средства установки Ubuntu и выберите Try Ubuntu, когда вам предложат, или же, если вы находитесь в меню Installation type во время установки Ubuntu, дважды нажмите Quit. В любом случае вы окажетесь на рабочем столе Ubuntu live. Нажмите на значок в верхней части Launcher слева, чтобы открыть Dash. Введите 'gparted', чтобы вызвать 'GParted Partition Editor', затем нажмите на него для запуска инструмента.



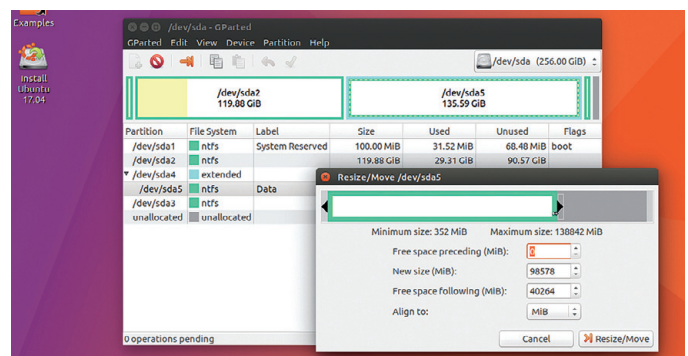
### 2 Выбор целевого диска

Для освобождения места на вашем жестком диске для Ubuntu мы используем GParted. По умолчанию в окне GParted должно быть выбрано /dev/sda — это диск, где стоит Windows. Огляните его объем и список разделов под ним, чтобы удостовериться: это именно тот диск, который вам нужен. Если вы собираетесь установить Ubuntu на другой диск, нажмите /dev/sda для просмотра других дисков, используя их объем и структуру разделов для определения должного диска.



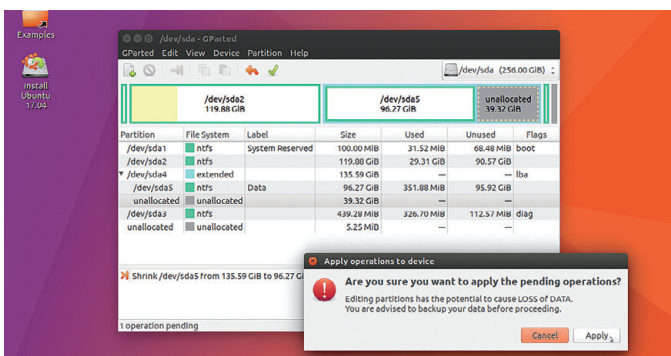
### 3 Подготовка диска

Если вы устанавливаете Ubuntu на отведенный для него диск, то он должен отображаться пустым (или «неотведенным»), если вы только что купили его и пока не задействовали. Если вы используете старый диск, существующие разделы, скорее всего, будут видны. Подтвердите, что это тот самый диск и вам не нужны никакие данные на нем, затем по очереди щелкните правой кнопкой на всех видимых разделах и выберите Delete для их удаления. После этого переходите к шагу 5.



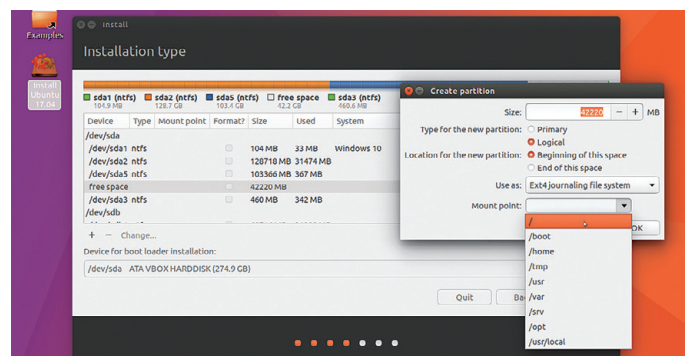
### 4 Освобождаем место для Ubuntu

Если вы ставите Ubuntu на свой диск Windows, посмотрите на структуру раздела. Если у вас один большой раздел, нажмите на нем правой кнопкой и выберите Resize/Move; в случае нескольких больших разделов нажмите правой кнопкой на крайний правый, чтобы изменить его объем, если он достаточно велик. Нажмите и перетащите правый ползунок влево, чтобы освободить место для Ubuntu, оставив как минимум 10 ГБ для существующего раздела. Нажмите на кнопку Resize/Move.



### 5 Примените изменения и установите Ubuntu

Нажмите на кнопку с зеленой галочкой. Если вас всё устраивает, нажмите Apply и подождите, пока закончится разбиение диска на разделы. После этого нажмите Close, затем выйдите из GParted и дважды нажмите Install Ubuntu 17.04. Следуйте инструкциям мастера установки, как указано в основном тексте на странице Installation type, затем выберите Something else, чтобы вручную выбрать, куда установить Ubuntu. Нажмите Continue.



### 6 Настройте раздел Linux

Выберите место для установки Ubuntu (скажем, под /dev/sda) и нажмите +. Оставьте размер, но создайте новый раздел Logical [Логический]. Убедитесь, что он установлен на beginning of this space [Начало этого места] и меню Use as указывает на 'Ext4 journaling file system'. В меню Mount point [Точка монтирования] выберите /, затем нажмите OK. Проверьте, что диск, где стоит Windows, выбран для Device for boot loader installation [Устройство для установки загрузчика], и нажмите Install Now.

# Сказание о стартапе

Для Сергея Ерёменко успех измеряется не деньгами, а приобретенным жизненным опытом.



**У** каждого из нас карьерный путь складывается по-разному. Но, думаю, многим из нас хотелось бы прийти к тому, чтобы успешно работать на самого себя и не зависеть от решений вышестоящих руководителей. Эта статья — пример того, как группа людей объединилась ради общей цели и постаралась прийти к такому итогу.

## Идея

Идея была новаторской: создать первый в мире информационный сервис с интерактивной картой загруженности различных заведений людьми в режиме реального времени. Таким образом, каждый пользователь мог бы получать точную информацию о количестве свободных мест и заполняемости любимых заведений. При этом пользователям предоставлялась возможность добавлять на интерактивную карту мира точки интереса и относящиеся к ним фотографии и видео; писать комментарии к уже добавленным местам; приглашать друзей на какое-либо мероприятие, в целом вести с ними переписку. Другими словами, в наличии было большое количество элементов популярных социальных сетей (рис. 1).

На тот момент аналогов действительно не существовало, что давало возможность плотно занять создаваемую нишу. После регистрации на сайте вы попадали на интерактивную карту, на которой можно было увидеть множество меток разных цветов. Метки делились по категориям в зависимости от типа заведения, цвета же

показывали его загруженность на данный период времени. Существовало 5 различных цветов, что потом также легло в основу логотипа компании (см. рядом с заголовком статьи).

По щелчку на конкретной метке можно было увидеть сводку данных по заведению, а также перейти на страницу с более подробной информацией, включающей грядущие мероприятия, акции и так далее.

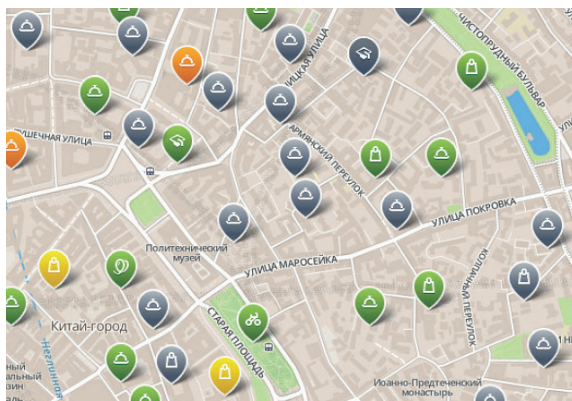
Для сбора информации о загруженности мы использовали системы подсчета посетителей на основе алгоритмов компьютерного зрения. Данный подход к разработке подобных устройств постепенно набирает обороты. Это связано со многими факторами — например, с возможностью расширения заложенной изначально функциональности, наличия аналитики. Допустим, если изначально система была рассчитана на подсчет всех посетителей без разбору, то в дальнейшем можно было бы добавить возможность не считать детей, поскольку они ведь не являются потенциальными клиентами.

## Сбор команды и организация рабочего процесса

Поиск заинтересованных в участии людей велся в основном в Москве и занял очень длительное время — около полугода. Почему так долго? Этому способствовало несколько факторов:

- » Выбранный подход к финансированию — всё делать самостоятельно за свои деньги, без привлечения инвесторов и венчурных фондов.
- » Сложность финансовой мотивации кандидатов. Зарплата не предусматривалась. Желающим присоединиться к делу предлагались условия процентной части от чистого дохода в будущем, в зависимости от диапазона решаемых задач. Всё оборудование и компоненты покупались за собственные деньги идеологов, у которых, конечно, был наибольший процент от чистой прибыли.
- » Наличие достаточного количества времени. Поскольку сделать предстояло многое, требовалась большая отдача. Фактически просто за идею далеко не каждый соглашался присоединиться и даже встретиться для разговора.

Хоть это и заняло длительное время, но в итоге у основателей получилось собрать целеустремленную команду, готовую к достижению поставленных целей, несмотря на наличие большого



» Рис. 1. Карта с нанесенными заведениями различных типов.

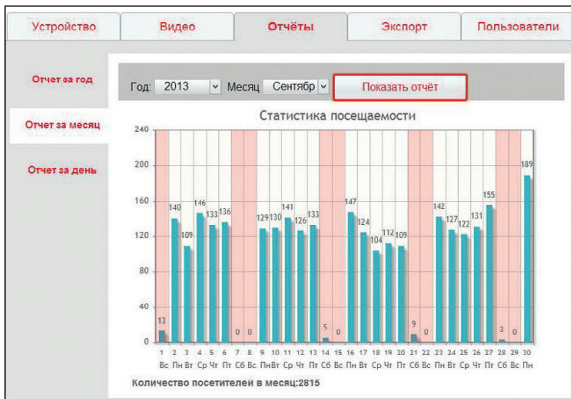


Рис. 2. Внешний вид окна пользовательского видеосистемы подсчета посетителей.

количества трудностей. Многие обладали большим опытом в определенной области. Имелась даже пара ребят web-разработчиков, работающих удаленно! Благодаря этому удалось с наименьшими потерями обойти многие сложности и нюансы. Когда было необходимо собраться и обсудить дальнейшие планы с достигнутыми результатами, арендовали в специальных местах офисные комнаты. Рисовали графики, дискутировали, тестировали функциональность сайта и не только. Для постановки задач использовали бесплатные инструменты — комплект «Битрикс24».

Меня всё это привлекло, и после обсуждения условий я с радостью согласился. Вступил в команду в качестве технического специалиста, диапазон задач которого был от организации процесса сбора информации о загруженности до общения с руководством заведения по соответствующим вопросам. При наличии у меня основного места работы это было хорошей возможностью сменить обстановку, отвлечься от повседневных задач.

Поскольку раньше я активно участвовал в разработке системы подсчета, как раз подходящей под наши требования, то выбор варианта был очевиден. Решение обладало собственным web-интерфейсом, позволяющим настраивать некоторые параметры алгоритмов; анализировать графики проходов за конкретный промежуток времени; выгружать отчеты в форматах CSV и XML (рис. 2).

Оставалось решить одну проблему. Поскольку продукт хранит все получаемые данные на встроенной памяти, необходимо было придумать алгоритм сбора данных с систем (одной или нескольких) и передачи их на наш сервер для дальнейшего отображения на интерактивной карте. Выбранная система подсчета посетителей предоставляла достаточно дружелюбный API, который позволял добыть всю необходимую информацию. Оставалось только создать решение, которое бы выполняло функцию агрегатора информации и промежуточного пункта между конкретными объектами и сервером.

## Создание собственного решения

Таким решением стало добавление устройства на базе операционной системы GNU/Linux Debian, с комплектом собственного программного обеспечения и пользовательского интерфейса на борту. В качестве аппаратной платформы мною вначале был выбран небезызвестный Raspberry Pi, версия 2B. Выбранный для установки дистрибутив назывался Raspbian, поскольку он активно поддерживается создателями устройства. Но после некоторых опытов выяснилось, что на тот период в версии операционной системы имелись проблемы с сетевой службой, в результате чего периодически пропадало сетевое соединение. В нашем случае это являлось недопустимым. Как следствие, было найдено более подходящее решение — тонкий клиент IP-Plug. Выбранная конфигурация включала всего два USB 2.0 и два сетевых разъема. В качестве ОС выступил один из моих любимых дистрибутивов — Debian.

- Характеристики тонкого клиента скромные:
- » микропроцессор Marvell Kirkwood 1 ГГц;
- » оперативная память 1024 МБ DDR2;
- » 1 ГБ NAND Flash.

Большим плюсом являлось наличие разъема UART. Данный интерфейс позволяет с помощью пары переходников легко подключиться к устройству через эмулятор терминала, поскольку в итоге на хост-системе мы получим последовательный порт (COM-порт). Из-за относительно скромных характеристик данный мини-сервер является одним из самых дешевых на рынке этого сегмента. Но для наших целей этого было вполне достаточно. Новый продукт получил название *CatchClient*.

Вся разработка велась на моих основных языках программирования — С и С++. Взаимодействие с системой подсчета осуществлялось с помощью запросов GET и POST. Для этих целей идеально подошла библиотека *libcurl*. Она развивается уже длительное время и является одной из самых популярных библиотек для взаимодействия с различными серверами, которые функционируют по протоколам с синтаксисом URL. Для параллельного взаимодействия со всеми имеющимися на объекте системами подсчета пригодился стандарт реализации потоков выполнения POSIX. Варианты данного API существуют для большого числа операционных систем — и, конечно, для GNU/Linux. На каждую систему подсчета, находящуюся в одной локальной сети с разработанным решением, создавался отдельный поток, в котором с помощью API происходила аутентификация, получение и отправка данных

## Достаточно дружелюбный API позволял добыть всю необходимую информацию.

на специальный PHP-скрипт на стороне сервера. Получаемые через определенный промежуток данные необходимо было локально где-то хранить. Для этих целей подошла встраиваемая реляционная база данных *SQLite*, в силу своей простоты и удобства. В дальнейшем вместе с коллегами по команде был создан дружелюбный интерфейс, позволяющий получать отчеты и выгружать их за конкретный промежуток времени, а также графически отображать имеющиеся данные. Это было полезно, например, для анализа активности посетителей, что в дальнейшем можно было бы применить для оптимизации бизнес-процессов заведения. Для написания интерфейса была использована популярная связка PHP+JavaScript, элементы CSS, а для требующей привилегий администратора функциональности — интерпретатор *Bash*. Опять пригодился опыт предыдущего места работы, где похожие задачи мы делали командой всего в несколько человек. В целом функциональность интерфейса была похожа на имеющуюся в видеосистеме подсчета, но у нас имелась дополнительная вкладка для

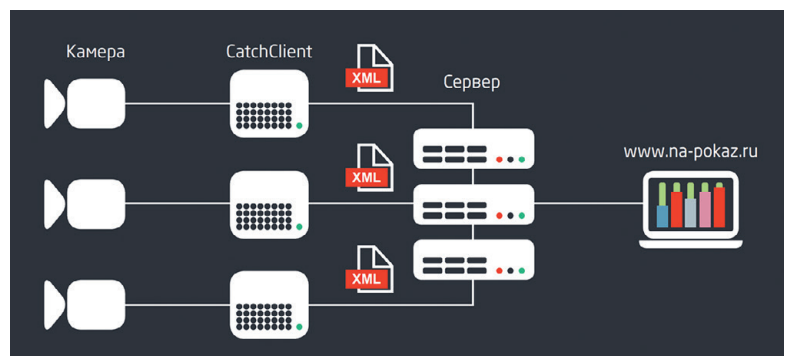
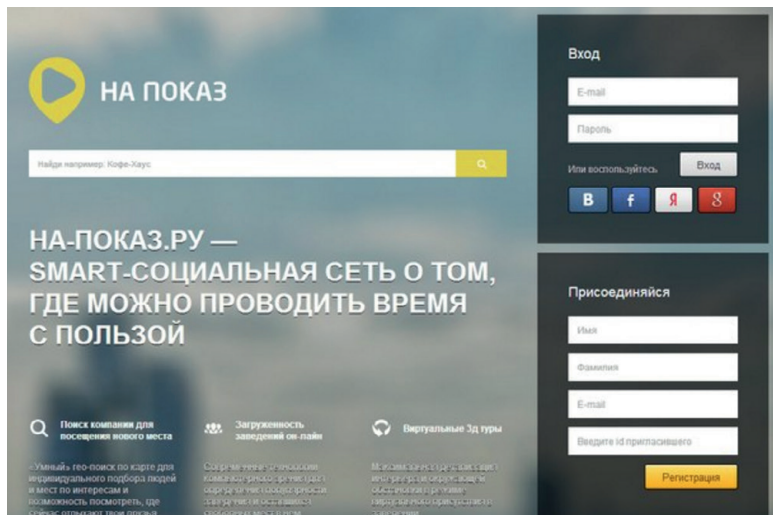


Рис. 3. Схема взаимодействия системы подсчета с сервером.



➤ Рис. 4. Стартовая страница сайта.

добавления/удаления/редактирования этих самых систем. В качестве web-сервера был выбран HTTP-сервер *Apache*, благодаря которому всё это «хозяйство» благополучно работало (рис. 3).

## Развитие событий

Параллельно с разработкой и тестированием решения для сбора и обработки данных шла аналогичная работа для сайта проекта. Работы по сайту были проделаны поистине титанические, но в итоге в один прекрасный день наконец завершились запуском его «боевой» версии (рис. 4).

Через некоторое время после старта «боевой» версии сайта у нас получилось договориться о первой установке комплекта систем на объект. Устанавливали долго, выявляя по ходу разные тон-

## Полученный опыт помогает на карьерном пути, подталкивает к реализации идей.

кости и устраняя проблемы. В итоге всё прошло успешно, данные поступали на сервер и отображались, как задумывалось. После успешного старта начала происходить целая череда интересных событий.

Поскольку в команде были профессиональные фотографы и видеооператоры, с некоторыми заведениями получилось договориться о создании первого совместного репортажа о заведении.

Вели переговоры с другими стартапами о возможных схемах взаимодействия. Участвовали в различных конкурсах. На бизнес-форуме "Start2Day", направленном на обзор, обсуждение и информационную поддержку начинающих предпринимателей, была нами подготовлена и показана презентация проекта. Рассказывали о преимуществах и возможностях сервиса, идеях по дальнейшему

развитию. Таким образом, получилось первый раз публично заявить о себе широкому кругу людей. Потом случился настоящий прорыв — мы стали одним из трех финалистов отборочного тура, организованного одной российской медийной интернет-площадкой. Это позволило презентовать наше детище на крупном международном конкурсе инвесторов и стартапов "LeWeb", проходящем в Париже (рис. 5).

Для команд вроде нашей это была возможность представить свой проект зарубежной публике. В целом он произвел на участников и организаторов положительное впечатление. По рассказу одного из заинтересовавшихся: «Мне нравится ходить в спортзал в то время, пока там мало занимающихся. Ваш проект позволил бы мне выявить ближайший спортзал, где мне было бы комфортно заниматься на текущий период времени». Далее получилось выйти в финал конкурса «Инновации для города». Цель данного конкурса — выявить наиболее перспективные проекты, наиболее талантливых предпринимателей, идеи которых могут улучшить жизнь жителей столицы. На этом конкурсе мы преподносили себя как сервис, помогающий решать одну из самых острых проблем городского жителя — очереди, большое скопление людей в мегаполисе. Цель проекта — помочь людям экономить свое время и деньги, выбирать маршруты посещения того или иного места, зная заранее, какое количество людей там находится, есть ли там очередь, остались ли свободные места. В финале же — заняли призовое место; в качестве награды организаторы обязались предоставить бухгалтерскую поддержку развивающемуся проекту на длительный период времени.

По окончании конкурса даже получилось организовать вечеринку в одном из столичных клубов.

Поскольку все затраты на развертывание решений на объектах покрывались собственными силами команды, процесс развития шел довольно медленно. В итоге пришли к тому, что без инвестора никак не обойтись. К сожалению, отношение инвесторов и различных инвестиционных фондов к российским стартапам довольно скептическое. Связано это с частым желанием команд по-быстрому подписать контракты, разделить деньги и разбежаться.

После продолжительных поисков нашлись заинтересованные лица. Длительное время к нам присматривались, следили за работой сайта, изучали риски. В итоге, как в дальнейшем рассказывал один из идеологов проекта, инвестор практически согласился обеспечить нас финансовой поддержкой на целых 5 лет. Но по стечению обстоятельств в нашей стране начались резкие финансовые изменения. В результате инвестор после недолгих раздумий передумал подписывать контракт. Это нас, конечно, очень подавило. Некоторое время мы пытались добиться договоров о сотрудничестве с хозяевами других объектов, один из которых должен был обеспечить нам на некоторое время спокойное развитие. К сожалению, руководство — видимо, всё из-за тех же финансовых изменений — пересмотрело свои планы и не захотело сотрудничать.

## Заключение

После серии ударов проект стал потихоньку затухать. Энтузиазм команды постепенно иссякал. Начались серьезные разногласия по поводу дальнейшего вектора действий. В итоге всё сошло на нет.

Несмотря на то, что нашей командой не было заработано ни рубля прибыли, мне и коллегам удалось получить огромный опыт. Для меня, в частности, это был опыт написания целого комплекса программ, составления проектов для крупных объектов, углубления в специфику работы систем подсчета посетителей, совместного преодоления трудностей. До сих пор полученный опыт помогает мне на карьерном пути, подталкивает к реализации появляющихся идей, позволяет легче находить решение в спорных ситуациях и, наконец, уверенней идти вперед к поставленным целям. Поэтому я считаю, что данный жизненный этап был настоящей находкой! LXF



➤ Рис. 5. Логотип конкурса "LeWeb".

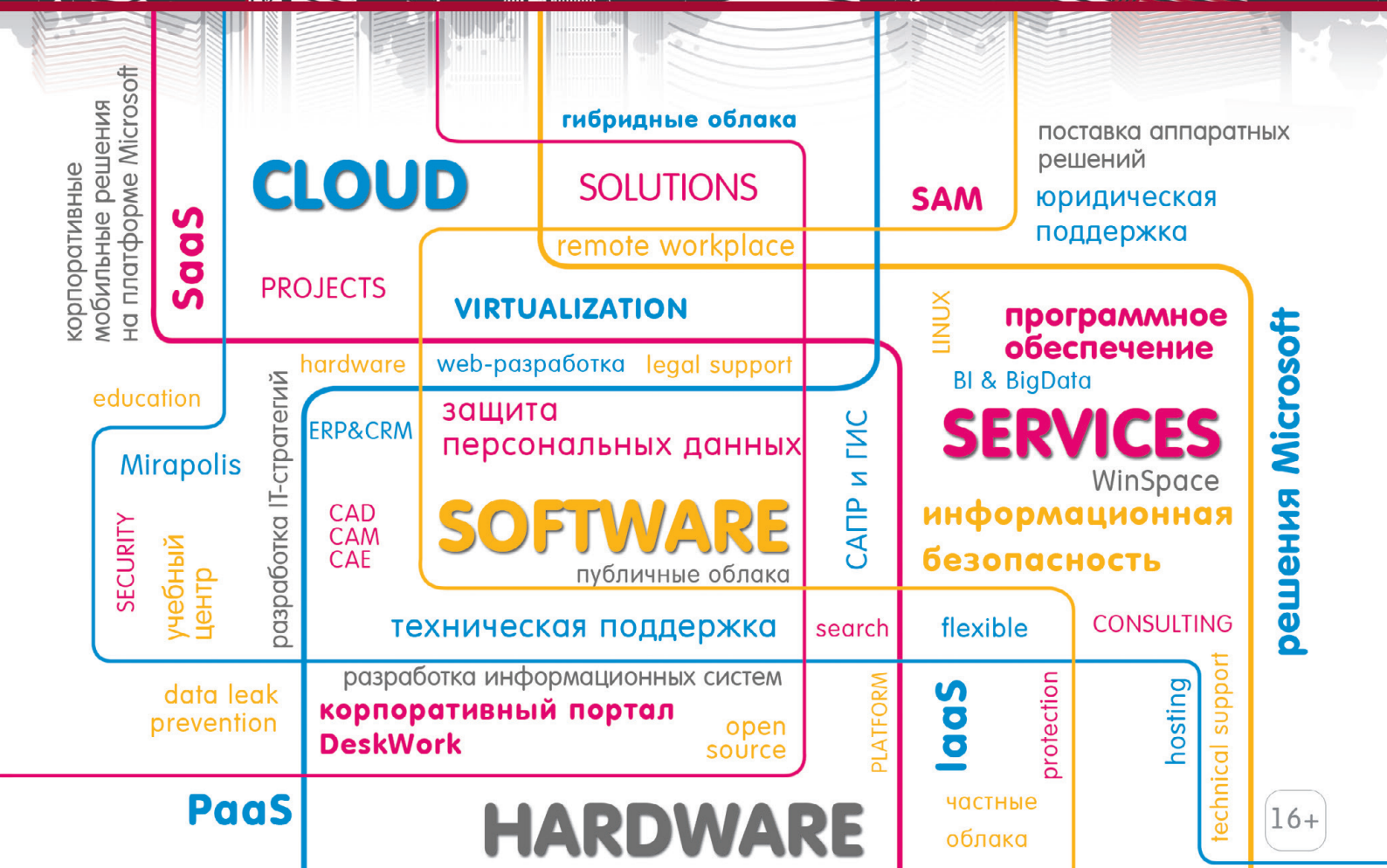
# softline®

Cloud Software Hardware Services

# 20+

Years in IT

## IT-архитектура вашего бизнеса



# 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.

## Напоминай о себе

Если не попытаюсь, то и не узнаю.  
*Луффи. One Piece. Эпизод 49*

**Ч**тобы люди ни делали, они всегда делятся на меньшинство и большинство.

Следует осознавать, что число пользователей GNU/Linux пока недостаточно, чтобы осознать себя большинством. Выходит, надо сложить ручки и затихнуть? Например, чтобы не дать повод для обструкции со стороны большинства? Вопрос риторический: отнюдь нет. Более того, есть надежда, что мы — то меньшинство, которое творит будущее. Только оставьте след, в коде или тексте!

Кореевед Андрей Ланьков подметил разницу в отношении к японской оккупации в первой половине XX в. у жителей Тайваня и Южной Кореи. Первые хоть и не ощущают особого пиетета, но вполне способны прикрепить на историческое здание табличку с именем японского архитектора, а у вторых от такого кощунства в буквальном смысле пена может пойти изо рта. Вроде и странно, ведь подавляющему большинству в оккупации жилось примерно одинаково поганно; но, в отличие от элиты Кореи, грамотную элиту Тайваня не «гнобили», и в исторической перспективе это оказалось важным. Неграмотное большинство книг не писало, и память о себе оставили одни грамотеи!

Не стесняйтесь что-либо написать или заявить. Пока вы молчите, никто о вас ничего не знает, но если все линуксоиды заговорят, то остальное молчаливое большинство можно смело игнорировать в угоду суперактивному меньшинству.  
[E.m.Baldin@inp.nsk.su](mailto:E.m.Baldin@inp.nsk.su)



## В этом месяце вы научитесь...

### Командовать медиа ..... 56 Оживлять картинки ..... 58

**Никю Пирсу** под силу воспроизведение мультимедиа прямо из командной строки. Да-да, и музыки, и видео — а иногда даже видео Blu-ray.

В прошлом номере **Матс-Таге Аксельссон** выбирал лучший инструмент анимации, а теперь настала пора употребить этот инструмент в деле.

### Перенаправлять пакеты ..... 62 Делать инъекции ..... 64

В компании **Эрмина Крепоника** препарируем атаки типа «человек-посредник». Вывод такой: главное — пользоваться HTTPS и HSTS.

О нет!!! **Нейт Дрейк** этим не занимается: он всего лишь покажет, как легко ввести вредоносный код на ваш ПК, вставив якобы USB-флешку. Будьте бдительны!

### Исследовать ядро ..... 66 Настраивать среду ..... 68

Что, собственно, делает ядро и как оно это делает? **Джонни Бидвелл** подробно описывает состав и функциональность самого сердца Linux.

**Джонни Бидвелл** презирает устаревшие подходы и пересматривает для вас множество рабочих столов, достойных XXI века.

### Спасать данные ..... 72 Управлять лежа ..... 76

Специально для этого **Марк Чизхолм** собрал отдельный дистрибутив на базе Arch. Хотя и тестировать на вторжение тоже получится.

Послушным Домом **Максима Черепанова** можно поругать прямо с дивана, взяв удаленный пульт от какого-нибудь поломанного бытового прибора.

### Интерактивно редактировать ..... 80

**Дмитрию Кузнецову** команды *sed* не кажутся китайской грамотой. Вы удивитесь, но с его подачи они и вам сделаются понятны.

## АКАДЕМИЯ КОДИНГА



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

### Ну-ка, отгадай ..... 84 Раскрасим число π ..... 88

Бросая вызов читателям, **Михалис Цукалос** затевает серию (как он надеется) головоломок — а вы попробуйте-ка его победить.

Полиглот **Михалис Цукалос** отобразит любимое математиками число в виде цветной картинки, на разных языках программирования.

# Terminal: Звук и Вид — Правим

Ник Пирс обнаружил, что в терминале можно делать больше, чем просто воспроизводить коллекцию файлов видео, музыки и изображений.



Наш эксперт

У Ника Пирса нездоровое пристрастие к цифровым медиа, чему не помогает его решимость утонуть в море DVD, Blu-ray и CD, собирая полную коллекцию таких.

Мультимедиа играет ключевую роль в вычислительной технике, и вам не обязательно заходить на рабочий стол, чтобы наслаждаться своей коллекцией музыкальных и видеофайлов и фотографий: командная строка тоже обеспечивает доступ ко множеству инструментов, которые позволяют копировать, кодировать, конвертировать и изменять ваши медиа-файлы.

Для начала обеспечим в Ubuntu полную поддержку мультимедиа. Даже если вы во время установки отметили поле для привлечения стороннего ПО, вам всё-таки не хватает нескольких ключевых элементов. Во-первых, давайте установим все дополнения "restricted", а не только те, что добавлены на этапе установки. Откройте терминал и введите следующее:

```
$ sudo apt-get install ubuntu-restricted-extras
```

По этой команде установится несколько дополнительных пакетов, а пакет *libavcodec-ffmpeg56* заменится на *libavcodec-ffmpeg-extra56*, в нем более широкий выбор плагинов *GStreamer* (см. <https://gstreamer.freedesktop.org>) для потоковой передачи мультимедиа, воспроизведения, редактирования и обработки. Вам будет предложено установить шрифты Microsoft TrueType — нажмите Tab, затем Enter, а затем выберите Yes с помощью клавиш курсора, после чего снова нажмите Enter.

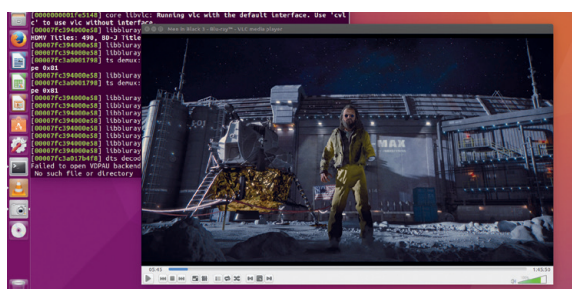
Поддержку воспроизведения зашифрованных DVD-дисков добавим командой

```
$ sudo apt-get install libdvd-pkg
```

После установки наберите

```
$ sudo dpkg-reconfigure libdvd-pkg
```

На этом процесс завершен. Если вы хотите воспроизводить Blu-ray на своем ПК, поставьте галочку (см. врезку вверху справа на стр. 57).



» Удобнее всего смотреть старые фильмы Blu-ray в VLC — запустите его из командной строки.

## Проигрывание мультимедиа

Для воспроизведения мультимедиа на ПК вариантов множество, и очевидный выбор — VLC, особенно если вы хотите воспроизводить коммерческие диски. Установите его с помощью `sudo apt-get install vlc`, и у вас появится возможность запускать его с рабочего стола или прямо из командной строки, при этом на рабочем столе откроется окно воспроизведения мультимедиа. VLC полностью управляется из командной строки — наберите `vlc --help` для получения списка команд, а для обычного воспроизведения достаточно набрать `vlc` и следом — источник с файлом мультимедиа (`file://[path/file]`), DVD (`dvd://[device]`) или Blu-ray (`bluray://[device]`), например:

```
$ vlc dvd:///dev/sr0
```

Вы также можете передавать медиа через Интернет (`http://host[:port]/file`, при необходимости замените `http` на `ftp` или `MMS`) и через потоковый сервер UDP:

```
$ vlc udp://[<source address>]@[<bind address>]:<bind port>]]
```

## Быстрое копирование CD

Если вы ищете способ быстро скопировать [rip] компакт-диски на ПК из командной строки, попробуйте *Abcde* (`sudo apt install abcde`). По сути это скрипт, связывающий другие инструменты — программное обеспечение для копирования компакт-дисков, кодировщики, средства идентификации CD и маркировки и т. д. — чтобы обеспечить решение «всё-в-одном» для копирования компакт-дисков с минимумом усилий.

После установки (выберите No configuration, если во время настройки вам это предложат), вы готовы к работе со следующей базовой командой:

```
$ abcde -o mp3
```

Другие варианты вывода включают *vorbis* (для OGG), *FLAC*, *m4a* и *WAV* по умолчанию. Вам могут предложить установить дополнительное программное обеспечение (например, *Abcde* требует, чтобы *eyeD3* отмаркировал файлы MP3).

После запуска CD сопоставляется с базой данных *cddb*. Вам предлагают вручную отредактировать эту информацию в текстовом редакторе (вариант *nano* — самый простой), и перед началом копирования спросят, один или несколько исполнителей записаны на компакт-диске.

Если вы хотите настроить определенные параметры по умолчанию, которые позволили бы копировать компакт-диски, просто вставив диск в дисковод и набрав *abcde*, следует создать файл конфигурации:

```
$ sudo nano ~/.abcde.conf
```

По этой команде создастся пустой текстовый файл. Чтобы его заполнить, посетите [www.andrews-corner.org/linux/abcde](http://www.andrews-corner.org/linux/abcde), где вы обнаружите скрипты, которые можно скопировать и вставить в этот файл в зависимости от желательного формата получаемых копий.

## Музыка командной строки

Если вам нужен музыкальный проигрыватель с интерфейсом командной строки (идеально подходит для удаленного доступа или использования за пределами рабочего стола), есть несколько вариантов. Начнем с инструмента для воспроизведения широкого спектра музыкальных форматов (FLAC, OGG, MP3 и т.д.) непосредственно из командной строки под названием *SoX*:

```
$ sudo apt install sox libsox-fmt-all
```

Библиотека *libsox-fmt-all* включает поддержку проприетарных форматов, таких как MP3. После установки перейдите в папку с вашей музыкой и проиграйте ее командой *play*:

```
$ play *.mp3
```

С помощью *Ctrl+C* перейдите к следующему треку или щелкните по треку дважды, чтобы остановить воспроизведение. Введите *play --help* для просмотра полного списка доступных параметров, включая широкий спектр эффектов и фильтров, таких как

```
$ play *.flac bass +2 gain +1 reverb
```

Если вам нужен более интерактивный плеер, попробуйте *cmus* (`sudo apt install cmus`). После запуска после короткой паузы появится окно с двумя панелями. Приложение управляется аналогично текстовому редактору *Vi* — например, чтобы загрузить свою папку **Music** в программу, введите

```
:add ~/Music
```

Затем можно просмотреть дорожки и плей-листы — введите *map cmus*, чтобы появилось подробное руководство пользователя. Можно также подключить дополнительные функции посредством скриптов расширения — см. <https://github.com/cmusc/cmusc/wiki> для получения более подробной информации.

## Мощь FFmpeg

Всё, естественно, усложняется, когда вы хотите конвертировать аудио и видео, независимо от пар форматов (например, AVI в MP4), записать прямую трансляцию или проделать нечто более сложное, например, встраивание субтитров в файл. Одним из известнейших инструментов для работы является *FFmpeg* (`sudo apt install ffmpeg`). Он распознает широкий диапазон форматов и заодно включает инструменты (*ffserver*, *ffplay* и *ffprobe*) для транскодирования, потоковой передачи и воспроизведения медиа.

Мы собираемся поверхностно ознакомиться с некоторыми основными применениями его в реальном мире. Сначала — конвертирование файла из формата MKV в MP4:

```
$ ffmpeg -i inputfile.mkv -vcodec copy -acodec copy outputfile.mp4
```

Можно пойти дальше и быстро конвертировать (или, по-другому, ремультимплексировать) папку с файлами:

```
$ for i in *mkv; do ffmpeg -i "$i" -vcodec copy -acodec copy "$i.mp4"; done
```

Этот простой пример изменяет тип файла (также известный как контейнер), но не меняет его кодеки (аудио- и видеоконтент). Работает почти мгновенно, потому что нет декодирования, конвертирования, а затем повторного кодирования видео или аудиопотоков: это просто переключение с одного кодека на другой.

Если вы хотите конвертировать кодек видео, а также его контейнер (введите *ffmpeg -codecs*, чтобы получить полный список поддерживаемых кодеков), попробуйте, будет ли работать следующий синтаксис:

```
$ ffmpeg -i inputfile.wmv -vcodec h264 -acodec aac outputfile.mkv
```

Вообще-то вы, вероятно, получите строку ошибки “too many channels [слишком много каналов]”, указывающую на наличие нескольких видео- и/или аудиопотоков, но *FFmpeg*, по крайней мере, попытается выбрать правильные (процесс, называемый потоковым отображением). Однако следующая ошибка будет убийственной — поскольку с вас потребуют указать определенные параметры кодировщика, выбранного для *-vcodec*.

Эти параметры можно определить, набрав команду

```
$ ffmpeg -h encoder=h264
```

## Добавьте поддержку Blu-ray

В отличие от DVD, воспроизведение Blu-ray в Linux более проблематично (см. **LXF223**) из-за наличия двух разных форм шифрования. Старые диски, закодированные с использованием шифрования AACs, с большой вероятностью будут работать с данным методом; некоторые диски, зашифрованные BD+, могут работать, если повезет... короче, это дело случая. Следующие команды устанавливают необходимые библиотеки вместе с *VLC Media Player*, который понадобится для просмотра Blu-ray:

```
$ sudo apt-get install vlc libaacs0 libbluray-bdj libbluray1
```

```
$ mkdir -p ~/.config/aacs/
```

```
$ cd ~/.config/aacs/ && wget http://vcl-bluray.whoknowsmy.name/files/KEYDB.cfg
```

После установки вставьте Blu-ray в дисковод и запустите *VLC* из командной строки:

```
$ vlc bluray:///dev/sr0
```

Замените */dev/sr0* на путь к своему приводу Blu-ray, и если диск поддерживается, он должен автоматически запустить основную контент.

Выведется длинный сложный список — руководство по кодированию в обычные форматы, такие как H.264, VP8/9 и AAC, прочитайте раздел *Encoding* на <https://trac.ffmpeg.org/wiki>.

## Другие инструменты

Давайте закончим обзором некоторых специфических инструментов. Сначала — *ffprobe*, который устанавливается вместе с *FFmpeg*. Это удобный способ просмотра метаданных, прикрепленных к конкретному файлу мультимедиа:

```
$ ffprobe file.mp3
```

Затем — *normalize*, инструмент, который, по сути, гарантирует, что все ваши музыкальные треки имеют стандартный уровень громкости. Это идеально подходит для плей-листов, когда вы не хотите постоянно дергать регулятор громкости вверх-вниз:

```
$ sudo apt install normalize-audio
```

Теперь перейдите в папку с файлами и введите

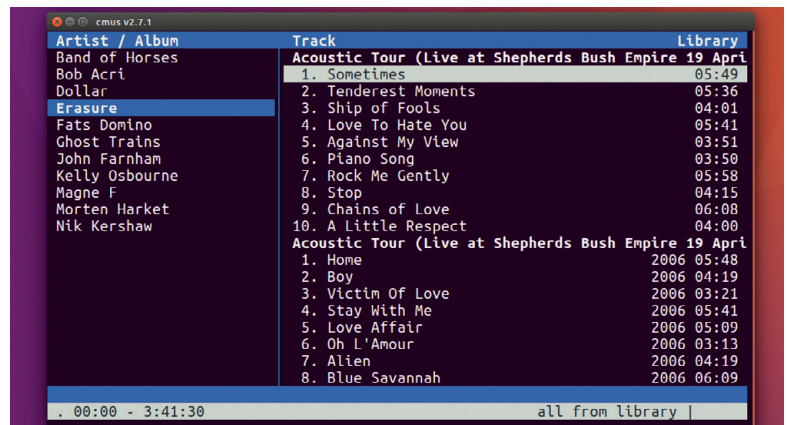
```
$ normalize-audio -bv *.mp3
```

Отсканируются все файлы в каталоге, с целью вычислить стандартное отклонение между дорожками (экстремальные отклонения игнорируются), затем применяется корректировка ко всем трекам, в том числе с экстремальными отклонениями, чтобы привести их в соответствие друг с другом. Если вам не нужен подробный вывод, вместо *-bv* используйте *-b*.

Наконец, *exiftool* (`sudo apt install libimage-exiftool-perl`) отображает метаданные для ряда медиа-форматов, а также способен записывать метаданные во многие популярные форматы изображений, включая JPG, TIF, GIF и PNG. Посетите <https://linux.die.net/man/1/exiftool> для получения подробного руководства по его использованию. **LXF**

## Скорая помощь

Инструменты просмотра изображений могут быть запущены из терминала. Но если вы заблокированы за пределами рабочего стола в оболочке, установите *fbi* (`sudo apt install fbi`) для просмотра ваших файлов (`sudo fbi path/picture.jpg`).



► Музыкальный проигрыватель *CMus* работает аналогично текстовому редактору *Vi*. Не забывайте перед командами ставить символ двоеточия (:).

# Анимация: Пусть фигура движется

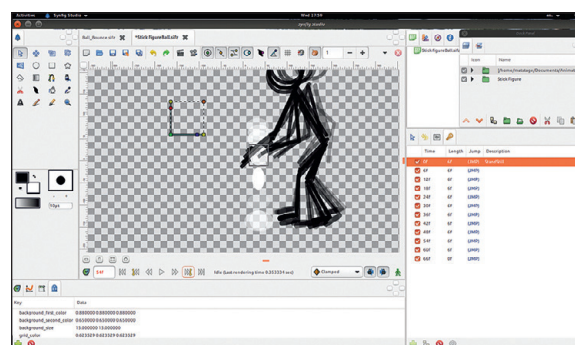
Матс-Таге Аксельссон растолкует вам азы анимации, начиная с основ: подпрыгивающего мячика и ходьбы.



Наш эксперт

## Матс-Таге

**Аксельссон** провел не один десяток лет, заставляя свои компьютеры работать на Linux. В прошлом выпуске он показал вам, какими анимационными программами пользоваться. Теперь учитесь анимировать.



» Вот как изменять форму мяча, когда он подскакивает. Контролируйте степень преувеличения, чтобы это соответствовало стилю вашей анимации.

В реальной жизни вы даже не заметите этого изменения формы, но при анимации это поможет сделать мяч интереснее. Лучший способ проиллюстрировать анимацию — использовать куб или схематичную фигурку из черточек. В руках умелого аниматора даже куб может выглядеть счастливым или грустным!

Возможно, вы видели пример анимации, где куб был нарисован на правой стороне страницы, а затем рисуется копия первого кубика, только уже похожая на желе. Если вы хотите, чтобы это выглядело действительно хорошо, нарисуйте нижнюю часть куба чуть крупнее, чем верхнюю часть, и искривите его боковые стороны. Таким образом, весь куб будет казаться немного согнутым, что заставит его выглядеть эластичным и натуральным. Добавление нескольких тонких линий позади куба создаст ощущение скорости.

## Вырывают опорные кадры

При рисовании множества кадров нужно как минимум десять кадров в секунду, чтобы получилось плавное и чистое движение. Однако вам скоро надоеет прорисовывать каждый кадр. Опорные (ключевые [key]) кадры помогают сократить работу в компьютерной анимации, как мы выяснили благодаря Сравнению ПО анимации в LXF225. Опорные кадры — это наиболее заметные позиции для вашей анимации. Например, когда вы поворачиваете голову из стороны в сторону, опорными кадрами будут начальная позиция, положение в середине и конечная позиция. А программа заполнит промежутки автоматически — это называется tweening.

Количество опорных кадров будет зависеть от баланса между тем, какой объем работы вы хотите выполнить и какой уровень контроля над своим персонажем вам нужен. Меньшее количество опорных кадров может привести к прерывистому или менее реалистичному движению. Возвращаясь к повороту головы — если у вас только три опорных кадра, то в результате вы можете получить

**К** концу данной статьи вы узнаете, как анимировать мальчика, который шагает к баскетбольной корзине, а потом начинает забрасывать мяч и набирать очки. Звучит вроде и не слишком впечатляюще, но, как вы вскоре обнаружите, есть множество деталей, на которые стоит обратить внимание, поскольку нам надо создать и контролировать всё, что увидит зритель на экране.

Сцена создается с фона и содержит людей и объекты: игровое поле, мяч, баскетбольная корзина и мальчик. Нарисовать каждый элемент относительно просто — в конце концов, мяч — это просто сфера; но чтобы всё это было захватывающим, надо подумать, как деформировать его, чтобы он отскакивал от земли, и как он будет лететь по воздуху.

Зрителю будет не очень интересно, если мальчик просто пойдет к кольцу. Чтобы оживить анимацию, надо внести интригу. Представьте, что вы собираетесь забросить мяч в кольцо: вначале вы сгибаете колени, а затем тянете голову к цели. Всё это имеет значение, если вам надо заставить всё выглядеть правильно в каждом кадре. Помните, что вам придется работать с 24 кадрами в секунду, хотя с этим можно немного поиграть.

Заставить анимированный мячик сжиматься и растягиваться, когда он прыгает по поверхности — это способ оживить действие.

## Скорая помощь

При использовании *Blender* загляните в Интернет для дополнений к программе. Прекрасным местом является [www.manuelbastioni.com](http://www.manuelbastioni.com). Художник создал плагин для прототипов людей, с которыми можно работать. Имеется множество моделей, различающихся по национальности, цвету кожи, фигуре и размеру — от совсем худых до толстых.

## Из чего состоит рабочий процесс анимации?

Основная цель создания анимации — рассказать историю. И чтобы привлечь внимание публики и убедиться, что они следят за историей от начала до конца, мы бы предложили придумать вашему персонажу путешествие, желательно с какими-то трудностями [Ред.: — Возможно, Звезда смерти?], которые он будет преодолевать.

Лучший способ спланировать вашу анимацию и получить хороший обзор вашей истории — особенно если она длинная — это создать раскадровку, называемую «аниматик»; она содержит основные моменты вашей истории. Аниматик отображает

опорные позиции и движения во всех сценах. Это дает возможность начать любую сцену с любым персонажем в любой позиции.

Если работу делает команда, то каждый может заняться остальными действиями независимо друг от друга. Если вы работаете в одиночку, вы можете по желанию переключаться между задачами.

Этап моделирования — это когда вы разрабатываете формы персонажей, их одежду, кожу и волосы — или чешую, если это рептилии. По завершении у вас будут фигуры или создания, готовые к анимации.

Оснащение ваших персонажей — это когда вы решаете, какие элементы тела персонажа будут передвигаться и на какой градус.

В случае с людьми вы можете добавить скелет. *Synfig* и *Blender* называют эту функцию Bones [Кости]. Когда скелет настроен, приложение будет использовать его для правильного перемещения рук и ног.

Именно на этапе настоящей анимации все детали объединяются вместе посредством ввода окончательных компонентов, таких как освещение, туман и углы обзора камеры.

поворот головы без всякого начального наклона или окончательной остановки.

Наверное, вам интересно, почему это так важно? Потому что любому движению предшествует подготовка. Скажем, прежде чем повернуть голову, вы кивнете или слегка оглянетесь в противоположную сторону.

Снимите себя или какого-нибудь знакомого, желательно замедленной съемкой, и вы увидите этот феномен в действии. Это называется предвосхищение, и оно крайне важно для того, чтобы движение выглядело реалистично, а также чтобы создать историю.

В качестве примера можно заставить своего персонажа исчезнуть из кадра, чтобы перевести историю на следующий кадр. Другим способом показать это будет заставить фигурку шагать. Чтобы действия персонажа выглядели действительно активными, ему нужно начинать из неподвижного положения, и ходьба начинается с небольшого сгиба коленей. Это приводит к тому, что голова слегка наклоняется. Этот наклон тоже хорошо бы продемонстрировать во время ходьбы, а при беге он создает еще более реалистичный эффект.

### Ходячая фигура

Чтобы фигурка «зашагала», надо разобраться с циклом ходьбы. При ходьбе вы сгибаете стопу, колено и бедро под разными углами. Это кажется очевидным, но детали очень тонкие, и их бывает непросто правильно реализовать. Лучшим способом справиться с этим будет изучение видео, где кто-то ходит. Вы даже можете сами встать [Ред.: — Что?!], с целью посмотреть, как это выглядит.

Вам надо создать все ваши элементы в *Synfig*. Затем дать каждому из них точку вращения и, в случае с ногами, сгруппировать всё с бедром наверху. Когда бедро поворачивается, оно тянет за собой остальную группу, заставляя двигаться всю ногу. Скопируйте ногу и заставьте ее вращаться в противоположную другой ноге сторону, смещая опорные кадры на полшага.

Повторите процесс с руками, и у вас получится идущая фигурка. Звучит достаточно просто... пока вы не попытаетесь сделать это в первый раз! Но, попрактиковавшись, вы сможете создавать более сложные анимированные действия.

Для полной анимации персонажа вам, возможно, понадобится показать второстепенные движения других частей персонажа. Простейшей демонстрацией этого будет нарисовать куб и заставить его сгибаться, как желе, в начальной и конечной точке. Вообразите мальчика, готовящегося забросить мяч: он сгибает колени, его волосы немного шевелятся для полноты картины.

Для создания среды вам, вероятно, также понадобятся кривые Безье. На них основана математическая модель, и в соответствии с ними действуют многие объекты в природе. Используя кривые Безье и другие вычисления, вы можете добиться высокой степени реализма при меньшем количестве усилий.

То же самое работает для траекторий, а также для растений, например, травы. Если вам нужно большое поле, вы можете создать общую картину или рисовать по травинке за раз. Однако если вы используете для описания кривые Безье, то сможете обойтись созданием всего лишь нескольких травинок, сделав их чуть разными, и добавить разнообразия с помощью генераторов случайности. Вскоре вы создадите поле, которое будет казаться состоящим из отдельно нарисованных травинок, хотя на самом деле разных будет всего несколько.

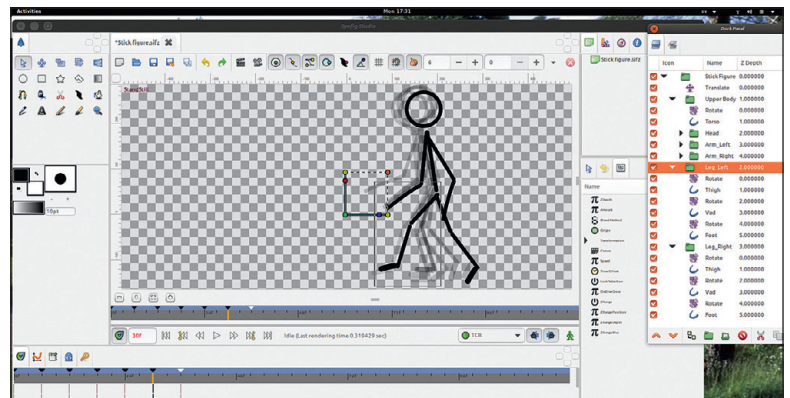
Полет мяча или чего-либо брошенного имеет траекторию параболы до момента соприкосновения с объектом, например, стеной. Единственным исключением будет, если учитывать ветер — и даже тогда результат будет сходным. Простейший способ нарисовать это в программе — использовать кривые Безье.

### Милые кости

Для персонажей, нарисованных не в *Synfig*, вам придется разобрать своего персонажа и добавить к нему «кости». То же самое с *Blender*. Процесс был вкратце описан в *Сравнении* программ анимации в прошлом месяце.

Это делается только для того, чтобы *Synfig* знал, где и до какой степени согнуть рисунок. Примером того, какой это имеет эффект, будет, как если бы вы нарисовали брюки, которые слегка смялись в районе колена, а потом программа смогла бы изменить складки в соответствии со сгибом. Для этой статьи мы сделали более простой пример — фигурку из суставов-палочек, и это позволяет перемещать и поворачивать разные части персонажа.

Когда вы закончите рисунок и примете решение по исходной позе, настраивается начальный опорный кадр. Теперь переходим к настройке следующего опорного кадра. Он может быть через несколько кадров, потому что *Synfig* заполнит для вас промежутки. »



» Заставляя своих персонажей ходить, позаботьтесь, чтобы движение начиналось естественно: например, ногу, прежде чем сделать шаг, надо слегка согнуть.

### Скорая помощь

По возможности ищите бесплатный клип-арт. Для 2D-анимаций его можно использовать в качестве шаблона и для обучения. Использование изображения места, похожего на вашу сцену, также ускорит ваш творческий процесс.

## Скорая помощь

Большая часть художественной работы делается вне анимационной программы. Поэтому при рисовании фона у вас будет более широкий выбор инструментов, цветовых опций и кистей. А потом можно импортировать свое произведение в *Blender* и *Synfig*.

Просто не забудьте добавить небольшие дополнительные движения в начале и конце каждого действия, чтобы сделать его более живым. Ваш персонаж будет, в этом случае, вести мяч, делать несколько шагов, сгибать колени, а затем нацеливаться в корзину.

Готовясь к броску, мальчик немного сгибается, чтобы добавить себе и мячу импульса и баланса. Чтобы быть точным, левая рука будет поддерживать мяч, а правая — бросать. (Наш мальчик — правша.) Сцена также снимается с правой стороны игрока. Чтобы нарисовать это, вначале мы решаем, где разместить игрока, затем — куда он будет двигаться, и, наконец, траекторию мяча в воздухе. Самая сложная часть траектории мяча — обработка отскоков, так что для простоты наш игрок сразу попадет в сетку.

В начале сцены мальчик держит мяч; начиная прицеливаться, он слегка отклоняется назад, чтобы направить взгляд между руками и корзиной. Сгибание рук реалистичным движением требует четко заданных кривых для рук, которые разгибаются в сторону корзины. Мяч пойдет по восходящему углу и полетит по параболе через всю сцену и над кольцом, после чего проскользнет через натянутую на кольцо сетку. Когда мальчику засчитают очко, он подпрыгнет от радости, а мяч отскочит и покатится по полю. И наконец, мальчик подбежит к мячу, поднимет его и отойдет от камеры.

## Отскакивающие мячи

Мяч у нас уже отскакивал в сторону в другой сцене. А теперь нам надо, чтобы он подпрыгивал вверх и вниз, чтобы мальчик мог поймать мяч, готовясь к броску.

Продолжая в *Synfig*, мы создаем мяч как отдельный объект и добавляем уровень трансформации и разворот игрока к мячу. Этот уровень затем будет использован для настройки положения мяча в сцене. Руки будут синхронизированы с мячом вручную, так что не забудьте сделать так, чтобы подскоки выглядели реально.

Чтобы поднять мяч, вам надо синхронизировать фигуру с движением мяча. В *Synfig* для этого надо переместиться по опорным

кадрам и изменить сцену на один кадр за один раз. К счастью, вам достаточно изменить лишь несколько опорных кадров, чтобы добиться реалистичного результата.

Это действие происходит после того, как мяч подпрыгнул несколько раз, так что вам надо настроить новый опорный кадр, где мяч только что вернулся к мальчику. Даже здесь происходит несколько вещей. Во-первых, руки мальчика возьмут мяч. Во-вторых, его колени немного согнутся. В-третьих, его голова повернется лицом к корзине. В-четвертых, руки поднимутся и опустятся. В-пятых, руки совершат движение вперед; и, наконец, мяч полетит через всю сцену. Если вы не потратите время на планирование этой подробной сцены, работа по ее анимации очень скоро усложнится.

## Обнаружение столкновений

Если сцена содержит несколько движущихся объектов, вам надо настроить обнаружение столкновений. Однако в *Synfig* положение мяча в руки персонажа зависит только от вас. Ничто не мешает мячу проскользнуть сквозь руки, или наоборот. А значит, надо задать позицию мяча позади рук и затем запустить его вместе с ожидаемой траекторией, чтобы он попал в корзину в реалистичной манере.

Составление плана движения всех объектов в сцене очень полезно, поскольку позволит вам создать свежие опорные кадры для всех действующих элементов в сцене. Уточнение уже созданного пути может быстро стать весьма запутанным и привести к отнюдь не идеальным результатам.

Вкладка *Physics* в *Blender* позволяет вам управлять всеми видами материалов и тем, что с ними происходит при движении. Чтобы мяч летел по воздуху, вы сперва вводите его в сцену и придаете ему начальную скорость. Вы также можете решить применить к нему некоторую силу или даже заставить его столкнуться с каким-то объектом.

## Пусть это будет просто фигурка

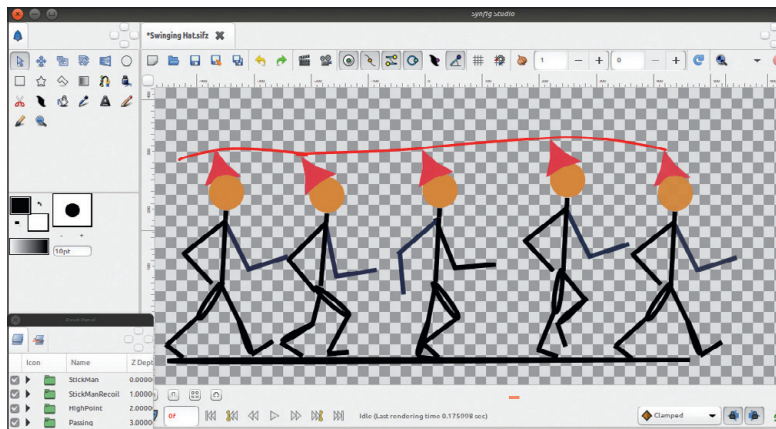
Начинать с фигурки из черточек удобно по нескольким причинам. Упрощенная форма поможет вам сформулировать ее движения, и вдобавок она будет не столь требовательна к вашему оборудованию.

Чтобы движения выглядели реальными, вам надо основывать движения на том, как перемещается персонаж. Лучший способ увидеть это — нарисовать скелет и обратить внимание на то, как движутся суставы. Изучение человеческой анатомии очень полезно для художника-аниматора, даже если его герои — инопланетяне!

Когда вы решите, как будет совершаться движение, следует узнать, где это движение ограничить, чтобы всё выглядело реально. Если вы не сделаете этого, ваша публика потеряет интерес к сюжету, и получится, что вы зря потратили время на свой проект.

Когда движения персонажа будут сбалансированы, можно приступать к моделированию. Это работа монотонная, и большую ее часть составляет рисование картинок перед добавлением

➤ Чтобы оживить персонажей, добавьте второстепенных движений, например, у шляпы. Также не забывайте, что при ходьбе люди немного подпрыгивают.



## Понимание функции частиц

Когда вы воссоздаете воду или что-то другое с поведением жидкости, используется система частиц. Суть системы в том, чтобы имитировать физику реального мира как можно точнее, и самый простой способ понять ее — поиграть с шариками для пинг-понга.

Возьмите миску и внимательно проследите, как шарики падают в нее. Еще лучше, если вы снимете их падение, а потом воспроизведете это действие кадр за кадром, чтобы прочувствовать движение. Система частиц программы рассчитывает

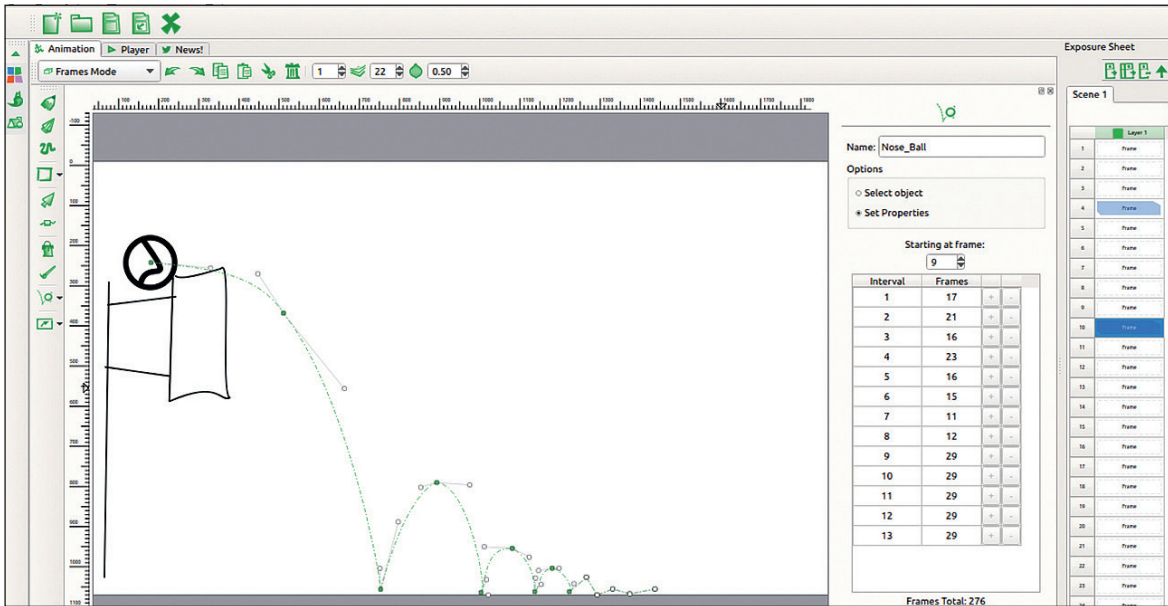
траекторию частицы, принимая во внимание начальную скорость, жесткость и скорость других близлежащих частиц.

В случае с водой программа также принимает во внимание, сколько частиц сливаются в одну, это называется мягкое столкновение. Странно здесь то, что при вычислении этих столкновений используется математическая аналогия тепла, где центр считается горячим, а линия контура рисуется при определенной более низкой «температуре» вокруг частицы. На некоторой дистанции,

которая определяется этой температурой, линии двух или более различных частиц сливаются, образуя жидкость.

С помощью этой системы вы можете создавать дождь, водопады и прочие водные объекты. Поместите лодку в океан, рыбу — в воду, и вы сможете создать сцену с рыбаками.

В Сети также есть прекрасные симуляции, и если вы запустите *Blender*, похожая система с таким же именем доступна во вкладке *Physics*, где можно ломать объекты и поджигать их!



► Используя Turi, вы можете задать траекторию объекта, и Turi создаст кадры, используя функцию под названием tweening ball.

дальнейших деталей. Чтобы ваш персонаж выглядел реалистично, вам надо добавить одежду, лицо и кожу.

## Добавляем реализма

Правдоподобный вид поможет вашим зрителям сопереживать персонажам. Для 2D-картинки надо рисовать персонажей с градиентами, чтобы убедительно передать форму. Всегда есть тени, которые слегка изменяются в разных частях поверхности, и лучше всего применять к ним градиенты.

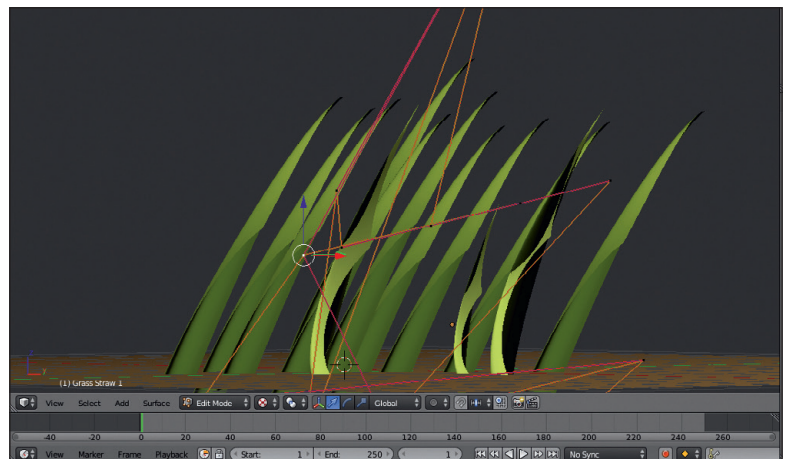
Не забудьте, что по всей сцене тени должны простираяться в одном и том же направлении. Иначе ваши зрители подумают, что персонажи немного не на месте. На очень ранней стадии вы должны решить, где будет находиться источник света, чтобы правильно разместить тени от ваших персонажей. Сначала создайте тени на полу (земле), а затем пользуйтесь этими тенями как основой для указания градиентов всех элементов в сцене.

Слово «моделирование» во всей полноте применимо именно в 3D. 3D-модели начинаются с простых форм, таких как кубы, сферы и конусы. Затем они редактируются в разных режимах, чтобы вы могли создать каждую деталь. Желая создать голову своему персонажу, вы обычно начинаете со сферы, расчерченной линиями. Это ваша основа. Следуя линиям и сгибая их, вы создадите черты лица, например, щеку. Однако было бы полезно приставить нос из другого элемента. Все эти формы определяются как сетки [mesh], что позволит вам впоследствии вычислить отражения.

Все сетки начинаются с некоторого числа граней, и изначально граней должно быть как можно меньше. Дело в том, что каждая

грань потребляет память, и когда вы начинаете с большого количества, страдает производительность. Когда вам понадобятся более тонкие детали, добавьте грани в течение процесса, а не в начале. Другим решением будет использовать плагин, содержащий модели людей, которые вы можете привести в соответствие внешнему виду своего персонажа, но моделирование — это отдельное обширное поле, и мы оставим его на другой раз.

Веселой вам анимации! LXF



► Здесь показано, как создается поле травы по технологии средней точки. В Blender этот метод называется NURBS.

## Делаем траву реалистичной

Возможно, вы думаете, что траву изобразить легко: просто рисуется несколько изогнутых зеленых линий. Правда заключается в том, что это редко выглядит реалистично, и вы уж точно не захотите вырисовывать каждую из множества травинок вручную.

К счастью, математика весьма проста и, таким образом, идеальна для применения на компьютере, вычисляющем для нас травяную лужайку. Добавленные доли случайности завершает натуральный вид. Каждая травинка будет определяться несколькими

углами между концами открытого треугольника, называемыми контрольными точками. При разработке травинки у вас будет две линии, соединенные посередине, а концы линий — это контрольные точки, которыми вы можете управлять. Математический элемент вычисляет середину опорной линии между двумя линиями.

Концы опорных линий являются единицами 't' от конца A и 1-t от конца C, где концы именуется от точки A на ближнем конце через точку B на пересечении и далее точкой C на дальнем конце.

Вы можете потренироваться на листке бумаги, нарисовав две линии, образующие угол, который будет задавать сгиб травинки.

Чтобы сделать это, найдите середину между двумя линиями и отметьте ее, а затем продолжите делить линию пополам, пока у вас не окажется по восемь точек на каждой линии. Пронумеруйте их от 1 до 8, начиная с точки A, и после точки B начните заново с одного. Дальше соедините точки 1, затем точки 2, и так со всеми до точки 8. Видите появившуюся параболу?



# Безопасность: Как удаляют шифрование

Эрмин Крепоник объясняет принципы классической переадресации пакетов и использования атаки «человек-посредник» для обхода защиты сайтов.



**Наш эксперт**

**Эрмин Крепоник** — эксперт в сфере ИТ и большой энтузиаст Linux; он обожает разбираться с проблемами сети. Спит он маловато.

В этой статье рассказывается об перехвате данных посредством *sslstrip* и ARP-спуфинга. Атаковать беспроводные сети можно не только путем взлома ключей аутентификации. Существует множество других техник, в особенности в публичных Wi-Fi, где аутентификацию пройти не проблема.

После аутентификации вы сможете прослушивать весь трафик в этой сети. Однако, скорее всего, этот трафик будет зашифрован, так что проку от этого вам, видимо, не будет. Вы сможете увидеть, какие IP-адреса посещают какие сайты, но не сможете увидеть что-либо полезное для злоумышленника.

Проблема заключается в HTTPS, который шифрует ваш веб-трафик. Входные учетные данные должны передаваться через HTTPS, чтобы их невозможно было перехватить. Программа *sslstrip* преобразует HTTPS-соединения в HTTP, эффективно удаляя шифрование. Если сделать это удастся, то все пакеты будут передаваться в открытую, и вы сможете увидеть любые имена пользователей и пароли, которые люди в сети используют для входа на сайты, как если бы эти сайты не использовали HTTPS.

На нашем уроке предполагается, что мы работаем с Fedora. Нам понадобится два инструмента: *sslstrip* и *arpspoof*. Последний служит для подтасовки [spoofing] протокола разрешения адресов (ARP), при котором мы подтверждаем маршрутизатору нашу псевдо-идентичность, чтобы он отправлял нам пакеты, отправленные пользователю. Далее мы снимаем шифрование и вновь передаем пакеты маршрутизатору, который затем отправляет их в Интернет.

Установите необходимые пакеты (все команды этого урока должны запускаться от имени root) таким образом:

```
# yum install sslstrip
```

```
# yum install dsniff
```

В большинстве других дистрибутивов второй пакет называется *arpspoof*, а если вы используете Kali, рассчитанный специально на хакеров, то оба они уже установлены.

Передавая перехваченные пакеты, связанные с жертвой через маршрутизатор, злоумышленник общается с жертвой через HTTP, поэтому всё отправляется в открытом виде. Это классическая атака по принципу «человек-посредник» (Man in the Middle, MitM).

Она не на 100 процентов эффективна. Иногда это зависит от установленного браузера (*Internet Explorer*, скорее всего, будет более подвержен ей, чем *Firefox* или *Chrome*); некоторые сайты, такие как PayPal и Facebook, предприняли меры, чтобы затруднить реализацию подобной атаки.

Тем не менее, поскольку люди склонны использовать одни и те же парольные данные на разных сайтах, детали, собранные с менее защищенного сайта, пригодятся и на том, где атака оказалась неудачной. Мы советуем иметь один набор паролей для более важных сайтов — банки, электронная почта, социальные сети — для всего, где известен ваш адрес, и один набор паролей для менее важных сайтов, вроде учетных записей форумов. Если в последних пароли совпадают — не страшно, учетные данные на них навряд ли будут перехвачены (с помощью методов, которые мы здесь приводим) с PayPal и ему подобных, так что в худшем случае взломана будет пара ваших учетных записей с форумов. Сомнительно, чтобы они были интересны хакерам.

## Построение атаки

Сначала надо настроить наше ядро для пересылки пакетов:

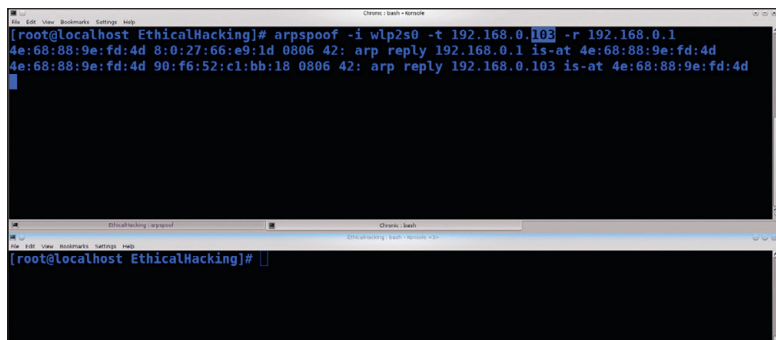
```
# echo 1 > /proc/sys/net/ipv4/ip_forward
```

Затем нам нужно сообщить *iptables* о том, как надо перенаправлять пакеты:

```
# iptables -t nat -A PREROUTING -p tcp --destination-port 80 -j REDIRECT --to-port 8080
```

Программа *sslstrip* будет прослушивать порт 8080, поэтому будут перенаправляться пакеты, предназначенные для веб-серверов (порт 80).

Теперь нам надо найти цель. В нашей статье предполагается, что это виртуальная машина Windows 8.1, но на практике это может быть всё, что угодно. Не атакуйте машину без разрешения. В реальной ситуации вы сначала сканируете сеть, можно сделать это с помощью известной утилиты *nmap*:



**arpspoof** расскажет вам обо всех ложных сообщениях ARP, которые он отправляет, чтобы обмануть нашу жертву, выдав себя за сетевой шлюз.



udemy

Запишитесь  
сейчас  
и сэкономьте  
£185!

## Учитесь и экономьте с Udemу

Если вам понравился этот маленький фрагмент «Полного курса этичного хакинга» от Udemу, вы можете получить неограниченный доступ к нему на [udemy.com](http://udemy.com) с эксклюзивной скидкой для читателей *Linux Format*. Вы приобретете практические знания в области этичного взлома, тестирования проникновения, web-тестирования и взлома Wi-Fi, оставаясь в безопасной среде. Вы также получите рекомендации о том, как, используя эти навыки, начать карьеру внештатного хакера и многое другое...

Чтобы записаться на курс по сниженной цене £10 (скидка 95%), посетите сайт <http://bit.ly/ETHICALHACKING10>. Нажмите кнопку Купить сейчас и зарегистрируйтесь для получения учетной записи на Udemу. Когда вы регистрируете учетную запись, вам предложат подтвердить вашу покупку. Выбор курса по скидочной стоимости £10 будет произведен автоматически при использовании кода **ETHICALHACKING10**. Введите данные своей кредитной карты и нажмите Оплатить сейчас. Вы успешно

зачислены на курс! Теперь у вас есть постоянный неограниченный доступ.

Платформа Udemу была основана в 2010 г., чтобы каждый мог улучшить свою жизнь обучением. Udemу — это глобальный рынок для обучения и преподавания в Интернете, где более 15 миллионов студентов обучаются в библиотеке из 45000 курсов, преподаваемых экспертами-инструкторами на 80 различных языках.

\* Скидка действительна до 7 августа 2017 г.

```
nmap 192.168.1.1/24
```

Наша целевая виртуальная машина расположена по адресу [192.168.1.100](http://192.168.1.100), поэтому *nmap* отвечает:

```
Scanning 192.168.1.100 [1000 ports]
```

```
Discovered open port 80/tcp on 192.168.1.100
```

```
Discovered open port 443/tcp on 192.168.1.100
```

Для этой атаки нам потребуется три окна терминала: два для построения самой атаки и одно для отслеживания результатов. Мы собираемся убедить маршрутизатор, что мы являемся виртуальной машиной Windows 8.1, а саму машину убедим, что мы являемся маршрутизатором. Наша атака ARP-спуфинга будет выглядеть так (здесь мы предполагаем, что атака выполняется по беспроводному интерфейсу `wlp2s0`):

```
# arpspoof -i wlp2s0 -t 192.168.1.100 -r 192.168.1.1
```

Аргумент `-t` — это целевой компьютер, а аргумент `-r` — шлюз по умолчанию. Команда перенаправляет весь трафик жертвы на нашу атакующую машину. Поскольку ранее мы включили пересылку с помощью *iptables*, подключение к сети жертвы не будет затронуто, за исключением того, что web-трафик будет перехвачен *sslstrip*, которую мы тут и запустим:

```
# sslstrip -l 8080
```

Как только *sslstrip* начнет перехватывать трафик, она выведет кучу абракадабры, поэтому проще отслеживать логи в нашем третьем окне терминала с помощью

```
# tail -f sslstrip.log
```

Теперь на целевой машине откройте *Internet Explorer* и *Firefox*. Иногда в одном браузере атака срабатывает, а в другом — нет. Иногда не срабатывает ни в том, ни в другом; но давайте не будем об этом беспокоиться. В реальной ситуации вам придется подождать, пока не начнется просмотр web-страницы. Попробуйте посетить <https://gmail.com> в *Internet Explorer*. Если атака работает, то *sslstrip* должен перехватить трафик, чтобы страница передавалась по HTTP — проверить это можно, просмотрев строку URL. Это определенно не будет работать с более новыми браузерами, но в нашей несколько искусственной целевой машине всё получается.

Введите в форму входа какие-нибудь учетные данные наобум, скажем, FAKE для имени пользователя и пароля, и нажмите Войти. Журнал *sslstrip* должен выдать немалую порцию данных, начиная с чего-то вроде

```
2017-01-07 00:01:02,345 SECURE POST Data (accounts.google.com)
```

Но где-то среди этой тарбарщины должно быть нечто вроде `...checkedDomains=youtube&Email=FAKE&Passwd=FAKE` — это показывает, что атака прошла успешно. На нашей целевой машине эта атака не сработала в *Firefox* или *Chrome*, но по данным недавнего опроса Netmarketshare, около 18% всё еще пользуется

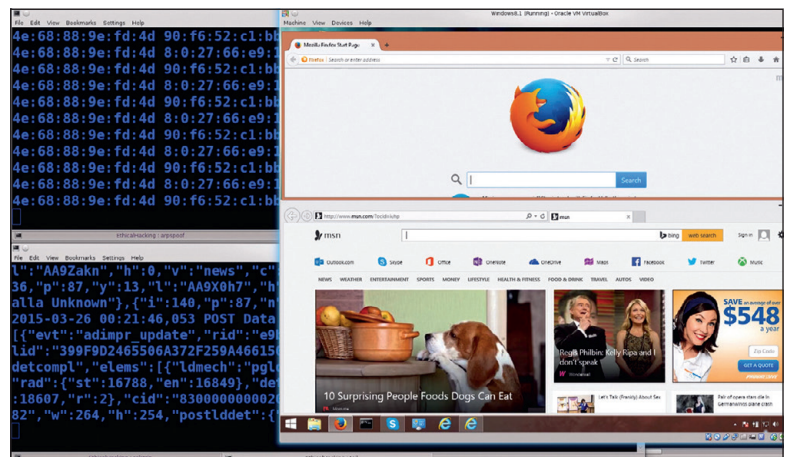
какой-нибудь версией *Internet Explorer*, поэтому не исключено, что и в реальности это где-то сработает. В конце концов, многие учреждения неаккуратно применяют обновления безопасности и заставляют пользователей по традиции использовать всякие ужасные программы. Столь же успешно мы перехватили регистрационные данные Facebook из *Internet Explorer*. Хотя в других браузерах атака *sslstrip* не сработала с основными сайтами, это не значит, что она не сработает для прочих.

## Как бороться с sslstrip

Обратите внимание, что сами сайты не обслуживают страницы входа в систему HTTP, они передаются на нашу атакующую машину через HTTPS, и мы отправляем их в целевой HTTP. Таким образом, противодействие этому со стороны сервера намного сложнее, чем просто обслуживание страниц HTTPS.

Лучший способ — использовать HTTP Strict Transport Security (HSTS). При этом сервер отправляет поверх соединений HTTPS дополнительный заголовок с флагом `max-age`. Это интервал (в секундах) после первого доступа, по прошествии которого сайт будет использовать только HTTPS. Таким образом, если первый раз наша жертва заходит на сайт без стороннего вмешательства, на любые последующие попытки просмотреть страницу через HTTP браузер будет жаловаться.

Другой вид защиты состоит в том, чтобы приучить пользователей в <https://> всегда искать `z` или же дружелюбный зеленый замочек. Но пользователи порой забывчивы. А вот расширение браузера HTTPS Everywhere пресекает такую атаку на корню, и хорошо бы оно стало органичной частью просмотра web-страниц. **LXF**



► Может показаться, что атаковать Windows 8 и *Internet Explorer* — всё равно что стрелять по селедке в бочке, однако люди до сих пор пользуются этими (жуткими) штуками.

Скорая  
помощь

Учтите, что из-за особенностей вашей системы могут возникнуть всевозможные проблемы. Могут возникнуть сообщения об ошибках, необходимость в установке или обновлении дополнительных библиотек. Не волнуйтесь: это нормально и является частью процесса обучения.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)

# Атака хакера: Скрытый ввод

USB Rubber Ducky кажется обычной флешкой, но с нее можно вводить, как с клавиатуры. **Нейт Дрейк** осваивает эту компактную платформу для атак.



## Наш эксперт

**Нейт Дрейк** — журналист-фрилансер в области технологий, специализирующийся в написании статей о кибербезопасности. Он просто в восторге от описываемого устройства.

**К**омпьютеры доверяют людям. Люди используют клавиатуры. И если до клавиатуры вашего компьютера дорвется неправильная личность, хаос обеспечен.

USB Rubber Ducky [Резиновая Уточка], несмотря на свое безобидное название, является одним из устройств, которые этому способствуют. С технической точки зрения это злонамеренное HID (устройство интерфейса человек–машина), которое можно вставить в USB-порт компьютера или телефона и запустить тысячи команд за несколько секунд. Фанаты сериала *Мистер Робот* могут вспомнить, что его персонаж Анжела Мосс [Angela Moss] использовала Ducky таким образом на компьютерах ФБР, возымев разрушительный эффект.

История устройства USB Rubber Ducky восходит к 2010 г. С тех пор оно претерпело некоторые изменения, в основном ради того, чтобы стать как можно более доступным и эффективным. Сам проект Ducky частично вдохновлялся одним из проектов Arduino, под названием Teensy ([www.pjrc.com/teensy](http://www.pjrc.com/teensy)), который тоже пригоден для тестирования на вторжения, но не так прост в использовании по умолчанию.

## Законная самозащита

Это отличная формулировка ответа на возражения тех, кто не верит, что есть законные способы применения Rubber Ducky. Черные хакеры давно умеют создавать похожие устройства, поэтому, будучи тестировщиком на вторжения, вы обязаны убедиться, что ваши устройства защищены от такого рода атак.

Некоторые из основных скриптов, доступных для скачивания, содержат простую инъекцию [payload] для создания нового администратора, отключения брандмауэров, установления

удаленного доступа, создания скрытых сетей Wi-Fi, и т. д. Официальным инструментом перевода данных на простом языке скриптов (Ducky Script) в файл .bin является *Duck Encoder*. На нашем уроке мы рассмотрим основы настройки *Duck Encoder* для запуска такой инъекции.

## Duck Duck Go

Если вам не терпится приступить, мы рекомендуем приобрести как минимум два устройства Rubber Ducky на Hak5 (см. *Скорая помощь*, слева). Тогда вы сэкономите на доставке, а также сможете дать второе устройство другу, занимающемуся тестированием на вторжения. То есть с вашего разрешения кто-то другой может выбрать инъекцию и выполнить ее на ваших устройствах. Это будет больше похоже на действия настоящего хакера, который пытается проникнуть в ваш дом или офис. Если вы сумеете убедить своего коллегу-тестировщика позволить вам сделать то же самое с его устройствами, то сможете насладиться незабываемым ощущением попытки взлома чужих компьютеров.

Базовый набор состоит из самого Rubber Ducky, карты microSD на 128 МБ, корпуса, который маскирует Rubber Ducky под флешку, читалки microSD и адаптера OTG Micro USB. Есть также несколько наклеек и справочник.

Не хватает только компьютера для подготовки вашей инъекции и целевого компьютера для ее развертывания. На нашем уроке для подготовки инъекции использовалась виртуальная машина с Ubuntu 17.04 (Zesty Zapus), а целевым компьютером был нетбук Lenovo с Windows 10. *Duck Encoder* многоплатформенный, и будет работать на любом устройстве, поддерживающем Java, так что можете спокойно использовать другую версию Linux — при условии, что у вас установлена Java Runtime Environment.

Наша инъекция откроет блокнот на компьютере с Windows и введет сообщение «ВАС КРЯЖНУЛИ!». Примеры более впечатляющих инъекций вы можете найти на странице Hak5 Github (<https://github.com/hak5darren/USB-Rubber-Ducky/wiki/Payloads>), форумах Hak5 (<https://forums.hak5.org>) и на сайте Duck Toolkit.

Хотя эти простые инъекции никоим образом не должны повредить систему, позаботьтесь, чтобы вы и ваш коллега-тестировщик (если таковой имеется) сделали резервные копии всех устройств, на которых будет использоваться Rubber Ducky.

К Rubber Ducky прилагается корпус, маскирующий его под флешку, но вы можете предпочесть сначала обойтись без него, потому что, скорее всего, свою карту microSD вам придется не раз вытаскивать и вставлять.

## Обсуждая Ducky Script

Инъекции готовятся с помощью Ducky Script. Любому, кто использовал язык программирования BASIC в былые времена, это покажется детским лепетом. Остальные научатся этому за пару минут.



## Скорая помощь

Закажите USB Rubber Ducky на <https://hakshop.com/products/usb-rubber-ducky-deluxe> за \$45 плюс доставка.

» Rubber Ducky поставляется вместе с корпусом, маскирующим его под флешку. Наклейка с уточкой является опциональной!

## Подготовьте себе Rubber Ducky

```

Resolving github.com (github.com)... 192.30.253.113, 192.30.
Connecting to github.com (github.com)|192.30.253.113|:443...
HTTP request sent, awaiting response... 302 Found
Location: https://cloud.github.com/downloads/hak5darren/USB-
duckencode.jar [following]
--2017-05-05 15:11:03-- https://cloud.github.com/downloads/
r-Ducky/duckencode.jar
Resolving cloud.github.com (cloud.github.com)... 54.192.28.1
4, 192.28.94, ...
Connecting to cloud.github.com (cloud.github.com)|54.192.28.
d.
HTTP request sent, awaiting response... 200 OK
Length: 13861 (14K) [application/java-archIve]
Saving to: 'duckencode.jar'

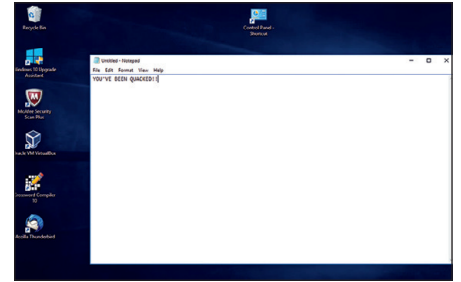
duckencode.jar 100%[=====] 13.54K ---
2017-05-05 15:11:03 (40.5 MB/s) - 'duckencode.jar' saved [13
nate@nate-VirtualBox:~$ java -jar duckencode.jar

```

```

GNU nano 2.7.4 File: helloworld.txt Modified
REM MY FIRST SCRIPT
DELAY 5000
CTRL ESC
DELAY 5000
STRING notepad.exe
ENTER
DELAY 5000
STRING YOU'VE BEEN QUACKED!!
ENTER

```



### 1 Скачайте Duck Encoder

Откройте на своем компьютере Terminal и убедитесь, что у вас установлен Java, посредством запуска `sudo apt-get install default-jre`. Скачайте Duck Encode, запустив команду

```
https://github.com/hak5darren/USB-Rubber-Ducky/blob/master/duckencoder.jar
```

Затем вставьте карту microSD с Duck в читалку и вставьте читалку в свободный порт USB. Проверьте точное имя вашей карты в вашей папке `/media`, например, `/media/nate/A87B-A154/`. Затем заново откройте Terminal и запустите `nano helloworld.txt` для создания текстового файла

### 2 Подготовьте свою инъекцию

Вставьте следующее:

```

REM МОЙ ПЕРВЫЙ СКРИПТ
DELAY 5000
CTRL ESC
DELAY 5000
STRING notepad.exe
ENTER
DELAY 5000
STRING ВАС КРЯКНУЛИ!
ENTER

```

Нажмите `Ctrl+X`, `Ctrl+Y`, а затем `Enter`, чтобы сохранить данные и выйти.

### 3 Запустите инъекцию

Затем запустите `java -jar duckencode.jar -i helloworld.txt -o <sd location>/inject.bin`, заменив `<sd location>` на соответствующее местоположение в вашей системе. Чтобы безопасно извлечь карту microSD, воспользуйтесь своим менеджером файлов, или запустите `eject <device>`.

Теперь извлеките карту из USB-читалки и аккуратно вставьте ее в USB Rubber Ducky. Найдите компьютер с Windows, к которому у вас есть легальный доступ, и вставьте устройство в любой момент после входа пользователя. Наш скрипт намеренно замедлен, чтобы показать работу Rubber Ducky.

Чтобы создать инъекцию, откройте любой текстовый редактор и в первой строке напишите `REM`. Как хорошо известно крутым пользователям BASIC, после оператора `REM` Ducky будет игнорировать текст, что позволяет вам ввести описание вашего скрипта, такое как `'Disable Firewall [Отключить брандмауэр]`.

Везде, кроме описаний, обязательно писать заглавными буквами. Ими должны быть написаны все команды, и каждая начинается с новой строки. Первая команда в скрипте обычно `DELAY <value>`, где `<value>` задает значение времени в миллисекундах, которое Rubber Ducky будет ждать перед попыткой исполнить остальную часть скрипта. Это важно, поскольку Ducky печатает куда быстрее обычного пользователя клавиатуры, а компьютеру может потребоваться некоторое время на распознавание устройства и/или запуск программ. Задержки, использованные в примерах скриптов в руководстве, составляют 5 секунд (5000 миллисекунд), так что вы можете следовать шагам, по которым проходит Ducky, но при желании изменить это значение.

Специальные клавиши вызываются посредством печати их имен заглавными буквами с новой строки. Например, `WINDOWS` соответствует нажатию на кнопку Windows на клавиатуре Microsoft. Другие команды, такие как `CTRL`, `ESC` и `SPACE` [пробел], не требуют объяснений. Для печати самого текста используйте команду `STRING` — например, `STRING ВАС КРЯКНУЛИ!`. Затем отпразднуйте освоение аж целого языка программирования всего за пару минут и сохраните свой скрипт в виде простого текстового (`.txt`) файла.

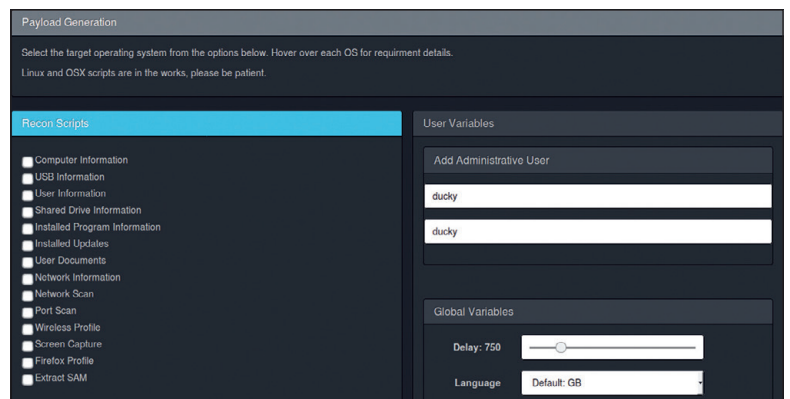
## Инструментарий Ducky

Руководство описывает, как воспользоваться официальным *Ducky Encoder*, чтобы взять ваш новосозданный файл `.txt`, написанный на Ducky Script, и трансформировать его в пригодную к употреблению инъекцию (файл `inject.bin`). Синтаксис команд не слишком сложный, а если вам захочется освежить память, запустите `java -jar duckencode.jar`, чтобы увидеть список доступных команд. При копировании данных укажите точное положение своей карты microSD,

иначе `.bin` файл просто появится в той же папке, где вы запустили *Ducky Encoder*.

Если вы предпочитаете использовать GUI, подумайте о том, чтобы заглянуть на <https://ducktoolkit.com>. Набор инструментов Duck Toolkit содержит готовые инъекции для Windows, Mac и Linux. Нажмите Payload Generator для их просмотра. Нажмите на вкладку Encode, чтобы написать скрипт прямо в своем браузере, затем нажмите на кнопку Generate Script справа. Duck Toolkit сгенерирует файл `inject.bin`, который можно скачать прямо на карту microSD, что избавит вас от проблемы работы в командной строке.

Мы еще раз настоятельно убеждаем вас сначала получить разрешение владельца системы на использование Ducky. Это обязательно, даже если вы знаете, что он может быть очень уязвим и только выиграет от ваших действий (и особенно тогда). Неавторизованный взлом вряд ли поспособствует укреплению доверия к вам, а вот дружеское предложение помощи способно творить чудеса... **LXF**



► Duck Toolkit очень легко использовать для подготовки скриптов и скачивания их напрямую в виде файла `inject.bin`.

**Скорая  
ПОМОЩЬ**

Нажмите на раздел User Scripts сайта Duck Toolkit, чтобы увидеть инъекции, разработанные сообществом.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)

# Ядро: Модули и настройки

Джонни Бидвелл продолжает свой путь к нирване сисадмина, останавливаясь, чтобы посидеть под деревом и поразмышлять о тайнах ядра и модулей ядра.



Наш эксперт

Джонни Бидвелл охвачен предвыборной лихорадкой. Какие глупые политики будут навязываться нам дальше? Какие основные права нам больше не нужны?

Ядро операционной системы лежит в основе всех систем Linux — от самой маленькой встроенной или для мобильного устройства до самого крупного суперкомпьютера. Примечательно, что в широком спектре ядро имеет те же основные задачи. Оно управляет доступом к оборудованию; конкуренцией за ресурсы между различными приложениями и другими задачами; операциями ввода-вывода; файлами; хранилищем данных; безопасностью; сетями и так далее. Выбрав параметры командной строки ядра при загрузке, можно заставить систему вести себя по-разному.

На этом уроке мы продолжим нашу презентацию курса Linux Foundation *LFS201 Essentials of Linux System Administration*. Здесь мы рассмотрим главу 6: что делает ядро и как оно это делает, параметры командной строки ядра и утилиту *sysctl* для их настройки. Мы также коснемся главы 7, дав обзор модулей ядра.

В узком смысле Linux — это только ядро операционной системы, которое включает множество других компонентов, таких как библиотеки и приложения, взаимодействующие с ядром. Ядро является основным центральным компонентом, который соединяет оборудование с программным обеспечением и управляет системными ресурсами, такими как распределение памяти и процессорного времени между конкурирующими приложениями и службами. Оно управляет всеми подключенными устройствами с помощью драйверов устройств и делает устройства доступными для использования операционной системой. Система, работающая только с ядром, имеет довольно ограниченную функциональность, и ее можно найти только в узкоспециализированных встраиваемых устройствах.

Основные сферы ответственности ядра таковы:

- » Инициализация системы и загрузка
- » Планирование процессов
- » Управление памятью
- » Управление доступом к оборудованию

- » Ввод-вывод между приложениями и устройствами хранения данных
- » Контроль безопасности — как локально (например, разрешения файловой системы), так и через сеть
- » Реализация локальных и сетевых файловых систем
- » Управление сетевыми интерфейсами

При загрузке системе передаются различные параметры в командной строке ядра. Обычно они находятся в строке `linux` в файле настройки *Grub*, но могут быть изменены при загрузке. Пример командной строки ядра в *Grub*, определенной в `/boot/grub2/grub.cfg`, может выглядеть так:

```
linux16 /boot/vmlinuz-3.19.1.0 \
root=UUID=01234567-8888-9abc-def0-fedcba987654 ro \
vconsole.keymap=us crashkernel=auto \
vconsole.font=latacyrheb-sun16 rhgb quiet LANG=en_US.UTF-8
```

или, возможно, несколько проще, например:

```
linux16 /boot/vmlinuz-4.7.3 root=LABEL=RHEL7 ro
```

Всё, что указано после файла `vmlinuz`, является параметром. Любые параметры, не распознанные ядром, будут переданы *init* — первому пользовательскому процессу (`pid1`), который запускается в системе. Обратите внимание, что `grub.cfg` не следует редактировать напрямую, так как при обновлении ядра он будет перезаписан. Вместо этого надо создать новый файл, отредактировав файлы в `/etc/grub` и запустив `grub-mkconfig`. Чтобы узнать, с какой командной строкой загружена система, введите

```
$ cat /proc/cmdline
```

Если была использована указанная выше короткая строка, то выводом этой команды будет

```
BOOT_IMAGE=/boot/vmlinuz-4.7.3 root=LABEL=RHEL7
```

## Параметры загрузки

Существует на удивление длинный список параметров ядра, который вы можете просмотреть на <https://www.kernel.org/doc/Documentation/admin-guide/kernel-parameters.rst>, в самих исходниках ядра в `Documentation/kernel-parameters.txt` или набрав `man bootparam`. Параметры могут указываться просто как значение, заданное в качестве аргумента, или в форме `param=value`, где значение `value` может быть строкой, целым числом или массивом целых чисел, как описано в документации.

Раньше мы использовали в длинной командной строке следующие параметры:

- » `root` Корневая файловая система
- » `ro` Монтирует корневое устройство только для чтения при загрузке
- » `vconsole.keymap` Какую клавиатуру использовать для консоли
- » `crashkernel` Сколько памяти зарезервировать для `crashdumps` [сброшенной информации о сбое] ядра
- » `vconsole.font` Какой шрифт использовать для консоли

» Драйвер сетевого устройства Intel PRO/1000 принимает ряд параметров, которые могут быть указаны в конце заклинания `modprobe`.

```
param: debug:Debug level (0=none,...,10=all) (int)
[.john@acerofspades ~]$ modinfo e1000 | grep -v alias
filename: /lib/modules/4.10.13-1-ORCH/kernel/drivers/net/ethernet/Intel/e1000/e1000.ko.gz
version: 7.3.21-k8-NAPI
license: GPL
description: Intel(R) PRO/1000 Network Driver
author: Intel Corporation, <linux.nics@intel.com>
srcversion: E7724229CD981592489467
depends:
intree: Y
vermagic: 4.10.13-1-ORCH SMP preempt mod_unload modversions
param: TxDescriptors:Number of transmit descriptors (array of int)
param: RxDescriptors:Number of receive descriptors (array of int)
param: Speed:Speed setting (array of int)
param: Duplex:Duplex setting (array of int)
param: AutoNeg:Advertised auto negotiation setting (array of int)
param: FlowControl:Flow Control setting (array of int)
param: XsumKX:Disable or enable Receive checksum offload (array of int)
param: TxIntDelay:Transmit Interrupt Delay (array of int)
param: TxAbsIntDelay:Transmit Absolute Interrupt Delay (array of int)
param: RxIntDelay:Receive Interrupt Delay (array of int)
param: RxAbsIntDelay:Receive Absolute Interrupt Delay (array of int)
param: InterruptThrottlesRate:Interrupt Throttling Rate (array of int)
param: SmartPowerDownEnable:Enable PHY smart power down (array of int)
param: copybreak:Maximum size of packet that is copied to a new buffer on receive (uint)
[.john@acerofspades ~]$
```



Подпишитесь сейчас и сэкономьте 10%

\* Действует до 30.12.2017

## Учитесь с Linux Foundation

Подпишитесь на любой учебный курс Linux Foundation сегодня и получите скидку 10%\* от обычной цены! Эксперты из Linux Foundation создали курсы, охватывающие весь спектр Linux: от классических курсов Sysadmin до новых технологий, таких как OpenStack и Open Daylight.

Если вам понравился этот небольшой образчик и вы хотите узнать больше — мы взяли эту часть из очень популярного курса *LFS201 Essentials*

*of System Administration*, который дает дополнительную квалификацию Linux Foundation Certified Sysadmin. Курс полностью интерактивен, самостоятелен и имеет более 80 экспертных лабораторных работ с решениями для вас для тренировки ваших новых навыков.

Чтобы получить 10-процентную скидку на ЛЮБОЙ курс, выполните следующие действия: зайдите на <https://training.linuxfoundation.org/linux-courses>

и просмотрите весь спектр имеющихся курсов.

Выберите тот, который вас интересует, например, популярный *LFS201 Essentials of System Administration*.

Наконец, нажмите кнопку Register и создайте учетную запись, заодно введя код купона\* **TECH10**, чтобы получить 10-процентную скидку. И вы уже студент!

- » **rhgb** Для графической загрузки (устарело, заменено Plymouth)
- » **quiet** Отключает большинство сообщений журнала
- » **LANG** Язык системы

По соглашению, намеренно скрытых или секретных параметров быть не должно. Все они должны быть объяснены в документации, а исправления к исходному коду ядра с новыми параметрами всегда должны включать исправления в файл документации.

Интерфейс `sysctl` можно использовать для чтения и настройки параметров ядра во время выполнения. Текущие значения могут отображаться при помощи

```
$ sysctl -a
```

Их довольно много, поэтому мы обрезали этот вывод, чтобы показать только избранное:

```
abi.vsyscall32 = 1
kernel.ctrl-alt-del = 0
kernel.pid_max = 32768
kernel.threads-max = 62409
net.ipv4.ip_default_ttl = 64
net.ipv4.ip_forward = 0
vm.nr_hugepages = 16
vm.swappiness = 10
```

Здесь каждое значение соответствует псевдофайлу, находящемуся в `/proc/sys`, при этом косые черты каталога заменяются точками. Например, эти две команды эквивалентны:

```
$ sudo sh -c 'echo 1 > /proc/sys/net/ipv4/ip_forward'
$ sudo sysctl net.ipv4.ip_forward=1
```

Вторая команда не будет работать, если вокруг знака равенства есть пробелы. Первая команда сложнее, чем простое `sudo echo 1>...`, поскольку часть переадресации (`>`) не управляется `sudo`, это приведет к ошибке отказа в доступе. Поэтому надо либо вызвать другую оболочку от имени `root` и передать команду ей (как в первой команде), либо просто запускать часть в кавычках с правами `root`.

Просмотр псевдофайлов в `/proc/sys` отобразит ту же информацию, что и `sysctl -a`. Можно получить полную информацию о том, как использовать `sysctl`, выполнив `man 8 sysctl`. Для получения информации об использовании функции `sysctl()` из программ, для выполнения тех же операций, выполните `man 2 sysctl`. Если параметры находятся в `/etc/sysctl.conf` (подробнее — см. `man sysctl.conf`), их можно исправить во время загрузки. Изменения в этом файле можно принять, выполнив

```
$ sudo sysctl -p
```

— это означает, что введенные изменения будут приняты немедленно. В дистрибутивах на основе `systemd` параметры считываются из `/usr/lib/sysctl.d/00-system`, но исходный файл по-прежнему поддерживается.

## Модули ядра

Ядро Linux широко использует модули, содержащие важное программное обеспечение, которое после запуска системы загружается и выгружается по мере надобности. Многие модули включают драйверы устройств. Имеются модули, управляющие сетевыми

```
### BEGIN /etc/grub.d/10_linux ###
menuentry 'Arch Linux' --class arch --class gnu-linux --class gnu --class os $menuentry_id_option 'gnulinux-simple 18372a0b-0db1-4ff2-9f8c-f33644fce0f8' {
    insmod gzio
    set root='lvmlid/R1d171-bv1t1-QV3t1-IX1-V-ufHY-VnVv-0z2NaC/Bdaub3-0ta8-0epm-LUD9-3nYy-CuDL-6V5Dd8'
    if ! $feature_platform_search_hint || ! $feature_platform_search || ! $feature_platform_search_hint || ! $feature_platform_search || ! $feature_platform_search; then
        search --no-floppy --fs-uuid --set-root --hint='lvmlid/R1d171-bv1t1-QV3t1-IX1-V-ufHY-VnVv-0z2NaC/Bdaub3-0ta8-0epm-LUD9-3nYy-CuDL-6V5Dd8' 18372a0b-0db1-4ff2-9f8c-f33644fce0f8
    else
        search --no floppy --fs uuid --set-root 18372a0b-0db1-4ff2-9f8c-f33644fce0f8
    fi
    echo 'Loading Linux linux ...'
    linux /boot/vmlinuz-linux root=/dev/mapper/mint-vq-arch-ru quiet init=/usr/lib/systemd/systemd-bootchart
    echo 'Loading initial ramdisk ...'
    initrd /boot/intel-ucode.img /boot/initramfs-linux.img
}
submenu 'Advanced options for Arch Linux' $menuentry_id_option 'gnulinux-advanced-18372a0b-0db1-4ff2-9f8c-f33644fce0f8' {
    menuentry 'Arch Linux, with Linux linux' --class arch --class gnu-linux --class gnu --class os $menuentry_id_option 'gnulinux-linux-advanced-18372a0b-0db1-4ff2-9f8c-f33644fce0f8' {
```

» **Файлы настройки Grub2** в наши дни склонны к беспорядочности, зато генерируются автоматически, и пользователям не надо пачкать руки. Виделенная строка загружает ядро.

протоколами, поддерживающие разные типы файловой системы и выполняющие многие другие задачи. Параметры могут быть указаны при загрузке модулей для контроля их поведения. В итоге получается большая степень гибкости и быстроты реагирования на меняющиеся условия и потребности.

Многие объекты в ядре Linux предназначены для встраивания в ядро при первой загрузке или для добавления (или удаления) позже в виде модулей по мере необходимости. На самом деле, все они, кроме большинства центральных компонентов ядра, интегрированы таким образом. Даже в тех случаях, когда функциональность необходима практически всегда, включение возможности загрузки и выгрузки в качестве модуля поощряет разработку, потому что для проверки изменений перезагрузка ядра не требуется. Можно использовать команду `insmod` для непосредственной загрузки модуля:

```
$ sudo /sbin/insmod /path/to/module_name.ko
```

Модули ядра хранятся в `/lib/modules/<kernel-version>` и имеют расширение файла `.ko`. Модули зависят от версии и должны соответствовать запущенному ядру, иначе они не загрузятся. Команда `lsmod` отображает загруженные модули и показывает их зависимости. Модули можно удалить с помощью `rmmod`. Заметьте, что здесь не требуется указывать полный путь или расширение `.ko`.

В большинстве случаев `insmod` и `rmmod` игнорируются в пользу команды `modprobe`. Полный путь не надо указывать, потому что `modprobe` полагается на модули, которые устанавливаются в требуемом месте, обычно в `/lib/modules/${uname -r}`. Команда автоматически загружает любые другие требуемые модули. Удаление модуля осуществляется с помощью ключа `-r`, который также выгружает любые модули, использовавшиеся с указанным, если они не требуются в другом месте.

Команда `modinfo` отображает информацию о модулях ядра, включая имена их файлов, версии, поддерживаемое оборудование и любые параметры, с которыми их можно загрузить. Информация о загруженных модулях также доступна в `/sys` — дереве каталогов псевдофайловой системы. Например, чтобы узнать о `e1000` на скриншоте, можно посмотреть в `/sys/module/e1000`. **LXF**

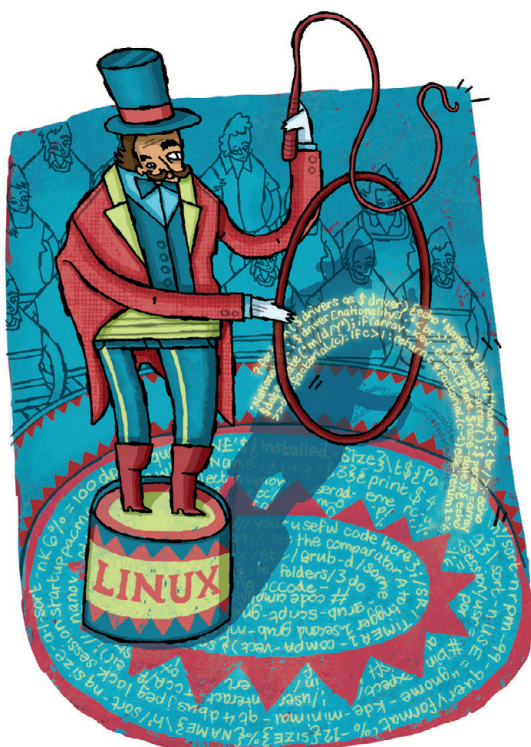
# Рабочие столы: Достичь власти

Присоединитесь к озадаченному **Джонни Бидвеллу** — он показывает, как делается рабочий стол Linux без устарелого подхода к GUI образца XX в.



**Наш эксперт**

**Джонни Бидвелл** начинает каждый рабочий день с чашки чая и нового рабочего стола. К концу дня он заканчивает настройку по своей прихоти и готов к завтрашней работе.



```
Activities
[lonni@acerofspades .themes]$ pacman -Ss conky
extra/conky 1.10.6-2
  Lightweight system monitor for X
community/conky-manager 2.4.2
  GUI for managing Conky config files with options to browse and edit themes
[lonni@acerofspades .themes]$ ls -l Unity7-master/
total 20
drwxr-xr-x 4 lonni lonni 4096 May 29 19:07 cinnamon
drwxr-xr-x 2 lonni lonni 4096 May 29 19:07 gnome-shell
-rw-r--r-- 1 lonni lonni 202 May 29 19:07 index.theme
drwxr-xr-x 2 lonni lonni 4096 May 29 19:07 metacity-1
-rw-r--r 1 lonni lonni 676 May 29 19:07 README.md
[lonni@acerofspades .themes]$ ls -l Unity7-master/gnome-shell/
total 868
-rw-r--r 1 lonni lonni 7587 May 29 19:07 bfh_launcher.png
-rw-r--r 1 lonni lonni 2532 May 29 19:07 calendar-arrow-left.svg
-rw-r--r 1 lonni lonni 2531 May 29 19:07 calendar-arrow-right.svg
-rw-r--r 1 lonni lonni 3318 May 29 19:07 calendar-day-base-active.svg
-rw-r--r 1 lonni lonni 3798 May 29 19:07 calendar-day-base-focus.svg
-rw-r--r 1 lonni lonni 5742 May 29 19:07 calendar-day-with-events.svg
-rw-r--r 1 lonni lonni 4376 May 29 19:07 calendar-today-focus.svg
-rw-r--r 1 lonni lonni 5486 May 29 19:07 calendar-today.svg
-rw-r--r 1 lonni lonni 4384 May 29 19:07 calendar_today_with_events.svg
-rw-r--r 1 lonni lonni 10940 May 29 19:07 checkbox-focused.svg
-rw-r--r 1 lonni lonni 7115 May 29 19:07 checkbox-off-focused.svg
-rw-r--r 1 lonni lonni 7839 May 29 19:07 checkbox-off1.svg
-rw-r--r 1 lonni lonni 10479 May 29 19:07 checkbox.svg
-rw-r--r 1 lonni lonni 4424 May 29 19:07 close.svg
-rw-r--r 1 lonni lonni 8418 May 29 19:07 close_window.svg
```

» Когда вы выберете рабочий стол, рекомендуем добавить выпадающий терминал. На Gnome это *Guake*, но пользователи Plasma могут предпочесть *Yakuake*.

существуют сегодня, и справедливо будет сказать, что они по-прежнему являются самыми крупными рабочими столами, но конкуренция сейчас жестче и сильнее, чем когда-либо. Многие новые дружелюбные к новичкам дистрибутивы до известной степени (будь то хорошо или плохо) определяют их рабочими столами.

Ubuntu (настольный) и Unity стали неразрывно связаны, потому что новички куда больше видят эту красно-оранжевую боковую панель, чем лежащую в основе структуру файловой системы. Mint предлагает более традиционный Cinnamon; Bodhi — свой просветительский Moksha; elementaryOS — похожий на macOS Pantheon, Solus — Budgie... честно говоря, список можно продолжать. Союз Ubuntu–Unity был уникален, потому что последний практически невозможно установить на другие дистрибутивы (из-за очень сложных правок в библиотеках Gnome со стороны Ubuntu), хотя спрос на него был в общем велик.

## Стряпаем компот

Другие упомянутые рабочие столы более или менее отделены от своих родительских дистрибутивов, и желающие с ними познакомиться не обязательно должны устанавливать себе новый Linux. Например, если вы хотите попробовать Cinnamon и целую кучу предлагаемых им прясней рабочего стола (Cinnamon — *англ.* корица) на Ubuntu, вам надо всего лишь сделать

```
$ sudo add-apt-repository ppa:embrosyn/cinnamon
$ sudo apt update && sudo apt install cinnamon
```

Конечно, Gnome, KDE Plasma, Xfce и многие другие рабочие столы с самого первого дня были независимы от дистрибутивов, хотя дистрибутивы по-прежнему подгоняют свою работу под их потребности. Как следствие — Gnome, который вы получаете на Ubuntu,

**К** настоящему моменту большинство людей здоровых уже перестали толковать о «где Linux на рабочем столе». Когда-то, примерно во время выпуска Windows Vista, была надежда на то, что гиганты Windows и Mac падут. Но инерцию ОС трудно преодолеть, люди увязли в дурных проприетарных привычках, и сейчас это песнопение стало достоянием сказок.

Не то чтобы это имело большое значение — экосистема рабочего стола Linux никогда не была более прочной; просто это скорее область особого интереса, а не основной игрок. По мере того, как повседневные компьютерные технологии всё больше перемещаются в царство смартфонов и планшетов, изменяется вся парадигма рабочего стола. Да, офисному планктону, как и раньше, нужны их настольные Windows и Mac, чтобы создавать отчеты и раскладывать пасьянсы, но традиционная идея семейного домашнего компьютера и всех этих нудных ОС, на которых он когда-то работал, подходит к концу.

С конца 1990-х и примерно до 2010 г. основных рабочих столов Linux было два: Gnome и KDE. Это замечание наверняка вызовет переполох среди многих читателей. А как же Enlightenment? DWM? CDE?! Да, они тоже существовали, и до сих пор существуют, но были и остаются в меньшинстве. Gnome и KDE по-прежнему

## Скорая помощь



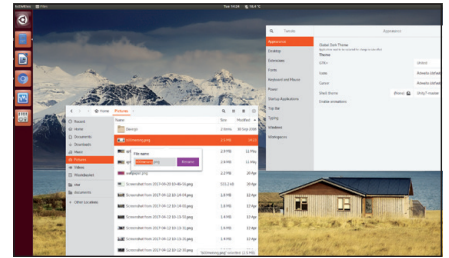
Какой бы рабочий стол вы ни выбрали, *Conky* везде будет отличным дополнением. Тема *Harmattan* приглаживает файл настройки, что дает в результате красивый и информативный рабочий стол.

## Причешем Gnome (более) под Unity

Фанатам Unity [Ред.: — Это у нас кто?!], удрученным переходом на Gnome в следующем релизе Ubuntu, не о чем волноваться. Во-первых, Unity 7 останется доступен в репозиториях. Кроме того, нарождающийся Unity 8 усыновлен сообществом и постепенно будет развиваться. Однако наличие рабочего стола Gnome, смутно напоминающего прежнее прибрежье, способно смягчить боль перехода в этот дивный новый мир рабочих столов. Мы уже упоминали популярное расширение Dash-to-Dock, обеспечивающее док, которое новым пользователям Gnome, вероятно, пригодится. Его можно настроить, или из *Gnome Tweak Tool*, или через сайт расширений, чтобы сделать всё более похожим на Unity. Оно может применить характерное размещение с левой стороны рабочего стола. Dash-to-Dock умеет

включать меню Applications из вида Activities, и его можно разместить сверху, чтобы он работал как HUD бедняка. Настройки автоскрывания также настраиваются по вашему вкусу. Установка их в режим Panel расширит док по длине/ширине экрана, прямо как панель Unity. Кнопки управления окном слева в Unity спорны, но их можно воссоздать в Gnome, установив *dconf-editor*, перейдя в `org.gnome.desktop.wm.preferences` и изменив значение раскладки кнопок `button-layout` на `close,minimize,maximize`: [закрывать, свернуть, развернуть:] (обратите внимание на двоеточие). Тема *GTK Ubuntu Ambiance* сейчас в процессе настройки для Gnome, и она может добавить в ваш Gnome еще немного стиля Unity. Для тех, кому нужен еще более Unity-подобный рабочий стол, есть тема для Gnome Shell от *b00merang*

(<https://github.com/B00merang-Project/unity7>) — работает со всем здесь упомянутым и с темой Ubuntu icon ради создания имитации, как на рисунке.



➤ Тема Unity 7 от *b00merang* похожа на Unity, насколько это возможно в Gnome.

выполняя (но делайте это только в том случае, если вы не против попортить некоторые свои настройки Unity)

```
$ sudo apt-get install ubuntu-gnome-desktop
```

отличается от более скромного опыта, который вы приобретете, выполнив следующее в Arch:

```
# pacman -S gnome
```

То, что люди привыкли к некой разновидности каскадного меню приложений внизу слева — изобретению, пришедшему из Windows 95 — не означает, что это лучший вариант рабочего стола. На самом деле, большинство критики рабочих столов происходит из-за нежелания людей пробовать что-то новое, а не из-за новых технологий. Это странно, ведь большинству пользователей Linux следовало бы помнить свой уход от Windows и отказ от старых привычек.

## Пересмотрите ваш рабочий стол

Существует общее недовольство насчет многих современных сред рабочего стола. Пользователи критикуют то, что размещение документов и ярлыков на рабочем столе (другое изобретение Windows 95) уже не одобряется. На самом деле, это не дает простого доступа к объектам, поскольку эти объекты в конечном счете будут выполняться приложениями.

Тем не менее, большинство рабочих сред можно заставить разрешить заполнить рабочий стол, и если вам действительно нужны такие функции, то они там есть. Далее, бытует мнение, что для оптимальной работы вам придется изучать загадочные комбинации клавиш. Конечно, определенные сочетания клавиш есть, но вам совершенно не обязательно учить те, которые вам не помогут, и большинство из них не особенно загадочные. Вообще-то большинство из них (Alt+Tab, Alt+F4) такие же, как в некой неназываемой ОС, которую, судя по всему, некоторые скряги покушаются эмулировать в свой Linux.

Новые рабочие столы могут обеспечить хорошие новые способы работы, переписав старые правила рабочих столов. Первопроходцем здесь является *Pantheon*, рабочий стол, поставляемый с *elementary OS*. К примеру, встроенные приложения *Pantheon* сохраняют свое состояние после закрытия, так что вы можете смело нажимать на кнопку закрытия и смотреть, как исчезает ваша работа, будучи уверенными, что всё (документы, структура, вообще всё) будет восстановлено в следующий раз. Таким образом, закрытие приложения и открытие его заново становится всё больше похоже на минимизацию и максимизацию. Это отчасти оправдывает то, что *Pantheon* отказался от кнопки минимизации. От нее также отказывается Gnome, но совершенно по иной причине. И кстати о Gnome...

## Gnome в строю

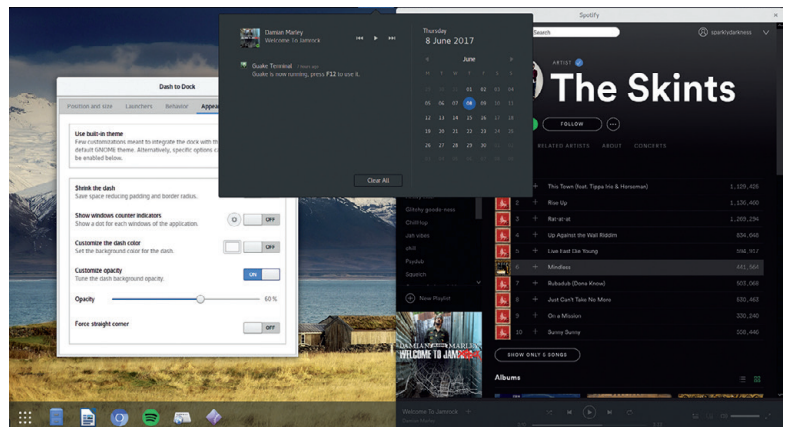
[Аллюзия на известную американскую песню “Home on the range”, — прим. пер.]

Что касается непопулярных технологий, проникших в экосистему Linux — и, судя по всему, без приглашения — мало о чем (кроме очевидного исключения в виде *systemd*) говорят так часто, как о Gnome 3. Этот рабочий стол избавился от таких «мелочей», как меню приложений, кнопки минимизации, системные лотки и панели задач. И в 2010 г. широким массам не очень-то понравился его внешний вид.

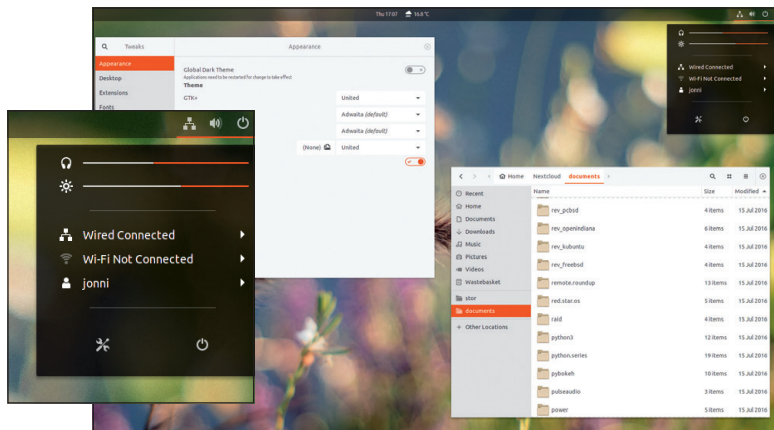
Рабочий стол *Mate* вырос из останков его предшественника, *Gnome 2*, и с тех пор приобрел значительную пользовательскую базу. Он не был готов к Ubuntu 11.04, но даже когда он стал готов, Ubuntu, ссылаясь на разногласия с командой Gnome, осталась со своим собственным рабочим столом *Unity*.

В зависимости от того, кого вы спросите, при выпуске *Unity* был таким же непопулярным, как *Gnome 3* (оба подвергались критике за то, что их интерфейсы были в первую очередь ориентированы на мобильные устройства и лишь во вторую — на рабочие столы, хотя мобильный Linux не был и до сих пор не является чем-то серьезным), но постепенно приобрел верных поклонников. А в апреле этого года было объявлено о прекращении его работы: следующий *Ubuntu* снова перейдет на *Gnome* (теперь это настройка по умолчанию в сборке 17.10), и круг замкнулся.

*Gnome 3* использует менеджер окон *Mutter* (слияние *Metacity*, бывшего менеджера окон *Gnome*, и *Clutter*) и композитор *Wayland*. Обращенные к пользователю части рабочего стола, *Gnome Shell*, ➤



➤ Расширение *Dash-to-Dock* позволит некоторым пользователям чувствовать себя больше как дома. Прямо как Джонни, когда он слушает *Welcome to Jamrock*.



➤ Тема **United** выводит на стол некоторые элементы в стиле **Unity**, в том числе более симпатичные кнопки управления аудио, как показано здесь.

реализованы как плагины для *Mutter*. А *Mutter*, строго говоря, не вполне компонент *GNOME* и может применяться в других рабочих столах. *Muffin*, ответвление *Mutter*, используется в *Cinnamon*.

## Что нам в GNOME

Среда рабочего стола состоит из множества уровней и компонентов, и иногда не совсем понятно, где заканчивается рабочий стол и начинается стек приложений (по восходящей) или стек отображения/системы (по нисходящей). В основном *GNOME* подвергается критике за якобы зависимость от *systemd*, особенно от компонента *systemd-logind*. Функционально это не должно беспокоить большинство пользователей рабочего стола, поскольку большинство основных дистрибутивов уже испили *Kool Aid* [англ. «крутое сое-действие» — порошок для приготовления прохладительных напитков, — прим. пер.] от *systemd*.

Попытки *Gentoo Linux* использовать свободный от *systemd* *GNOME* завершились кучей ошибок, так что теперь *systemd* стал там зависимостью *GNOME*. Вам не обязательно делать *systemd* своей системой *init*, но в *Gentoo* он должен присутствовать, если вы хотите использовать *GNOME* без сторонних наложений (см. [https://wiki.gentoo.org/wiki/GNOME/GNOME\\_Without\\_systemd](https://wiki.gentoo.org/wiki/GNOME/GNOME_Without_systemd)). *PCLinuxOS* не предлагает *GNOME 3*, а вот *Devuan* (свободная от *systemd* ветка *Debian*, <https://dev-1.org>) хотя и предлагает пакет *GNOME*, но допускает шансы сбоев. *Devuan* покамест достиг версии 1.0.0, и эти сбои в скором времени, надо надеяться, будут исправлены. Справедливости ради следует отметить, что их пакеты *KDE* и *Cinnamon* тоже грешат ошибками.

Однако мы считаем, что *GNOME* надо дать шанс. Возможно, когда он появился, он и был ужасен, но у него было время созреть, и если его критики оставят свои предубеждения относительно

того, каким должен быть рабочий стол, то, возможно, тоже получат от него удовольствие.

Первая жалоба заключается в том, что перемещение к верхнему левому углу для вызова *Activities* долгое и трудозатратное. Вторая заключается в том, что это перемещение нужно производить не только когда надо открыть программу, но и при смене приложений. Но вид *Activities* можно вызвать, нажав на клавишу *Super* (она же *Windows*), и вместо того, чтобы рыскать в меню *Show Applications*, чтобы найти определенную программу, просто введите первые буквы ее имени.

Переключаться между приложениями лучше всего с помощью *Alt+Tab*. Если вы используете многооконные приложения, то вас переключит *Alt+`* (обычно это клавиша над *Tab*). Нам кажется, что после привыкания к такому употреблению клавиатуры работа в *GNOME* станет для многих куда лучше. Некоторые предпочитают использовать *Alt+Tab* иначе, и им стоит обратить внимание на такое расширение, как *AlternateTab*, которое не группирует окна по приложениям.

Работа с рабочими областями (ранее — виртуальными рабочими столами) в *GNOME* ведется не так, как раньше. Вместо фиксированного их числа, они распределяются динамически, и всегда есть пустая, готовая к тому, чтобы ее заполнили приложения. Это расстроит тех, кто любит заполнять конкретные рабочие области конкретными приложениями по мере их открытия, но *GNOME Tweak Tool* может восстановить здесь традиционную структуру. Как и с приложениями, эффективнее всего переключаться по рабочим областям, пользуясь клавиатурой: ваш помощник — *Ctrl+Alt+стрелка вверх/вниз*.

## GNOME там, где сердце

Когда *GNOME* только появился, его критиковали за отсутствие индивидуальной настройки. Нет возможности изменить тему и даже шрифт по умолчанию. Суть в том, что базовый *GNOME* должен подходить всем пользователям, и предложение подобной настройки будет только отвлекать пользователей. Но теперь всё это можно сделать через удобный инструмент *GNOME Tweak Tool* (`sudo apt-get install gnome-tweak-tool`).

*GNOME 3* с самого начала поддерживал расширения. Они, как и солидная доля интерфейса *GNOME Shell*, написаны на *GNOME JavaScript* (*gjs*), так что в некоторой мере это соответствует официальному способу их установки через сайт <https://extensions.gnome.org>. Чтобы всё это работало не в том браузере, который предоставлен самим *GNOME*, надо установить дополнение браузера и приложение-соединение. Ваш браузер подскажет вам установить первое, а второе можно установить из менеджера пакетов вашего дистрибутива. Он будет называться как-то наподобие *chrome-gnome-shell*.

Среди самых популярных расширений — те, которые воссоздают функции из *GNOME 2*, например, индикатор *Places* и меню

## Обилие функций GNOME

Новые функции в *GNOME* добавляются постоянно. Одна из самых последних, появившихся в *GNOME 3.24* — *Night Light*, которая настраивает цветовую температуру в зависимости от времени суток и времени года. Более теплые температуры в ночные часы могут уменьшить напряжение глаз и усталость.

Эта функция была доступна через программу освещения *RedShift*, но приятно видеть, что сейчас это встроено в *GNOME*. Особенно поскольку *RedShift* использует устаревший значок в системном лотке, который не соответствует видению

*GNOME*. В *GNOME 3* также имеется встроенная программа записи экрана — просто нажмите *Ctrl+Alt+Shift+R*, и в верхнем правом углу появится красная точка, которая обозначит, что ведется запись. Используйте ту же комбинацию клавиш, чтобы остановить запись, и получившийся в результате шедевр кинематографии будет сохранен в директории `~/Videos`.

Поставщики высококачественных машин *Linux System76* активно трудились над новой темой *GNOME*, готовясь к переходу *Ubuntu* с *Unity*. Она называется *Pop* и является ответвлением,

вдохновленным материальным дизайном темы *Adapta*. Это вовсе не попытка заставить *GNOME* выглядеть, как *Unity* (см. врезку на стр. 69, чтобы узнать, как это сделать), хотя и предлагает симпатичные красные и оранжевые тона (цвета *System76* и *Unity*). Ее можно установить в *Ubuntu* с помощью

```
$ sudo add-apt-repository ppa:system76-dev/stable
$ sudo apt update
$ sudo apt install system76-pop-theme
```

А чтобы установить ее в других системах, следуйте инструкциям на <https://github.com/system76/pop-theme>.



## Припасть к чистому KDE neon

Разработка KDE ведется достаточно быстро — иногда настолько, что дистрибутивы не успевают за ней. Например, Plasma 5.10 вышел в конце мая, а пакет KDE Applications будет обновлен в августе, но вышедший [upstream] релиз нуждается в тщательно отобранных пакетах, протестированных каждым дистрибутивом, и нередко (скажем, если требуется обновление каких-то фундаментальных частей *Qt*) они не будут включаться до следующего релиза.

Подобные связанные с релизами огорчения весьма обычны, и хотя их можно исцелить использованием сторонних репозиториях (таких как PPA Kubuntu Backports), результаты бывают нестабильными.

По этой причине KDE поддерживает свою систему, основанную на релизе Ubuntu LTS и предлагающую самый последний стек KDE под названием KDE neon. Назвать его дистрибутивом было бы неправильно, потому что его репозитории

не содержат ничего, кроме программ KDE. Чем бы он ни был — это, вероятно, лучший способ использовать самые последние новоиспеченные технологии KDE.

Многие предпочитают KDE neon привычным дистрибутивам KDE (например, Kubuntu), не только из-за более новых компонентов, но из-за его принципа «KDE, предназначенный для KDE». Скачайте его с <https://neon.kde.org> и попробуйте разобраться, по поводу чего поднялся такой шум.

Application, сгруппированное по категориям. Еще одно — Dash-to-Dock, которое перемещает Dash из Activities Overview в собственный док, чтобы открытые приложения были доступны внизу экрана (или еще из какого-нибудь экстремального места по вашему усмотрению).

Еще одна приятная вещь — возможность устанавливать пользовательские темы через *Tweak Tool*, и для этого понадобится расширение User Themes, которое вы найдете в верхней части списка на сайте расширений. С его помощью можно добавлять сторонние темы, просто распаковывая их в папку `~/.themes`.

## Игра с цифрами KDE

KDE прошел болезненный путь реформации при переходе с KDE 3 на KDE 4 и последующей модернизации последнего. Косметически последний релиз под номером 5 оказался похож на своего предшественника, только быстрее и утонченнее. В результате довольно мало кто требовал, чтобы всё было «так же, как раньше» в KDE 4. Однако некоторые и правда скучали по KDE 3, до такой степени, что появилось его ответвление под названием Trinity. Однако за внешностью крылись солидные изменения, такие, что стало неправильно называть его KDE 5 (см. [LXF205/206](#), Plasma 5). Помимо этого, Plasma 5 является прекрасным примером того, как традиционную метафору рабочего стола можно привести в соответствие с современностью.

Будучи когда-то раскритикованным как пожиратель ресурсов, рабочий стол Plasma сейчас использует куда меньше памяти, чем Gnome, и благодаря использованию аппаратного ускорения *Qt* более быстр и отзывчив — при наличии соответствующего (то есть не старше десяти лет) GPU.

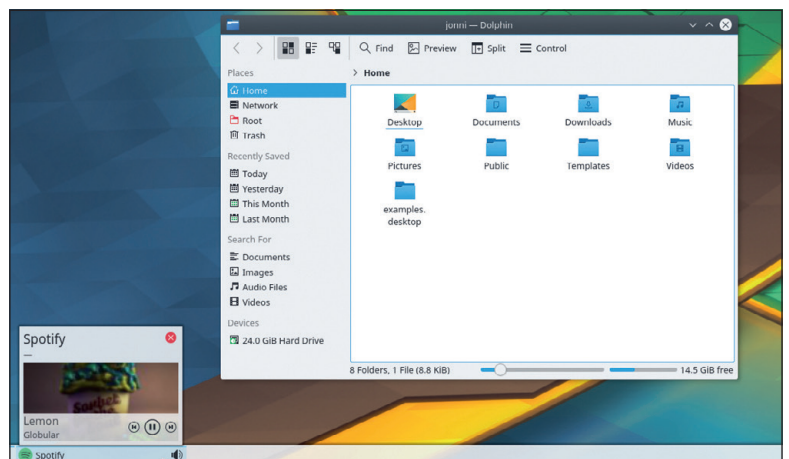
Сессию Plasma 5 можно добавить в Ubuntu с помощью обычного

```
$ sudo apt-get install kubuntu-desktop
```

Как и с пакетом *ubuntu-gnome-desktop*, упомянутым ранее, установится полный пакет рабочего стола, как в Kubuntu, который при тестировании на Ubuntu 17.04 весил 1,5 Гб. Он также предложит вам на выбор использование *sddm*, работающего на *Qt*, или традиционного менеджера входа в систему *lightdm*. Если вы используете более старую версию Ubuntu, то вы можете использовать репозитории PPA обратных портов Kubuntu для получения более современных, но не так хорошо протестированных пакетов KDE. Этот архив включает Plasma 5.8 для 16.04 и 16.10, а также Plasma 5.10 для 17.04.

## GTK+ или не GTK+

Среда рабочего стола частично определяется набором инструментов виджета, на котором основаны его приложения. Plasma использует *Qt* (*Qt*, по сути, целая платформа для разработки приложений), а Gnome использует *GTK+ 3*. Частичная причина длительного отсутствия новых функций у рабочего стола *Xfce* в том, что титанические усилия портировать все его инструменты из *GTK+2* в *3* наконец-то пришли к завершению.



► Plasma 5 предлагает интерактивный предпросмотр окон, чтобы вы могли управлять своим аудиоплеером (здесь — Spotify) из комфортной среды дока.

Рабочий стол *LXDE*, некогда бывший частью *Raspbian* и составивший там основу нового рабочего стола *Pixel*, прекращает свое существование. *LXDE* основан на старой библиотеке *GTK+ 2*; в 2013 г. его автор решил, что *GTK+ 3* ему не подходит, и в том же 2013 г. начал работу по портированию *LXDE* на *Qt*. В тот момент был еще один проект, *Razor-Qt*, целью которого было создание легковесного рабочего стола на *Qt*, и мы увидели крайне редкий случай объединения сил проектов Linux, который привел к рождению *LXQt*. Рабочий стол *LXDE* по-прежнему используется в *Lubuntu*, но его дни сочтены.

Дистрибутив *Solus* наделал шуму со своим рабочим столом *Budgie* на базе Gnome, также доступном и в других дистрибутивах. Ubuntu 17.04 даже увидел добавление новой разновидности *Budgie*. Однако не отставать от Gnome — задача нелегкая, и следующий релиз *Budgie* будет основан на *Qt*. Узнайте всё об этом на <https://budgiesdesktop.org/2017/01/25/kicking-off-budgie-11>.

Переход на *Budgie* [Пед.: — Я притворюсь, что этого не читал] говорит о растущей озабоченности тем, что *GTK+* и связанные с ним технологии всё больше погружаются в Gnome. Это усложняет задачу создания рабочего стола на этих технологиях без имитации или использования неких частей стека Gnome. Поскольку эти компоненты становятся всё более «гномистыми», попытка применить их где-нибудь еще становится стрельбой по движущейся цели.

*Budgie* отказался от этой погони, и встают вопросы о будущем Linux Mint и его рабочего стола Cinnamon. Ответвление самых общих приложений Gnome (для создания X-Apps) является значительным шагом в направлении свободы, но в долгосрочной перспективе, в зависимости от того, как будет развиваться эта тенденция, может возникнуть необходимость в ответвлениях более фундаментальных библиотек и компонентов. Мы будем держать вас в курсе. **LXF**

# Arch Linux: Дистристрой

Не ждите, пока грянет катастрофа. **Марк Чизхолм** учит, как сделать собственный дистрибутив на основе Arch Linux для спасения ваших данных.



**Наш эксперт**

**Марк Чизхолм**  
Arch держит Марка на переднем крае мира Linux. Он поддерживает Morpheus Arch Linux, который специализируется на восстановлении данных.



**Д**жадд Винет [Judd Vinet], создатель Arch Linux, классно сказал, что Arch Linux — это то, что вы из него делаете. Имея это в виду, мы считаем, что Arch Linux — идеальная основа для начала сборки собственного дистрибутива.

На нашем уроке основное внимание будет уделено восстановлению данных, поскольку наличие диска восстановления полезно как среднему домашнему пользователю, так и опытному сисадмину, и ничто не сравнится с приятным сознанием того, что такой диск у вас есть. Попутно мы покажем, как настроить его дальше, чтобы выполнить практически любую задачу, включая тестирование на вторжения [pen test].

Arch Linux, может быть, менее настраиваемый, чем Gentoo, где исходный код нередко подстраивается к оборудованию, но это относительно простой способ создания собственного образа диска Arch Linux.

Основная часть этой статьи будет связана с использованием пакета *archiso*. Именно этот инструмент применяется для создания официальных образов дисков Arch Linux, которые вы можете

получить на сайте [www.archlinux.org/download](http://www.archlinux.org/download). Но по использованию *archiso* имеется ряд предостережений. Требуется платформа x86\_64 (32-разрядная версия будет упразднена в течение 2017 г.). Второе условие — всё надо делать от имени root. Не часто здесь, в Учебниках LXF, можно услышать предложение всё делать как root. Однако невыполнение этого требования приведет к установке ложных разрешений, и всё будет работать не так, как предполагалось: всё, от отказа загрузки во вновь созданный ISO до невозможности войти в систему под учетной записью пользователя. Далее мы немного больше поговорим о разрешениях и о том, как их правильно установить.

Сначала вам понадобится рабочая установка Arch Linux. Если вы это делаете впервые, рекомендуем виртуальную машину. Для этой задачи подойдет *Qemu* или *VirtualBox*, в зависимости от того, насколько строго вы придерживаетесь свободы ПО в вашей системе. Система плавающего релиза Arch Linux быстро развивается, с частыми обновлениями, и мы рекомендуем потратить несколько минут на чтение руководства по установке.

При настройке вашей виртуальной машины вам придется выделить пространство для свеже созданного ISO, поэтому тщательно оцените объемы пакетов, которые будут установлены в системе. Если ваша система позволяет использовать дополнительные ядра процессора, рекомендуем дать ей столько, сколько вам удобно.

Предоставление виртуальной машине большего количества ядер процессора будет означать, что для собственно вывода файла образа диска потребуются меньше времени. Для этой задачи лучше избегать чего-нибудь вроде установки полноценного KDE из-за его производительности в некоторых виртуальных средах (хотя он улучшает обработку этой задачи) и места, которое потребуются для ISO.

## Получение ArchISO

*Archiso* — это набор скриптов оболочки *Bash*, которые выглядят довольно примитивными. Как может нечто настолько маленькое дать возможность создать себе дистрибутив Arch Linux, соответствующий вашим потребностям? Но, в духе Arch Linux, это действительно всё необходимое, чтобы придерживаться философии Arch «будь проще».

Основной частью *archiso* является *mkarchiso*. После установки виртуальной машины Arch Linux первым делом надо открыть терминал и ввести следующую команду от имени root:

```
# pacman -S archiso
```

*Archiso* поставляется с двумя профилями. Профилем *releng* мы воспользуемся для настройки списка установленных пакетов, а базовый профиль предназначен для мелкой настройки

## Скорая помощь

Быстрый способ установить работающую среду Arch Linux — посмотреть на другие дистрибутивы на основе Arch, содержащие графический установщик, вроде Arch-Anywhere (<https://arch-anywhere.org>).

без включенных в него пакетов. Затем надо создать новый каталог, чтобы начать настройку нашего нового дистрибутива Arch Linux. Поэтому снова запустите терминал или просто введите эту команду, если терминал у вас еще открыт:

```
# mkdir ~/archive
```

Создастся каталог **archive**. Нам он нужен, чтобы мы могли скопировать профиль и начать его настройку. *Archiso* устанавливается в каталог **/usr/share/**, и следует скопировать файлы из него в **~/archive**. Для этого введем следующую команду:

```
# cp -r /usr/share/archiso/configs/releng ~/archive
```

Флаг **-r** рекурсивно копирует файлы и всё прочее из каталога **/usr/share/archiso/configs/releng**. Вкратце опишем некоторые каталоги в профиле **releng**. Каталог **airootfs** станет корневым каталогом во вновь создаваемом файле образа диска. Всё в текущей системе Arch, что вы хотите скопировать в новый образ диска, например, **/etc/dhcpd.conf**, должно быть скопировано в каталог **~/archive/releng/airootfs/etc/**.

Позже мы скопируем еще некоторые файлы. Каталог **efiboot** в профиле **releng** содержит всё необходимое для загрузки в системах с UEFI, и мы не собираемся его трогать на протяжении всего нашего урока. **Package.both** предназначен для создания файлов с двойным образом диска, но поскольку 32-разрядная поддержка прекращается, мы пока оставим в покое и это, и **packages.i686**. Будем использовать **package.x86\_64**. Если вы выберете **package.both**, то время сборки двойного индивидуального образа диска будет больше, поскольку надо подготовить файлы и для 32-, и для 64-разрядной системы. Файл **pacman.conf** тоже оставим как есть.

Всё это происходит в **build.sh**, поэтому не стесняйтесь заглянуть в него, по команде

```
# nano ~/archive/releng/build.sh
```

Можно использовать и *vim*, если при виде *nano* вас бросает в дрожь. Просто будьте осторожны, чтобы ничего не изменить в файле **build.sh**, иначе добра не жди. Всё, что мы здесь делаем, не повредит имеющейся у вас установке Arch Linux и на самом деле безопаснее, чем кажется. В худшем случае — ну, получится файл образа диска, который не сможет загрузиться.

## Настройка дистрибутива

Теперь начнем добавлять пакеты в систему Arch. Затем мы можем скопировать их в список **packages.x86\_64**. Хотя можно войти и отредактировать их позже, мы бы этого не советовали, если вы не твердо знаете, какой пакет является зависимостью для другого пакета. Наблюдательный читатель заметит, что когда мы установили *archiso*, заодно установился *lynx*. Чисто из личного предпочтения мы вместо него установим *elinks*. А вы можете установить хоть оба, поскольку они легковесны и это слабо отразится на итоговом объеме образа. Итак, введите команду

➤ На скриншоте показан скрипт **build.sh**, содержащийся внутри **/root/archive/releng/**. Не меняйте скрипт, иначе сборка может рухнуть!

```
# pacman -S elinks
```

Теперь вы и вправду можете начать установку пакетов по своему выбору, но поскольку огород мы городили ради восстановления данных, надо будет установить некоторые пакеты для этой цели, такие как *photorec*, *testdisk* и *ddrescue*.

Можно установить все три пакета сразу, одной командой:

```
# pacman -S testdisk ddrescue
```

Если вам интересно, почему здесь фигурирует только два пакета, то дело в том, что *photorec* входит в пакет *testdisk*. *Testdisk* — это GPL-пакет, который помогает при восстановлении разделов, делает незагружаемые устройства снова загружаемыми, восстанавливая таблицы разделов, удаленных аппаратным сбоем или ошибкой человека.

*Photorec*, несмотря на «как бы намекающее» имя, способен, кроме фото, восстанавливать более 400 типов файлов. Вы можете узнать, какие типы поддерживаются *Photorec*, посетив [www.cgsecurity.org/wiki/File\\_Formats\\_Recovered\\_By\\_PhotoRec](http://www.cgsecurity.org/wiki/File_Formats_Recovered_By_PhotoRec). Пакет *ddrescue* многократно пытается восстановить данные, считывая диск. Он постарается исправить ошибки и сохранит файл журнала

**Скорая помощь**

Не удается установить пакеты в новый корень? Обычно это вызвано устаревшим набором ключей. Подробнее см. <http://bit.ly/2rdgZ3s>. Если это произойдет, вам потребуется также удалить рабочий каталог, чтобы перезапустить процесс сборки.

## Больше власти — больше ответственность

Есть немаловажная причина, по которой пользователи видят такое сообщение, впервые входя в *sudo*. Дело в том, что это правда. По одной лишь той причине, что вы можете создать индивидуальный образ диска Arch Linux, разместить его на SourceForge и попасть в Distrowatch, вы отнюдь не всегда должны так делать.

Надо осознавать, что люди, скачивающие ваш ISO, не будут иметь того же оборудования,

на котором вы смогли его протестировать. Посыплются сообщения об ошибках, и люди попросят помочь в использовании. Поддержка дистрибутива Linux, предложенного для загрузки широкой публике, требует времени и усилий. Некоторые дистрибутивы Linux для поддержки полностью полагаются на пожертвования.

Если вы пойдете путем загрузки на такие сайты, как SourceForge, то мы желаем вам удачи с вашим

новым созданием и прибавлением семейства Linux, но обязаны предупредить вас об этих проблемах, а также о том, что публиковаться вам следует, только если у вас есть время на поддержку и исправление ошибок или дефектов во вновь созданном образе диска. Подобная деятельность также требует серьезных знаний о внутренней работе Linux — не только дистрибутива Arch Linux, но и Linux в целом.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)

для проверки в случае необходимости. Вам также может пригодиться великолепная статья Шона Конвэя о *Photorec* в **LXF190**.

Теперь допустим, что вам нужна помощь по Arch Linux при загрузке в живую среду. Для этого понадобится клиент IRC. Есть несколько доступных, таких как *irssi* и *weechat*. Оба являются пакетами командной строки и будут работать с рабочим столом или без него. Помните, что устанавливая среду рабочего стола, вы должны иметь пакет *xorg-server* и установленную среду рабочего стола вместе со всеми зависимостями и соответствующими графическими драйверами.

Arch, как правило, может определить, какой из них использовать в зависимости от системы, но легковесности ради вы можете полностью исключить среду рабочего стола. Для установки *irssi* и *weechat* введите следующую команду:

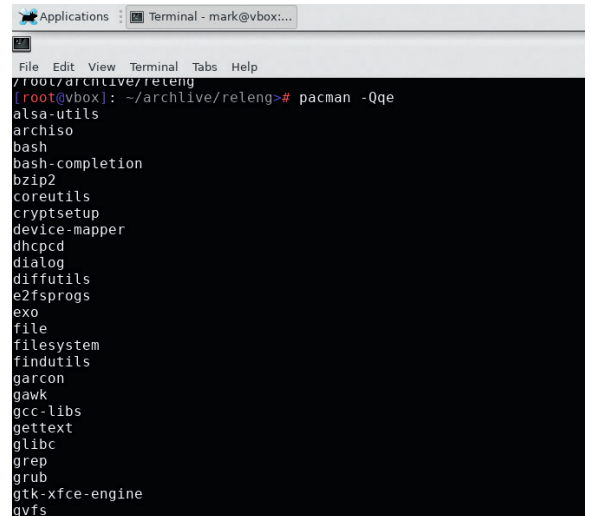
```
# pacman -S irssi weechat
```

Вероятно, со своим ISO Arch вы также захотите выполнять сетевую диагностику. Для этого можно установить *ntmap* (*zenmap* — аналог с графическим интерфейсом). *Nmap* — бесплатная утилита с открытым исходным кодом для диагностики сети и проведения аудита безопасности.

Если вы — тестировщик на вторжения, то можете даже установить ежедневно требуемые вам пакеты в собственном легковесном ISO и всегда носить его с собой на диске или загрузочном USB-накопителе. Для этого скомандуйте

```
# pacman -S nmap
```

Установите пакет *extundelete*, полезный для восстановления данных. Это инструмент терминала, который восстанавливает файловые системы ext3 и ext4. Однако, чтобы использовать *extundelete*, разделы должны быть размонтированы. Все файлы,

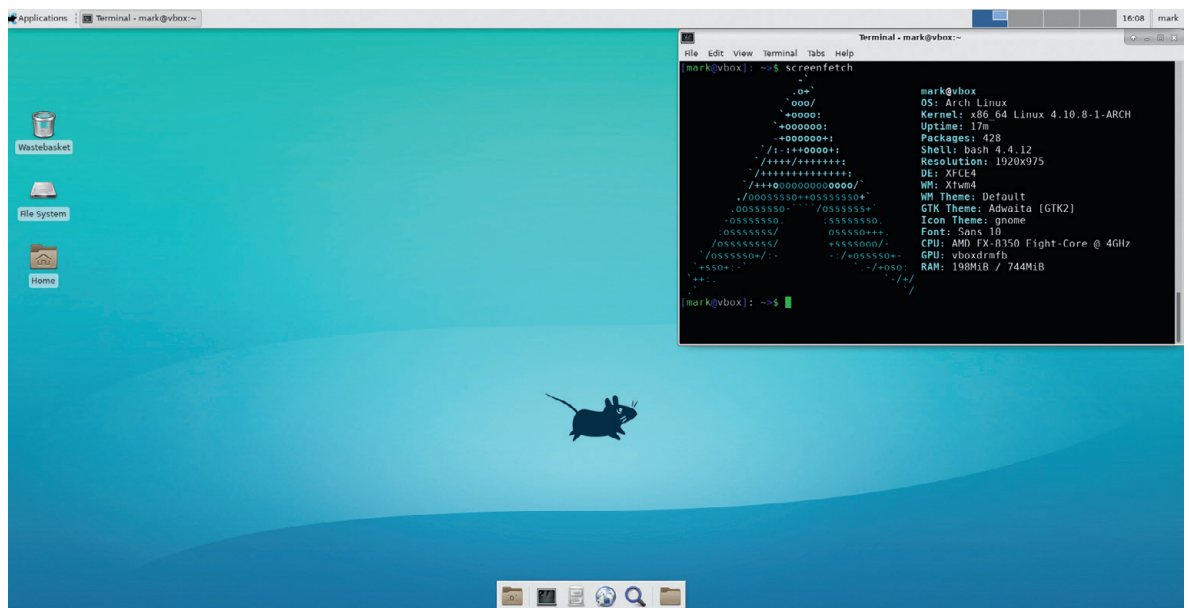


```
File Edit View Terminal Tabs Help
[root@vbox]: ~/archive/releng># pacman -Qqe
alsa-utils
archiso
bash
bash-completion
bzip2
coreutils
cryptsetup
device-mapper
dhcpcd
dialog
diffutils
e2fsprogs
exo
file
filesystem
findutils
garcon
gawk
gcc-libs
gettext
glibc
grep
grub
gtk-xfce-engine
qvsf
```

➤ Вывод `pacman -Qqe` покажет все пакеты, установленные в системе.

## Скорая помощь

Если вы хотите установить среду рабочего стола, вам также надо будет установить соответствующие графические драйверы и пакет *xorg-server*. В большинстве случаев Arch будет знать, какой графический драйвер использовать в конкретной системе.



➤ Arch весьма гибко настраивается, если вы отважитесь расстаться с настройками Xfce по умолчанию.

## Укрепите безопасность своей системы

*ClamAV* — это бесплатный многопоточный антивирусный сканер с открытым исходным кодом; в 2013 г. Cisco systems приобрела его у Sourcefire и с тех пор развивает. Его тоже лучше установить на новый ISO, на случай попытки восстановить данные после вируса. Как ни мало вероятно, что система попорчена вирусом, предустановленный *ClamAV* бывает полезен, как и некоторое другое ПО.

Другой полезной программой будет *AIDE*. Учтите, что она только выполняет проверку целостности

файла, а не наличие признаков руткитов или чего-нибудь подозрительного в логах. Для этого следует рассмотреть нечто вроде *logwatch*. Для руткитов мы, как и многие другие, рекомендуем *chkrootkit* — это инструмент, который проверяет локально наличие признаков руткита, и, к его чести, он входил в Top 100 Network security tools [Лучшие инструменты сетевой безопасности] 2006 г.

Для получения рекомендаций по обеспечению безопасности системы обдумайте возможность

использования инструмента под названием *lynis*, разработанного CISOfy, который специализируется на Linux и UNIX. Существует корпоративная версия (за нее надо платить), но и бесплатная версия неплохо определяет, насколько безопасна ваша система по шкале от 0 до 100, и анализирует результаты, позволяя вам узнать, какие шаги следует предпринять для обеспечения безопасности системы. Бесплатный авторитетный совет — разве такое бывает лишним?

➤ Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

восстановленные *extundelete*, помещаются в рабочий каталог в папке с именем **RECOVERED\_FILES**:

```
# pacman -S extundelete
```

Учтите, что это всего лишь рекомендации. Не стесняйтесь добавлять любые пакеты через *pacman* и настройте его подлинно по своему вкусу.

## Готовим сборку

Пора добавить список установленных пакетов и приступить к сборке собственного дистрибутива Arch. Это делается с помощью вышеупомянутого файла **packages.x86-64**, к тому же добавление списка пакетов, которые мы хотим иметь в новом дистрибутиве, не сложнее, чем внесение их в этот файл. Для этого введите команду

```
# pacman -Qqe >> ~/archive/releeng/packages.x86-64
```

>> вносит вывод команды *pacman -Qqe* в список **packages.x86-64**. Чтобы просмотреть список пакетов, введите

```
# nano ~/archive/packages.x86-64
```

Если вы используете среду рабочего стола с графическим менеджером входа, вам также потребуются скопировать файл **conf**. Для *lightdm* его можно найти в **/etc/lightdm**, что позволяет скопировать файл **conf** и любые другие файлы, необходимые для работы *lightdm*. Введите следующую команду:

```
# cp -r /etc/lightdm/* ~/archive/releeng/airootfs/etc/
```

Еще скопируйте файл **/etc/systemd/system/display-manager.service**, командой

```
# cp /etc/systemd/system/display-manager.service ~/archive/releeng/airootfs/etc
```

Пожелав, чтобы новый дистрибутив имел в системе arch тех же пользователей, что и сейчас, вы также должны скопировать соответствующие файлы, командой

```
# cp /etc/(shadow.passwd.group) ~/archive/releeng/airootfs/etc/
```

Если вы используете среду рабочего стола, необходимо скопировать файлы **~/xinitrc** и **~/config**. Неплохо бы также скопировать и файл **.bashrc**. Однако сначала создайте каталог в **airootfs**, командой:

```
# mkdir ~/archive/releeng/airootfs/etc/skel
```

После создания каталога можно скопировать файлы **~/xinitrc** и **~/config** командой:

```
# cp ~/.bashrc && cp ~/config && cp ~/xinitrc ~/archive/releeng/airootfs/etc/skel
```

Как упоминалось ранее, все эти действия должны выполняться от имени **root**. Однако такой подход не рекомендуется для повседневной работы. И если вы хотите войти как обычный пользователь, а не **root**, надо внести изменения в файл **etc/systemd/system/getty@tty1.service.d/autologin.conf**. В конце концов, это Arch, и ручное редактирование файлов конфигурации здесь довольно распространено. Введите команду

```
# nano ~/archive/releeng/airootfs/etc/systemd/system/getty@tty1.service.d/autologin.conf
```

Вы увидите нечто похожее на

```
ExecStart=/sbin/agetty --autologin root --noclear %l 38400 linux
```

— замените **root** на имя своего пользователя. Мы уже насоздавали столько конфигураций, что вам пора уже счесть это последней конфигурацией, которую надо сделать. Однако нам осталось еще настроить *airootfs* — это делается через файл **customize\_airootfs.sh**, имеющийся в **~/archive/releeng/airootfs/root**. Чтобы изменить файл, введите следующее:

```
# nano ~/archive/releeng/airootfs/root/customize_airootfs.sh
```

Мы просто вносим небольшие изменения в файл **customize\_airootfs.sh**, поэтому прокрутим вниз и убедимся, что запущены следующие службы:

```
systemctl enable pacman-init.service choose-mirror.service
```

```
systemctl set-default graphical.target
```

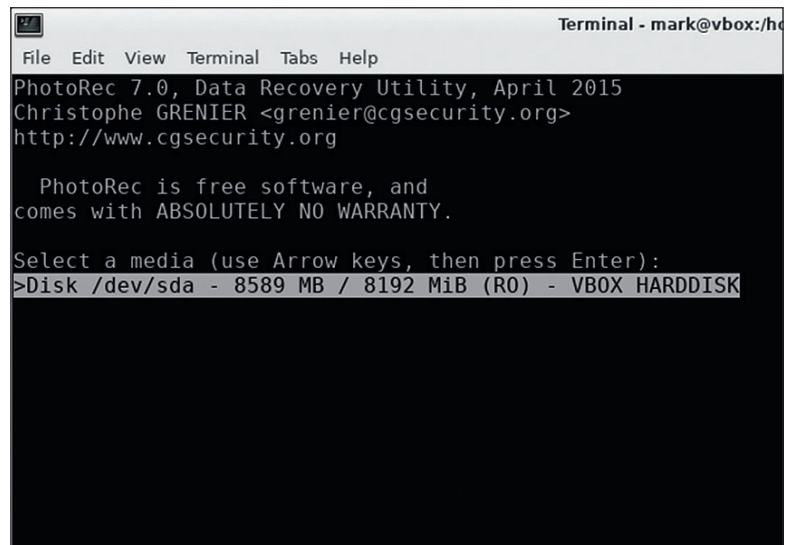
```
systemctl enable graphical.target
```

```
systemctl enable lightdm.service
```

Вот и всё — окончательная конфигурация создана! Мы готовы войти в каталог **~/archive/releeng** и собрать собственный ISO командой

```
# ./build.sh -v -N ExampleISO -V 0.1 -L ExampleISO_0.1
```

*-v* сообщает скрипту *build.sh* выводить в подробном режиме, чтобы вы могли получить более детальную информацию о происходящем. *-N* показывает, какое имя присвоить ISO, *-V* — номер версии, а *-L* — метка создаваемого ISO. Процесс сборки может потребовать времени, которое зависит от ресурсов виртуальной машины. По завершении процесса сборки вы найдете готовый ISO в каталоге **~/archive/releeng/out**. **LXF**



► **Photorec** может восстановить большинство типов файлов. Обратите внимание — для получения доступа ко всем дискам требуются права **root**.

## Осторожнее с командой dd...

Использование команды **dd** может фактически уничтожить данные, а не восстановить их, но только при неправильном использовании. Пример правильного использования можно увидеть в команде

```
# dd if=/dev/target_partition of=/home/user/partition.image
```

Часть **if=** команды должна быть первой при выполнении команды, и никогда — часть **of**. Легкий способ запомнить это — подумать об алфавите. Буква **l** идет раньше буквы **o**, поэтому раздел **if** выполняется перед разделом **of**.

В качестве примера команды **dd** можно привести следующее:

```
# dd if=/dev/sda of=/dev/sdb
```

выполнит резервное копирование всего содержимого **/dev/sda/** на диск **/dev/sdb/**, но только обязательно проверьте диск с помощью команды **lsblk**.

Вы даже можете создать задание *Cron* со сценарием оболочки BASH для запуска и делать регулярные резервные копии с интервалом по вашему выбору. Только предварительно убедитесь, что на втором диске хватит места на резервную копию.

*Cron* — это планировщик заданий, позволяющий выполнять повторяющиеся системные задачи. Для одноразового задания можно использовать утилиту *at*. *Cron* работает на таблице *crontab* или *crontable*, которая сообщает системе, какие команды запускать и когда. Например, команда **10 \* \* \* print** > **/var/log/apache/error\_log** очищает журнал ошибок *apache* каждый день через минуту после полуночи. Обратите внимание, что это произойдет, только если оболочка по умолчанию для пользователя *Cron* совместима с *Bash*.

ЧАСТЬ 9

# IoT: Дом, который послушен нам

Максим Черепанов предпочитает управлять своим Послушным Домом, лежа на диване. Лень — двигатель прогресса!



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».



**Ж**елаю здоровья всем, кто всё ещё с нами — со мной и с моим Послушным Домом!

Мы много сделали на пути разработки различных вариантов управления. Казалось бы, предусмотрено всё, что можно придумать: и почта, и телефон, и web-интерфейс. И прямое управление нагрузками по SSH-протоколу не забыто. Казалось бы — хватит, можно претворять в жизнь.

Но человек — странное существо. Всё время ищет возможность улучшить то, что кажется и так хорошим. Не обошла эта тенденция и меня.

Представим себе ситуацию: вы сидите дома, смотрите телевизор (или слушаете музыку, что гораздо лучше). И вам захотелось включить кондиционер. Казалось бы, что может быть проще — протянул руку за пультом, и готово. Но не всё так просто, если вы применили для дистанционного запуска технологии нашего ПД: цепь подачи напряжения разорвана нашим актуатором, и пульт от кондиционера тут не поможет. Надо встать и переключить вилку кондиционера из розетки ПД в розетку, на которую подается 220 вольт постоянно. Или уже пользоваться ПД: либо включить компьютер или смартфон и зайти на web-мордочку, либо звонить на Asterisk.

Согласитесь, это как стрелять из пушки по воробьям.

А решение — на поверхности: надо просто привязать к ПД пульт на инфракрасных лучах (ИК). Да, у вас в комнате станет ещё одним пультом больше. Но удобство его использования скомпенсирует наличие ещё одной маленькой пластмассовой коробочки.

Прежде чем засучить рукава, я изложу вам

## Совсем немного теории

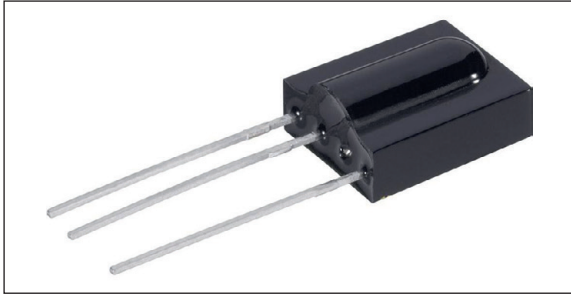
Инфракрасное, или тепловое излучение — это электромагнитное излучение, которое испускает любое нагретое до определенной температуры тело. ИК-диапазон лежит в ближайшей к видимому свету области спектра, в его длинноволновой части и занимает область приблизительно от 750 нм до 1000 мкм. Источником ИК-лучей, кроме нагретых тел, наиболее часто используются твердотельные излучатели — инфракрасные светодиоды, ИК-лазеры; для регистрации применяются фотодиоды, фототрезисторы или болометры. Некоторые особенности инфракрасного излучения делают его удобным для применения в устройствах передачи данных:

- » Твердотельные ИК-излучатели (ИК-светодиоды) компактны, практически безынерционны, экономичны и недороги.
- » ИК-приемники малогабаритны и также недороги.
- » ИК-лучи не отвлекают внимание человека в силу своей невидимости.
- » Несмотря на распространенность ИК-лучей и высокий уровень «фона», источников импульсных помех в ИК-области мало.
- » ИК-излучение низкой мощности не сказывается на здоровье человека.
- » ИК-лучи хорошо отражаются от большинства материалов (стен, мебели).
- » ИК-излучение не проникает сквозь стены и не мешает работе других аналогичных устройств.

Всё это позволяет с успехом использовать ИК-способ передачи информации во многих устройствах.

## Работа систем (пультов) управления бытовой электроникой

Пульт ИК-управления при нажатии кнопки излучает кодированную посылку, а приемник, установленный в управляемом устройстве, принимает ее и выполняет требуемые действия. Чтобы передать логическую последовательность, пульт формирует импульсный пакет ИК-лучей, информация в котором модулируется или кодируется длительностью или фазой составляющих пакет импульсов. В первых устройствах управления использовались последовательности коротких импульсов, каждый из которых представлял собою часть полезной информации. Однако в дальнейшем



► Рис. 1. Типичный TSOP.

стали использовать метод модулирования постоянной частоты логической последовательностью, в результате чего в пространство излучаются не одиночные импульсы, а пакеты импульсов определенной частоты. Данные уже передаются закодированными длительностью и положением этих частотных пакетов. ИК-приемник принимает такую последовательность и выполняет демодулирование. Такой метод передачи и приема отличается высокой помехозащищенностью, поскольку приемник, настроенный на частоту передатчика, уже не реагирует на помехи с другой частотой.

Сегодня для приема ИК-сигнала обычно применяется специальная микросхема, объединяющая фотоприемник, усилитель с полосовым фильтром, настроенным на определенную несущую частоту, усилитель с АРУ и детектор (англоязычное сокращение для этой микросхемы — TSOP, Temic Semiconductors Opto Electronics Photo Module), рис. 1.

Кроме электрического фильтра, такая микросхема имеет в своем составе оптический фильтр, настроенный на частоту принимаемого ИК-излучения, что позволяет в максимальной степени использовать преимущество светодиодного излучателя, спектр излучения которого имеет небольшую ширину. В результате таких технических решений стало возможным принимать маломощный полезный сигнал на фоне ИК-излучения других источников, бытовых приборов, радиаторов отопления и т.д. Работа современных устройств ИК-управления достаточно надежна, а дальность составляет от единиц до 40 и более метров, в зависимости от варианта реализации и уровня помех.

Практический вывод из всего сказанного один: в связи с тем, что TSOP имеет в своем составе фильтры, он предназначен для приема сигналов какой-то одной частоты. Т.е. с ним будут работать только определенные пульты.

## Форматы ИК-передачи данных

Различные производители бытовой аппаратуры применяют в своих изделиях различные пульты ИК-управления. Поскольку пульт должен общаться только с конкретным устройством, он формирует последовательность данных, уникальную для своего типа оборудования. Кроме собственно команды управления, передаваемые данные содержат адрес устройства, проверочные данные и другую сервисную информацию. Более того, различные производители используют различные способы формирования последовательности данных и различные способы передачи логических состояний. Наиболее распространенные способы кодирования битов информации — это изменение длительности паузы между пакетами (метод интервалов) и кодирование сочетанием состояний (бифазный метод). Однако встречаются способы кодирования бит информации длительностью, сочетанием длительности и паузы и т.д. Наиболее распространенные форматы передачи:

- » RC-5 — протокол компании Philips
- » NEC — протокол одноименной компании

Форматы RC-5 и NEC используются многими производителями электроники и занимают львиную долю на рынке стандартов для ИК-управления. Некоторые производители разработали свой стандарт, но в основном используют его сами. Менее распространенные форматы пультов управления мы даже упоминать не будем, это нам ни к чему.

«Очень вероятно, что производители моего BananaPi применили нечто стандартное», сказал я себе, когда брался за эту часть ПД. Можно долго читать теорию, искать апноуты производителя. А можно просто попробовать. Еще совсем чуть-чуть, и мы начнем это делать.

## Так вот ты какой, модуль TSOP!

Совсем немного времени потребовалось мне, чтобы выяснить технические характеристики этого модуля на BananaPi. Вот он, установлен рядом с Ethernet-коннектором и USB-разъемами (рис. 2).

Обратившись всё же к технической документации, я определил марку микросхемы: “AX-1838HS or comparable type”. Превосходно — значит, он на 38 кГц, а это как раз частота стандарта NEC: “The NEC IR transmission protocol uses pulse distance encoding of the message bits. Each pulse burst (mark — RC transmitter ON) is 562.5µs in length, at a carrier frequency of 38kHz (26.3µs)”. Мне становилось всё спокойнее (крепла уверенность, что бубен сегодня не пригодится), и росла внутренняя благодарность к производителям моего микрокомпьютера.

## Проверим модуль ядра

Прежде всего надо включить модуль, отвечающего за поддержку ИК-приемника в ядре. Попробуем его включить и проверить, включился ли он:

```
$ modprobe sunxi-ir && lsmod
```

А не тут-то было! Модуль не появляется в списке доступных. Я приготовился было менять ядро или дистрибутив (уже второй раз), но вовремя прочитал в документации: “Now, sunxi\_ir was built into kernel...” — встроен в ядро с версии ядра 3.4 (uname -r успокоила и обнадежила меня). Быстро набрав 2 команды —

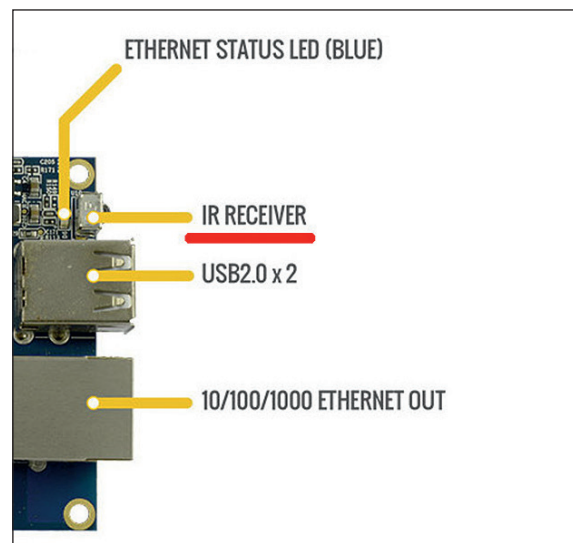
```
$ dmesg | grep sunxi-ir
```

```
$ cat /proc/bus/input/devices
```

я убедился, что всё в порядке: устройство /devices/virtual/input/input0 существует.

Значит, наша задача облегчилась более чем наполовину, осталось установить уже готовые программные пакеты.

»



► Рис. 2. TSOP на плате BananaPi.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)



➤ Рис. 3. Подходящие пульты (протокол NEC).

## EVTEST и LIRC

*Evtest* (events test) — программа для отслеживания сообщений от устройств ввода, клавиатуры и ПДУ. Она поможет нам определить подходящий пульт.

```
# apt-get install evtest
```

После установки запускаем программу:

```
# evtest /dev/input/event0
```

В консоль вывалится нечто похожее:

```
Input driver version is 1.0.1
Input device ID: bus 0x19 vendor 0x1 product 0x1 version 0x100
Input device name: "sunxi-ir"
Supported events: Event type 0 (EV_SYN)
Event type 1 (EV_KEY)
Event code 1 (KEY_ESC)
Event code 2 (KEY_1)
[...]
```

Вам покажут все поддерживаемые коды и предложат понюхать кнопки на пульте. Давите любую. Если в консоли появится что-то типа

```
Event: time 1500141888.532021, type 1 (EV_KEY), code 20 (KEY_T), value 1
Event: time 1500141888.532030, ----- SYN_REPORT -----
Event: time 1500141888.893360, type 1 (EV_KEY), code 20 (KEY_T), value 0
Event: time 1500141888.893373, ----- SYN_REPORT -----
```

то вас можно поздравить: ваш пульт поддерживается. Первое событие — кнопка нажата, второе — отпущена.

Таким образом я отобрал из кучки пультов, которые остались от разных устройств, всего три (рис. 3).

Теперь надо как-то привязать нажатия кнопок к действиям, которые мы хотим произвести. В этом нам поможет следующий пакет.

*LIRC* (Linux Infrared Remote Control — «инфракрасный ПДУ для Linux») — открытый программный пакет, позволяющий принимать инфракрасные сигналы пользователям операционной системы Linux. *LIRC*, кроме модуля ядра (который у нас в наличии), состоит из двух демонов:

➤ Один декодирует сигналы от пульта управления, принятые инфракрасным приемником, и предоставляет полученную информацию через сокет.

➤ Второй, подключаясь к этому сокету, использует принятую информацию о нажатиях кнопок пульта для выполнения конечных действий.

Без лишних комментариев устанавливаем его:

```
# apt-get install lirc
```

И сразу корректируем конфигурационный файл:

```
# nano /etc/lirc/hardware.conf
```

Приводим его к такому виду:

```
# /etc/lirc/hardware.conf
```

```
#
```

```
# Аргументы, используемые при запуске lircd
```

```
LIRCD_ARGS=""
```

```
# Не запускайте lircmd, даже если config выглядит хорошим
```

```
#START_LIRCMD=false
```

```
# Не запускайте irexec, даже если хороший config file вроде бы существует
```

```
#START_IEXEC=false
```

```
# Попробуйте загрузить подходящие модули ядра
```

```
LOAD_MODULES=true
```

```
# Запустите "lircd --driver=help" для получения списка поддерживаемых драйверов.
```

```
DRIVER="devinput"
```

```
# Для систем с udev параметр /dev/lirc0 обычно правильный
```

```
DEVICE="/dev/input/event0"
```

```
MODULES="sunxi-ir"
```

```
# Файлы конфигурации по умолчанию для ваших устройств, если таковые есть
```

```
LIRCD_CONF=""
```

```
LIRCMD_CONF=""
```

Теперь надо сконфигурировать свой пульт. Я выбрал тот, что побольше; на фото он в середине (у кого-то из знакомых необратимо сгорел плейер Shinco, отремонтировать его не удалось). Меня привлекло аккуратное исполнение пульта и размер: он удобнее лежит в руке, чем остальные, те уж слишком маленькие. Да и кнопок много, можно много чего к ним привязать, и они уже подписаны.

Иду на сайт <http://lirc.sourceforge.net/remotes> и обнаруживаю на FTP-сервере папочку **shinco**. Вхожу — и, о чудо! Есть конфигурационный файл для моего пульта! Я думаю, что так повезет и вам: на сайте невероятное количество папочек по названию фирм, и вы почти с гарантией найдете конфигурационный файл под свой пульт. Но даже если не найдется файл именно для вашего пульта, то его можно будет создать заново.

Пока что скачиваю конфиг:

```
# wget http://lirc.sourceforge.net/remotes/shinco/RC-1730
```

Запустим программу для создания нового конфигурационного файла на базе имеющегося:

```
# irrecord -H devinput -d /dev/input/event0 RC-1730
```

и просто нажмите 2 раза Enter. Будет сгенерирован новый, пустой файл с нужным заголовком, в котором будут указаны все тайминги (временные константы), необходимые для нормальной работы конкретного пульта.

Теперь стоит посмотреть на доступные наименования клавиш, которые понимает команда `irrecord`:

```
# irrecord -l
```

Вывалится листинг на несколько экранов. Это псевдонимы команд; стоит всё это скопировать в текстовый редактор. Выберите несколько, которые будут впоследствии понятны для привязки к командам. Я выбрал четыре:

```
KEY_1
```

➤ Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.



```
KEY_2
KEY_3
KEY_4
```

Просто привяжу их к кнопкам 1–4 пульта и таким же каналам управления на ПД. И никто никогда не запутается.

Теперь запускаем программу на запись команд:

```
# irrecord -H devinput -d /dev/input/event0 RC-1730.conf
```

Поочередно вводите псевдонимы, которые вы выбрали, и нажимайте Enter, а потом нажимайте нужную кнопку пульта. По окончании записи нажмите Enter на пустой строке, и запись будет закончена.

В текущей директории появился файл **RC-1730.conf.conf**, в котором есть только те кнопки, которые вы задали. Откройте его редактором и найдите строки для каждой из кнопок. У меня они такого вида:

```
KEY_1 0x01000900000001 0x00000000000000
```

Удалите 3-ю группу цифр — нули, чтобы было вот так:

```
KEY_1 0x01000900000001
```

Сохраните файл как **/etc/lirc/lircd.conf**.

Всё, у вас полностью работоспособный конфиг для вашего пульта! Проверить это можно так:

```
# irw
```

И нажимайте те кнопки, которые вы задали. Вы увидите те (и только те) коды и те псевдонимы, которые вы им задали (я задал только 4, для эксперимента):

```
0001000900000001 00 KEY_1 RC-1730.conf
0001001d00000001 00 KEY_2 RC-1730.conf
0001001f0000000EY_1 00 K3 RC-1730.conf
0001000d00000001 00 KEY_4 RC-1730.conf
```

## IREXEC

Теперь пора привязать к псевдонимам действия. Для этого в домашней директории пользователя, от которого запущен *lircd*, нужно создать файл **.lircrc**. Это конфигурационный файл для действия демона *irexec* — программы (части *LIRC*), которая выполняет действия по нажатию кнопки на пульте.

Синтаксис файла таков:

```
begin
prog = приложение
button = кнопка
config = команда
end
```

Для каждой кнопки должна быть одна такая секция. Вот например, как выглядит секция для кнопки №1 у меня:

```
begin
prog = irexec
button = KEY_1
config = touch /tmp/11 &
config = touch /tmp/10
end
```

2 строки `config` написаны для задания на нажатие одной кнопки двух действий — включения и выключения нагрузки. Как видите, я применил уже готовый способ управления нагрузками — создание файлов-флагов во временной директории **/tmp**. А далее алгоритм работы ПД уже реализован.

По окончании заполнения файла и его сохранения запустите *irexec* в режиме демона:

```
# irexec -d
```

И можете получать удовольствие от управления ПД еще и пультом.

Немного по поводу строки “`config`”: записать в эту строку можно что угодно. Но для удобоваримого чтения этого файла впоследствии лучше создать скрипты на *Bash* и присвоить им говорящие названия. Допустим, вы хотите с пульта запускать или останавливать *motion* для записи на диск (очень логично: уходим



► Рис. 4. Универсальный ИК-пульт.

из дома — запустили запись, вернулись — остановили). Для этого надо в конфиг *motion* внести изменение и перезапустить программу. Вносите всё это в скрипт, называете его **motion\_start\_record.sh**, делаете его исполняемым, указываете на него в строке `config` кнопки, которую вы обозвали REC, и готово.

И не забудьте добавить запуск *irexec* в `crontab`:

```
# crontab -e
```

Вставьте строку

```
@reboot /root/irexec_start.sh
```

Создаем в директории **/root** файл **irexec\_start.sh**, делаем его исполняемым и наполняем:

```
#!/bin/bash
/usr/bin/irexec -d
exit 0
```

Перезагрузите микрокомпьютер и убедитесь, что служба запустилась:

```
# ps aux | grep irexec
```

Теперь — бонус! Наверняка эту статью скучновато было читать тем, у кого нет микрокомпьютера Raspberry или Banana Pi. Тем не менее, вы с успехом можете применить всё, что написано в этой статье, для обычного компьютера с ОС Linux.

Представляю вам универсальный пульт для ПК (рис. 4). Он состоит из собственно пульта и ИК-приемника, включаемого в свободный порт USB любого компьютера. У меня он увиделся так:

```
lsusb
Bus 003 Device 002: ID 1d57:ad02 Xenta — на BananaPi
Bus 004 Device 002: ID 1d57:ad02 Xenta SE340D PC Remote Control — на ноутбуке
```

На ноутбуке это устройство сразу определилось как мышь и клавиатура. Проверим на BananaPi? Для этого достаточно посмотреть, какие устройства нам выдаст *evtest*.

```
# cat /proc/bus/input/devices
```

Увидите 2 новых устройства: клавиатуру и мышь.

Какие события к ним подвязаны? Давайте выясним и это:

```
# ls /dev/input/
```

Увидим 2 новых блочных устройства, `event2` и `event3`. Пробуйте нажимать на кнопки пульта и управлять манипулятором мыши на пульте, предварительно запустив *evtest*:

```
# evtest /dev/input/event2
```

Как вы понимаете, дальнейшая настройка совершенно аналогична тому, что написано выше.

В следующем месяце продолжим, до встречи! LXF

# Sed: Неинтерактивный редактор

Дмитрий Кузнецов забирается в дебри загадочного редактора командной строки и убеждается, что тот вполне поддается пониманию.



Наш эксперт

Дмитрий Кузнецов более 10 лет ведет неравный бой с монстрами сложности, хотя некоторые почему-то их считают ветряными мельницами.



Вот уже почти полвека неинтерактивный текстовый редактор `sed` применяется для автоматической обработки текстов в ОС семейства `*nix`. Современному пользователю, привыкшему к GUI, синтаксис его команд кажется сложным. Да и само словосочетание «неинтерактивный текстовый редактор» для новичков выглядит довольно странно. В результате многие ограничиваются лишь небольшим набором где-то подсмотренных и механически запомненных «мантр». Однако, как и у большинства других удачных программ, вся мощь `sed` заключена в синергетическом взаимодействии всего лишь нескольких простых принципов, поняв которые, можно легко научиться строить `sed`-выражения любой сложности.

## Принцип работы

Концептуально `sed` очень напоминает математическую функцию  $T1 = R(T0)$ , которая преобразует входной текст ( $T0$ ) в выходной ( $T1$ ) по определенным правилам ( $R$ ). Механизм этого преобразования довольно прост. Входной текст обрабатывается построчно. Прочитав очередную строку, `sed` отрезает от нее завершающий символ “\n”, помещает ее в буфер, выполняет над его содержимым набор заданных пользователем команд и выводит полученный результат в выходной поток вместе с “\n” на конце. Автоматический вывод после каждой итерации можно отключить, запустив `sed` с ключом

`-n`. В таком режиме вывод приходится выполнять явно, с помощью специальных команд. Эту простую схему немного нарушают лишь две команды, позволяющие прочитать следующую строку входного текста во время обработки предыдущей.

Выглядит довольно примитивно. Однако два обстоятельства кардинально меняют дело. Во-первых, у `sed`, помимо уже упомянутого основного буфера, есть еще и дополнительный, в котором при необходимости можно что-то сохранить для использования на следующих итерациях. А во-вторых, система команд `sed` образует полноценный, хотя и достаточно специфичный, язык программирования с регулярными выражениями и передачей управления по меткам. Всё это позволяет вытворять с текстом весьма замысловатые трюки.

## Пробуем sed на практике

В мире программирования уже стало традицией начинать изучение нового языка с написания простейшей программы “Hello, world!”. Это, конечно, не пустой ритуал. Работающая, хоть и примитивная, программа — отличный испытательный полигон для отработки изучаемых языковых конструкций на практике. Для командного языка `sed` такой подход тоже вполне пригоден.

Рассмотрим команду `sed`, вставляющую в начало входного текста приветствие. Выглядит она так: `'iHello, world!'`. А понимать ее нужно следующим образом: перед строкой номер 1 вставить текст “Hello, world!”. В тестовых целях в качестве входного текста удобно использовать вывод команды `seq N`. Она генерирует последовательность натуральных чисел от 1 до N, разделенных символом перевода строки (`'\n'`). Итак, в итоге получается:

```
seq 3 | sed -e 'iHello, world!'
```

Это лишь один из вариантов использования `sed`, когда входной текст подается в `stdin` через конвейер. А если текст хранится в файле? В этом случае, конечно, можно использовать `cat`:

```
cat ./in_file.txt | sed -e 'iHello, world!'
```

Или просто указать имя файла в аргументах `sed`:

```
sed -e 'iHello, world!' ./in_file.txt
```

Во всех приведенных выше примерах результат выводится в `stdout`. При необходимости его нетрудно перенаправить в файл:

```
sed -e 'iHello, world!' ./in_file.txt > ./out_file.txt
```

Иногда необходимо просто отредактировать файл. Первое, что приходит в голову — заменить в предыдущем примере `out_file.txt` на `in_file.txt`. К сожалению, такой прием не работает. Однако эта задача так часто встречается, что в `sed` для ее решения есть специальный ключ `-i`:

```
sed -i -e 'iHello, world!' ./in_file.txt
```

Также следует отметить, что `sed`-команды не обязательно каждый раз вручную набирать в командной строке. Их можно сохранять в файле, а затем передавать его `sed` с помощью ключа `-f`:

```
sed -i -f ./cmd.sed ./in_file.txt
```

Ключи `-e` и `-f` могут использоваться вместе и по несколько раз. Каждый из них добавляет команды к итоговой `sed`-программе, которая будет обрабатывать текст.

## Элементы командного языка

В общем случае команда `sed` имеет вид `'[addr]X[options]'`. Здесь `X` — односимвольное обозначение команды, `options` — дополнительные аргументы, а `addr` — адресное выражение, ограничивающее действие команды определенным набором строк входного текста. В некоторых случаях `options` может отсутствовать. Если опущен `addr`, команда применяется ко всем строкам. В рассмотренном выше примере `'iHello, world!'` присутствуют все три компонента: `addr` равно 1, `X` — `i`, а `options` — `"Hello, world!"`.

Ключи `-e` и `-f` часто задают не одну команду, а целую последовательность. В таких случаях команды должны быть разделены символом перехода на новую строку или точкой с запятой. Первый способ универсален; второй, к сожалению, иногда не работает. Дело в том, что `sed` не всегда может корректно интерпретировать символ `;`. В качестве примера можно привести уже упоминавшуюся команду `i`. Как понимать точку с запятой в ее конце? Это конец команды или часть вставляемого текста? Интерпретатор `sed` считает, что это текст, и требует, чтобы такие команды всегда заканчивались переходом на новую строку.

Одно адресное выражение можно применить к нескольким командам; для этого их надо объединить в группу, заключив в фигурные скобки.

## Адресные выражения

Адресное выражение задает подмножество строк входного текста. Оно определяет часть текста, на которую распространяется действие команды. Существует два принципиально разных способа идентификации строк: по номерам и по содержанию. Оба они активно применяются в адресных выражениях `sed`.

В основе первого способа лежит довольно очевидная идея. Каждой строке входного текста ставится в соответствие порядковый номер, начиная с 1, а адресное выражение задает некоторое подмножество этих номеров. Для последней строки используется специальное обозначение — `$`. Если нужно нацелить команду на какую-то одну строку, достаточно в качестве адресного выражения указать ее номер. Примером может служить приведенная выше команда `'iHello, world!'`. Диапазон строк задается его границами, перечисленными через запятую. Например, выражение `1,$` означает весь текст, от первой до последней строки. GNU-версия `sed` добавляет еще ряд возможностей:

» Выражение `"first-step"` задает множество строк, номера которых определяются формулой `first + (n * step)`. Например, `2-2` — все четные строки, `1-2` — все нечетные, а `1-1` — экстравагантный синоним `1,$`.

» `n,m` — диапазон из `m+1` строк, начиная с `n`-й. Например, `1,+3` — первые четыре строки.

» `n,~m` — диапазон строк от `n`-й до следующей за ней с номером, кратным `m`. Например, выражения `7,~5` и `7,10` эквивалентны.

Для идентификации строк по содержанию `sed` использует регулярные выражения. Адресное выражение этого типа имеет вид: `/regex/`. Оно задает подмножество строк входного текста, соответствующих регулярному выражению `regex`. Например, `/[2-5]/` означает все строки, в которых встречаются цифры 2, 3, 4 или 5.

Регулярные выражения — отдельная и довольно известная тема. Они активно используются не только в `sed`, но и во многих других программах, библиотеках и языках программирования, поэтому на нашем уроке их синтаксис рассматриваться не будет. Если для обрамления `regex` символ `"/"` по каким-то причинам

» Рис. 1. Различные варианты вызова `sed`.

не устраивает, можно использовать альтернативный синтаксис — `%regexpr%`, где `%` — любой символ. Например, приведенный выше пример можно переписать так: `\+[2-5]+`.

В адресных выражениях, задающих диапазон, оба способа идентификации строк можно смешивать. Например, `5,/Hello/` задает все строки от пятой до первой после нее, в которой встречается `"Hello"`.

Добавление восклицательного знака в конце адресного выражения инвертирует его. Например, если `2` означает «вторая строка», то `!2` — «все строки, кроме второй».

## Основные команды

В общей сложности в `sed` всего несколько десятков команд. Каждая из них, как уже было сказано, обозначается одним символом. Эта краткость часто отпугивает новичков. Однако обозначения выбраны осмысленно, а потому, обладая минимальными знаниями английского языка, их нетрудно запомнить.

В этом разделе будет рассмотрена довольно разношерстная категория команд. Объединяет их лишь простота и частота использования. Все они выполняют какую-нибудь операцию над основным буфером или что-то добавляют к выходному тексту.

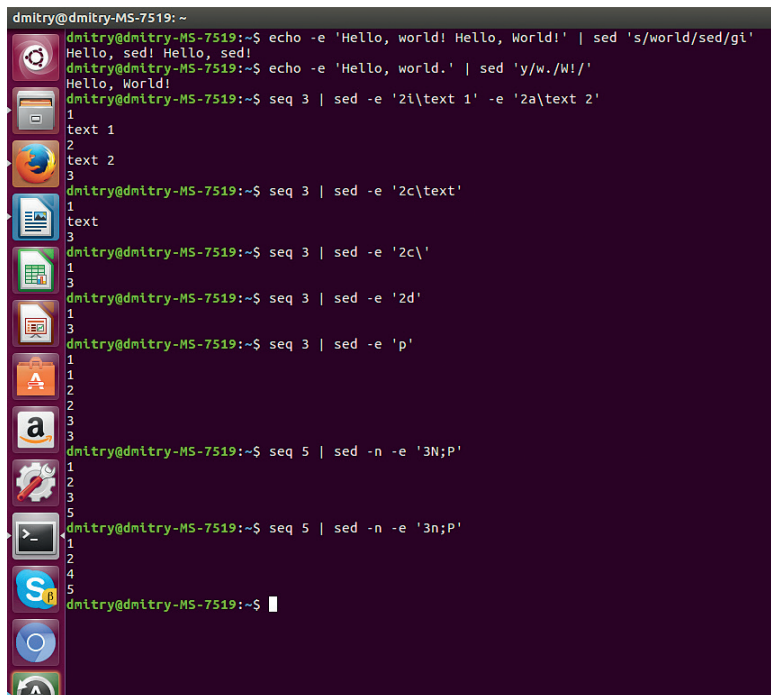
Начнем с команды подстановки по регулярному выражению `s` (от *англ.* substitute). Это, вероятно, самая известная и мощная команда `sed`. Благодаря регулярным выражениям, она в одиночку способна выполнять весьма замысловатые преобразования текста. Точный формат команды такой: `'s/regex/rep/flags'`. Она выполняет поиск в основном буфере по регулярному выражению `regex` и заменяет найденные последовательности символов на `rep`. Поле `flags` задает различные режимы. Если оно опущено, выполняется замена только первой найденной последовательности символов. Флаг `g` позволяет заменить все соответствия, а в GNU-версии `sed` есть флаг `i`, включающий режим регистронезависимого поиска. Это лишь наиболее часто используемые флаги. Полный список возможных значений `flags` можно найти в документации `sed`. Вот пример, демонстрирующий описанные возможности команды:

```
echo -e 'Hello, world! Hello, World!' | sed 's/world/sed/gi'
```

Здесь во входном тексте заменяется `"world"` на `"sed"`.

»

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)



► Рис. 2. Основные команды и команды чтения строк.

Немного похожая по синтаксису команда используется для замены отдельных символов в тексте: `'y/src-chars/dst-chars/'`. Она никак не связана с регулярными выражениями, а просто заменяет в основном буфере все символы, содержащиеся в `src-chars`, на соответствующие символы `dst-chars`. Например, в тексте “Hello, world.” заменим “w” и “.” на “W” и “!” соответственно:

```
echo -e 'Hello, world.' | sed 'y/w./W!/'
```

Вряд ли можно представить редактирование текста без возможности вставки новых строк. В `sed` для этого предусмотрены две команды: `'i\text'` (от *англ.* insert — вставить) и `'a\text'` (от *англ.* append — приставить). Обе они вставляют текст `text` в выходной поток. Но первая это делает до обработки строки входного текста, а вторая — после. Так, например, можно перед второй строкой вставить “text 1”, а после нее — “text 2”:

```
seq 3 | sed -e '2i\text 1' -e '2a\text 2'
```

Вставляемый текст может содержать несколько строк, которые должны быть разделены символом “\n”. Обратная косая черта между обозначением команды и текстом может быть опущена. В этом случае все пробелы в начале текста игнорируются. Наличие символа “\” говорит о том, что вставляемый текст начинается сразу после него, поэтому ведущие пробелы допустимы.

Есть еще одна похожая команда — `'c\text'` (от *англ.* change — заменить). Она тоже вставляет текст в выходной поток, но при этом очищает основной буфер и заставляет `sed` перейти на следующую итерацию цикла обработки строк. Иными словами, она заменяет текст. Вот, например, так можно заменить вторую строку на “text”:

```
seq 3 | sed -e '2c\text'
```

У команды `c` есть одно важное отличие от `i` и `a`. Оно заключается в методе обработки диапазонов строк. Команда `c` удаляет все строки диапазона, а затем один раз вставляет указанный текст. Команды `i` и `a` работают с каждой строкой отдельно, они выполняют вставку текста при обработке каждой строки диапазона.

А если надо удалить строку? Это тоже довольно важная операция. Можно попробовать применить `c` с пустым текстом:

```
seq 3 | sed -e '2c\''
```

Такой трюк работает, но в `sed` для этого имеется специальная команда `d` (от *англ.* delete — удалить):

```
seq 3 | sed -e '2d'
```

Она интересна еще и тем, что относится к так называемым парным командам. Это обычные команды, но у каждой из них есть

команда-двойник, обозначаемая тем же символом, но в верхнем регистре. Смысл парных команд похож, но не идентичен. Различия обычно касаются обработки многострочного текста. Так, если `d` очищает основной буфер и завершает текущую итерацию, то `D` удаляет из основного буфера лишь часть текста до первого символа “\n” включительно, и снова запускает всю `sed`-программу над текстом, оставшимся в буфере. Если же “\n” не обнаружен, то `D` работает так же, как `d`.

Еще одна из команд-двойняшек — `p` (от *англ.* print — печатать). Она просто выводит содержимое основного буфера в выходной поток и обычно применяется при запуске `sed` с параметром `-n`, подавляющим автоматический вывод содержимого основного буфера в конце каждой итерации. Например,

```
seq 3 | sed -e 'p'
```

печатает каждую строку входного текста дважды: один раз — с помощью `p`, второй — автоматически при завершении итераций. Команда `P`, как нетрудно догадаться, выводит только первую строку текста основного буфера. А если там однострочный текст, то `P` идентична `p`.

## Принудительное чтение строк

Каким образом в основном буфере может оказаться многострочный текст, если на каждой итерации `sed` считывает туда лишь одну строку входного текста? Если ограничиться только рассмотренными выше командами, то такая ситуация невозможна. Именно поэтому при описании команд `D` и `P` не были приведены примеры их использования.

В `sed` есть два способа загнать несколько строк в основной буфер. Первый — операции с дополнительным буфером. Им посвящая следующий раздел статьи. Второй — команды принудительного чтения строк. О них речь пойдет прямо сейчас.

Как уже было сказано, `sed` в цикле читает строки входного текста, применяя к каждой из них заданные пользователем команды. Однако есть возможность нарушить этот порядок. Пара команд `n` и `N` (от *англ.* new — новый или next — следующий) позволяет прочитать следующую строку, не дожидаясь окончания текущей итерации.

Сначала рассмотрим последнюю, как более простую. Она читает следующую строку входного текста и присоединяет ее к содержимому основного буфера, предварительно добавив к нему символ “\n”. Например,

```
seq 5 | sed -n -e '3N;P'
```

печатает все строки, кроме четвертой. Здесь `sed` запускается с параметром `-n`, поэтому команда `P` — единственный источник выходного текста. Во входном тексте пять строк. Первые две считываются в основной буфер автоматически в начале соответствующих итераций цикла и выводятся командой `P`. Символов “\n” в них, разумеется, нет, поэтому в данном случае `P` равносильна `p`. Третья строка считывается так же, но при ее обработке выполняется команда `3N`, которая добавляет к основному буферу, уже содержащему третью строку, “\n” плюс только что прочитанную четвертую. Теперь в основном буфере текст из двух строк. Затем выполняется команда `P`, печатающая только первую из них, то есть третью строку входного текста. Это была последняя команда; дальше — очистка буфера и переход к следующей итерации. Она будет работать с пятой строкой, поскольку четвертая уже читалась командой `N`, и абсолютно аналогична первым двум.

Команда `n` немного сложнее. Она выполняет два действия: печатает основной буфер и считывает в него следующую строку, при этом, в отличие от команды `N`, затирает его содержимое. Первое выполняется только в том случае, если `sed` запущен без параметра `-n`. Если текущая строка — последняя, то есть следующую прочитать невозможно, то на `n` выполнение `sed`-программы завершается. Заменив в предыдущем примере `N` на `n`, можно заставить `sed` пропустить третью строку, а не четвертую:

```
seq 5 | sed -n -e '3n;P'
```

Команда `3n`, в отличие от `3N`, затрет содержимое основного буфера следующей прочитанной строкой. В остальном оба примера аналогичны.

## Работа с дополнительным буфером

Рассмотренных команд вполне достаточно для большинства задач, но иногда всё же требуется нечто большее. Чаще всего проблемы возникают из-за независимости итераций цикла обработки строк входного текста. В конце каждой итерации основной буфер очищается, и обработка следующей строки начинается с чистого листа. Поэтому дополнительный буфер, в котором можно что-то сохранять, а на следующих итерациях извлекать и использовать, оказывается очень кстати. В *sed* для работы с ним предусмотрены пять простых команд: одна обычная (`x`) и две парные (`h`, `H`, `g`, `G`).

Команда `x` (от *англ.* *eXchange* — обменять) меняет местами содержимое основного и дополнительного буферов. Вот простейший пример, демонстрирующий ее поведение:

```
seq 3 | sed -n -e '2x;p'
```

Команда `p` выполняется для всех строк, а для второй перед ней еще и `2x`. В этот момент в основном буфере вторая строка, а дополнительный пуст. Команда `x` меняет их местами. В результате вместо второй строки `p` напечатает пустую.

Если `x` выполнить дважды, первоначальное содержимое буферов будет восстановлено. А значит,

```
seq 3 | sed -n -e '2{x;x};p'
```

должно быть равносильно

```
seq 3 | sed -n -e 'p'
```

Эксперимент показывает, что так оно и есть.

Команды `h` и `H` (от *англ.* *hold* — придержать; в документации *sed* дополнительный буфер называется “hold space”) отправляют содержимое основного буфера в дополнительный. При этом первая выполняет присваивание, то есть прежнее содержимое дополнительного буфера теряется, вторая — добавляет строку в конец, отделяя ее символом “\n”. Команды `g` и `G` абсолютно аналогичны, но действуют в обратном направлении — из дополнительного буфера в основной. Например, так можно напечатать строки входного текста в обратном порядке:

```
seq 5 | sed -n -e 'G;h;$p'
```

Основная часть работы выполняется командами `G` и `h`; `$p` только выводит конечный результат при обработке последней строки. Перевернутый входной текст постепенно накапливается в дополнительном буфере. На каждой итерации он извлекается из него командой `G` и добавляется к основному буферу, содержащему обрабатываемую на текущей итерации строку. После чего `h` сохраняет очередной промежуточный результат, обогащенный новой строкой, в дополнительный буфер. Таким образом, на последней итерации после выполнения `G;h` в основном буфере оказывается перевернутый входной текст, который и выводится командой `$p`.

Этот пример работает, однако в конце перевернутого текста печатает пустую строку. Почему это происходит? Причина кроется в первой итерации. В основном буфер загружается первая строка, а дополнительный пуст. Поэтому команда `G` добавляет к основному буферу только “\n”. Это и есть та самая пустая строка. Избавиться от нее очень просто. Нужно, чтобы `G` не выполнялась для первой строки. Этого нетрудно добиться с помощью адресного выражения `2,$`:

```
seq 5 | sed -n -e '2,$G;h;$p'
```

## Управление потоком выполнения

Возможность управлять потоком выполнения превращает командный язык *sed* в полноценный, хотя и немного необычный, язык программирования. Однако злоупотреблять ею не стоит. Каждый случай использования команд этой категории — повод задуматься о применении более развитого и удобного языка

```
dmitry@dmitry-MS-7519: ~
dmitry@dmitry-MS-7519:~$ seq 3 | sed -n -e '2x;p'
1
3
dmitry@dmitry-MS-7519:~$ seq 3 | sed -n -e '2{x;x};p'
1
2
3
dmitry@dmitry-MS-7519:~$ seq 5 | sed -n -e 'G;h;$p'
5
4
3
2
1
dmitry@dmitry-MS-7519:~$ seq 5 | sed -n -e '2,$G;h;$p'
5
4
3
2
1
dmitry@dmitry-MS-7519:~$
```

Рис. 3. Команды для работы с дополнительным буфером.

программирования (например, Perl). Разработчики *sed*, по-видимому, придерживались такого же мнения, а потому реализовали весьма аскетичный набор команд:

- » `:label` устанавливает метку `label` в *sed*-коде.
- » `b label` — безусловный переход на метку `label`.
- » `t label` — условный переход на метку `label`. Он срабатывает, только если предыдущая команда подстановки (`s`) выполнена успешно.
- » `q [exit_code]` — завершить выполнение *sed* с кодом `exit_code` (если опущен, считается равным 0).

Вот, например, бесконечный цикл, печатающий первую строку входного текста:

```
seq 3 | sed -n -e ':lb; p; b lb'
```

А это пример условного перехода:

```
seq 3 | sed -n -e 's/2//; t lb; p; :lb'
```

Он печатает все строки, не содержащие “2”. При обработке каждой строки команда `s/2//` пытается заменить текст “2” на пустую строку. Если замена успешна, срабатывает условный переход `t lb`, заставляющий *sed* перепрыгнуть команду печати `p`. Того же самого можно добиться с помощью безусловного перехода:

```
seq 3 | sed -n -e '/2/b lb; p; :lb'
```

Здесь адресное выражение фактически превращает безусловный переход в условный.

Однако проще всего было бы вообще обойтись без всяких меток и переходов, к чему и следует всегда стремиться:

```
seq 3 | sed -n -e '/2/d;p'
```

Напоследок — пример использования команды `q`. Он просто печатает строки, а когда встретится строка, содержащая “2”, прерывает выполнение:

```
seq 3 | sed -n -e '/2/q;p'
```

## Заключение

Эта статья не претендует на исчерпывающее руководство по *sed*, хотя почти все его команды здесь описаны достаточно полно. Главная ее цель — рассказать о принципиальных идеях, лежащих в основе этого инструмента. Понимая их, незнакомые команды или детали синтаксиса уже известных всегда легко подсмотреть в документации. **LXF**

```
dmitry@dmitry-MS-7519: ~
dmitry@dmitry-MS-7519:~$ seq 3 | sed -n -e 's/2//; t lb; p; :lb'
1
3
dmitry@dmitry-MS-7519:~$ seq 3 | sed -n -e '/2/b lb; p; :lb'
1
3
dmitry@dmitry-MS-7519:~$ seq 3 | sed -n -e '/2/d;p'
1
3
dmitry@dmitry-MS-7519:~$ seq 3 | sed -n -e '/2/q;p'
1
dmitry@dmitry-MS-7519:~$
```

Рис. 4. Команды управления потоком выполнения.

# Загадка LXF: Скажи пароль

В первой загадке **Михалис Цукалос** генерирует сложные пароли на разных языках программирования и подначивает вас его превзойти.



**Наш эксперт**

### Михалис Цукалос

Михалис — администратор UNIX, программист-математик и администратор баз данных, который любит писать статьи. С ним можно связаться через Twitter: @mactsouk.



```
2. mtsouk@mail: ~/docs/article/working/puzzle1.LXF226/code (ssh)
code$ gcc -Wall -o random random.c
code$ ./random
Wrong number of arguments. Exiting...
code$ ./random 10 10
1215069295 1311962008 1086128678 385788725 1753820418 394002377 1255532675 906573271 5
4404747 679162307
code$ ./random 10 100
1215069295 1311962008 1086128678 385788725 1753820418 394002377 1255532675 906573271 5
4404747 679162307 131589623 179656373 2097642528 2100364173 1876662344 113981581 57764
8320 198294838 1891854071 1208116668 1424885268 2107628344 2065686955 299460181 479838
562 460457228 25739512 1864854523 712507288 1151291766 1195282557 1927576583 315770127
133927588 165881660 2069590545 527929965 1421414335 828680169 582334712 2100576643 90
0269792 761991085 2050735523 913150317 491169781 17233456 1490798638 689464620 1909087
527 551431658 2114349888 1869232223 469634965 266326421 201387137 930092193 292065933
2066441660 1642599492 1443357700 1114240570 1422692417 1759127827 1248168158 158837407
8 1681234724 1776098123 862504765 362431245 210949187 815597760 1322701037 972940272 7
18849635 88367707 1464110054 736083092 1579166345 6091026 497686971 2130598003 2120440
914 219435547 452749320 239283687 421022684 1382841513 531349621 339980697 877957347 1
974707321 1454221267 153166117 1586351500 554905777 1741740195 1120102576 183520252 45
6761312
code$ ./random -1 100
254925627 1205188300 366127624 1401405153 76053476 1604170158 1302235366 362229243 334
960208 1882140968 960816832 627031785 1185930294 1926318319 973293805 2114955885 16268
37490 1438327473 1589924831 1644716430 749102841 1323486100 1527510893 1148413881 1450
167526 1972740189 382098517 1483145305 1658242591 112136530 1605621846 1943168218 1317
324830 1971749470 1197089724 1393378386 1428439800 351841442 1755607549 1763396188 864
98762 568940733 242944325 1272429056 34775404 1216238130 1239901294 1974612895 507881
955 682342477 1471845677 1256184797 2005828578 851872922 257115030 1308512456 67712946
3 639213547 644174613 217888406 751350077 102312811 13572976 2068674908 2074062281 121
0662700 1314569566 1355014613 1562504143 922693468 970927153 1649002905 1491634201 121
3871479 773948314 1839409606 282625961 2013849608 1666538853 789707917 548708437 99090
0882 2045892714 407053367 1842773804 155524096 1715565823 372419619 794737643 21225678
8 590308025 1546087721 314569599 608381001 1467278981 241148232 1814543702 63464899 1
```

➤ **Рис. 1.** Вывод программы `random.c`. При одном и том же начальном значении последовательности сгенерированных случайных чисел всегда будут одинаковы!

Итак, настоящая проблема заключается в том, как перейти от случайного числа к символу ASCII в C, а также в Python 3, Perl и Go. Если вы используете другой язык программирования, вам придется найти соответствующий метод для преобразования целого числа в символ ASCII.

Простейший способ — уменьшить большое целое число, чтобы оно попало в заданный диапазон, после чего его можно будет преобразовать в символ ASCII. Для этого используется операция деления по модулю, которая находит остаток от деления двух целых чисел. Если вы делите число по модулю 10, то остаток будет целым числом от 0 до 9. Поскольку число допустимых символов ASCII составляет около 130, вам нужно разделить сгенерированное случайное число на 130. Но можно выбрать и меньшее число. Однако если в вашем языке программирования можно генерировать случайные числа в заданном диапазоне, то операция деления по модулю может не потребоваться.

## О, эта случайность!

Первый шаг в решении исходной задачи — создание случайных чисел. В этом разделе будет представлена небольшая программа на C, которая считывает из командной строки начальное значение и количество случайных чисел, которые нужно создать, а затем выводит созданные числа на экран.

Имя программы — `random.c`, и она содержит следующий код:

**П**рочитав эту статью, вы узнаете, как создавать надежные пароли на разных языках программирования. Однако основная цель этого учебника — вовлечь вас в процесс, заставить думать и, возможно, даже придумать лучшее решение по сравнению с предлагаемыми здесь.

В некоторых языках программирования есть модули и пакеты, которые генерируют хорошие пароли автоматически. Однако проблема в том, что с ними у вас остается мало простора для обучения, настройки и экспериментов, и это лишает вас удовольствия от программирования вообще.

В следующем выпуске *Linux Format* мы опубликуем код и список участников, которые хорошо справились с поставленной задачей. Следите за новостями...

## Думать — это трудно

Компьютеры работают с цифрами и не могут генерировать буквы или слова напрямую. Вам надо найти способ создания случайных чисел и их преобразования в слова, содержащие буквы и цифры.

Прежде всего надо решить, будем ли мы создавать пароли с символами Unicode. Ответ довольно-таки прост: символов ASCII вполне достаточно для достаточно сложных паролей, поэтому добавлять к ним символы Unicode незачем.

### Бросаем вызов!

Посоревнуйтесь с Михалисом! Отправьте свой код по адресу электронной почты [LXF: linuxformat@futurenet.com](mailto:linuxformat@futurenet.com) с темой: **Загадка LXF226**. Убедитесь, что ваш код компилируется без ошибок (также не забудьте хорошо задокументировать код и снабдить его подробными комментариями), и вы получите шанс обойти нашего умника.

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char **argv)
{
    if (argc != 3)
    {
        printf("Неверное число аргументов. Выходим...\n");
        return -1;
    }
    int seed = atoi(argv[1]);
    int count = atoi(argv[2]);
    srand(seed);
    int i = 0;
    for (i=1; i<=count; i++)
        printf("%d ", rand());
    printf("\n");
    return 0;
}
```

Как видите, программа на C относительно невелика. Однако в большинстве скриптовых языков создание случайных чисел занимает не более пяти строк кода.

Сначала вы считываете два аргумента командной строки, а затем вызываете функцию `srand()`, которая устанавливает начальное значение («затравку [seed]») для генератора случайных чисел. После этого вы просто вызываете функцию `rand()` столько раз, сколько случайных чисел нужно создать.

Если вы используете одну и ту же «затравку», то и результаты каждый раз будут одинаковыми. Это превращает весь процесс в неслучайный, и такие числа легко взломать:

```
$ ./random 123 4
128959393 1692901013 436085873 748533630
$ ./random 123 3
128959393 1692901013 436085873
$ ./random 123 5
128959393 1692901013 436085873 748533630 776550279
```

Как видите, при использовании одной и той же затравки первое число для всех последовательностей случайных чисел всегда одно и то же, поэтому все последовательные числа тоже получаются одинаковыми. Итак, можно ли назвать программу `random.c` относительно хорошим генератором случайных чисел? Ответ — да, если вы не используете одну и ту же затравку постоянно. Мы видим, что иногда мелкие детали имеют значение и следует проявлять большую осторожность. Тем не менее, существуют и другие способы настройки генератора случайных чисел. Об этом мы поговорим в следующем разделе.

```
2. mtsouk@mail: ~/docs/article/working/puzzle1.LXF226/code (ssh)
code$ gcc -Wall randomPass.c -o randomPass
randomPass.c: In function 'main':
randomPass.c:24:14: warning: unused variable 'myRead' [-Wunused-variable]
    uint16_t myRead = read(getSeed, &mySeed, sizeof(mySeed));
                   ^
code$ ./randomPass
Please give me length of password.
code$ ./randomPass 10
SDXe.r5nhX
code$ ./randomPass 100
RxAWV.%1VY2J'W6+TqaBX~'uE,Np&,.W$MhSWe>.Y[rA2Ag 2IAha wEFJJJPW;1$y{t'8<9q,v>
0;0Yc}08i23}0.
code$ ./randomPass 300
Vza-pC...-l)seN\~TIV\51#\k3G_g1<-+7b6n*'}'Qqj9M0k5-HH0>cy0+'PT\'x-gH-oo700(Sr
bAj$.Yx3Q24nW$'m2AsE3{'*9Bb{+WV(PG*\-<*iY<zK>z8iUEH'AI+X%'m.vI4F*V#Bp++dE%/I
5q8";.dlS1=~/~8+VV81nuk[x3IU Z%-p?E]#;+jFr7l%gus=H'EWK ivb?2=-\~yzD7X#?J.TF
0Fe2}ugYLrmFQ$13A-fs[-b;fm@>{x' Bw7?(13L+~
code$ ./randomPass 10
jqQ3mID>8k
code$ ./randomPass 10
mGvL_@}n^
code$
```

► Рис. 2. Вывод программы `outputPass.c`, а также команда, используемая для ее компиляции.

## Поговорим о безопасности

Цель этого урока довольно сложна: вы хотите повысить безопасность своей компании или дома, и вам нужен способ получить более безопасные пароли, трудные для угадывания. Идея состоит в том, чтобы написать программу на любом языке программирования, которая будет делать это за вас.

Сначала эта задача может показаться простой, но ее решение может быть сложным, так что не стоит недооценивать простые задачи. Было бы неплохо подумать об этой задаче самостоятельно и, возможно, написать несколько небольших программ, но не мешает обратиться и к нашему уроку за рекомендациями.

```
1  #!/usr/bin/env python3
2
3  import sys
4  import random
5
6  if len(sys.argv) != 2:
7      print("I need just one argument!")
8      sys.exit(0)
9
10 if sys.argv[1].isdigit():
11     count = int(sys.argv[1])
12 else:
13     print(sys.argv[1], "is not an integer")
14     sys.exit(0)
```

► Рис. 3. Python 3 творит чудеса несколькими строками кода! Скрипт `randomPass.py`.

На рис. 1 показан результат нескольких запусков программы `random.c`.

### Еще случайнее

Код этого раздела (на C) будет сохранен в файле `randomPass.c`. Его общая идея основана на коде из файла `random.c`, используемого для генерации случайных чисел. Однако на сей раз начальное значение для генератора случайных чисел не задается как аргумент командной строки, что является очень распространенной практикой.

Существует множество способов выбора начального значения, которые включают использование текущего времени или сочетания текущего времени с идентификатором сети, а также использование устройства UNIX, такого как `/dev/random`, способного генерировать случайные числа без необходимости писать какой-либо код. Кроме того, мы будем выводить не цифры, а символы ASCII — к чему мы и стремимся. В файле `randomPass.c` важен следующий код:

```
char c = FROM + rand() % TOTAL + 1;
printf("%c", c);
```

Как вы видели в предыдущем коде, C дает самый простой способ преобразования целого числа в символ ASCII, поскольку в C символы ASCII рассматриваются как целые числа, и наоборот (но не следует забывать, что подобная гибкость способна стать источником кучи неприятных ошибок).

Также интересно посмотреть, как мы получаем начальное значение с помощью устройства `/dev/random`:

```
int getSeed = open("/dev/random", O_RDONLY);
unsigned short mySeed;
uint16_t myRead = read(getSeed, &mySeed,
sizeof(mySeed));
```

### Скорая помощь

Исчерпывающее руководство по теории случайных чисел написал Дональд Кнут. Это второй том его монографии «Искусство программирования». За время, потраченное на чтение его книги, вы станете отличным программистом.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)

Вы просто читаете `/dev/random`, как будто это файл. На рис. 2 показан результат нескольких запусков программы `randomPass.c`, которая принимает всего один аргумент — длину генерируемого случайного пароля.

Мы создали рабочую версию программы на C; попробуем реализовать ее на других языках программирования. Это можно сделать по разным причинам: во-первых, из-за радости узнавания, а во-вторых, чтобы сравнить различные реализации и определить преимущества и недостатки каждого языка программирования. Итак, далее мы напишем программы на языках Python 3, Perl и Go. Все они будут генерировать случайные пароли, используя те же принципы, что были применены в программе на C.

**Скорая помощь**

Желая сделать нечто поинтереснее, можете превратить скрипт `dictPass.py` в веб-приложение или поместить его в другой скрипт, который автоматически отправляет сгенерированный пароль пользователю электронной почтой или по SMS.

## Случайность в Python 3

В этом разделе будет представлена программа на Python 3, основанная на том же самом методе, который применялся в файле `randomPass.c`. Имя скрипта Python 3 — `randomPass.py`, и в нем важен следующий код:

```
for i in range(0, count):
    while True:
        randomNumber = random.randrange(MIN, MAX)
        char = str(chr(randomNumber))
        if char.isspace() == False:
            break
    print(char, end="")
```

Этот код не использует оператор деления по модулю, поскольку функция `randrange()` и модуль `random` позволяют генерировать случайные числа в желаемом диапазоне. В результате программа выполняется быстрее, поскольку в ней меньше команд.

Вдобавок в Python 3 есть очень удобная функция `isspace()`, которая позволяет узнать, является ли символ пробелом, и исключить его из вывода. Наконец, модуль `random` в Python 3 создает заставку автоматически.

При запуске программы `randomPass.py` будет получен следующий результат:

```
$. /randomPass.py 20
eK1M;w#*F@ludpm&W/G>
$. /randomPass.py 20
M8w[o/JROb&A_""K)Bc>
```

На рис. 3 показан код скрипта `randomPass.py`. Вообще говоря, для решения одной и той же задачи в языках скриптов требуется меньше кода, чем в языках программирования. Однако исполняемые файлы компилируемых языков программирования, как правило, выполняются быстрее, чем скрипты. Если у вас есть время, можете попытаться замерить производительность каждой программы, представленной на нашем уроке. Лучший способ сделать это — изменить код каждой программы так, чтобы она генерировала 100 000 случайных паролей, и использовать утилиту `time(1)` для измерения времени выполнения каждой программы. Экспериментируя, всегда старайтесь узнать как можно больше.

## Случайные пароли в Perl

Perl — еще один хороший язык для генерации случайных паролей на основе случайных чисел, и в этом разделе мы напишем программу `randomPass.pl`, которая позволяет создавать случайные пароли. Скрипт `randomPass.pl` имеет относительно небольшой размер, но делает свое дело хорошо и принимает только один аргумент командной строки — длину создаваемого пароля

В файле `randomPass.pl` наиболее важен следующий код:

```
print chr($OFFSET + int(rand($MAX)));
```

Вся работа выполняется функцией `chr()`, которая принимает целое число и возвращает символ. После преобразования случайного числа в символ ASCII его надо просто вывести на экран.

Запуск программы `randomPass.pl` выдаст следующий результат:

```
$. /randomPass.pl 20
[wPyWpJ6iUZ#%!F:p45
$. /randomPass.pl 10
,H,4|>|G)a
```

Решение задач с использованием небольших скриптов на любом скриптовом языке — неплохая идея: это экономит ваше время и усилия.

## Безопасный Go

Go — современный язык программирования, вполне пригодный для создания случайных паролей. В этом разделе мы напишем программу под названием `randomPass.go`. В ней наиболее важен следующий код:

```
for i = 0; i < LENGTH; i++ {
    anInt := int(random(MIN, MAX))
    newChar := string(startChar[0] + byte(anInt))
    if newChar == " " {
        i = i - i
        continue
    }
    fmt.Print(newChar)
```

Как видите, в Go символы ASCII обрабатываются довольно странным образом.

Изучение таких вещей при написании простых приложений ради развлечения — огромное преимущество для каждого программиста, так как это облегчит вам жизнь при разработке более крупных приложений на Go в будущем.

```
1 package main
2
3 import (
4     "fmt"
5     "math/rand"
6     "os"
7     "path/filepath"
8     "strconv"
9     "time"
10 )
11
12 var MAX int = 90
13 var MIN int = 0
14
15 func random(min, max int) int {
16     return rand.Intn(max-min) + min
17 }
18
19 func main() {
20     if len(os.Args) != 2 {
21         fmt.Printf("usage: %s length\n", filepath.Base(os.Args[0]))
22         os.Exit(1)
23     }
24
25     LENGTH, _ := strconv.ParseInt(os.Args[1], 10, 64)
26     rand.Seed(time.Now().Unix())
27     startChar := "!"
28     var i int64
29     for i = 0; i < LENGTH; i++ {
30         anInt := int(random(MIN, MAX))
31         newChar := string(startChar[0] + byte(anInt))
32         if newChar == " " {
33             i = i - i
34             continue
35         }
36         fmt.Print(newChar)
37     }
38     fmt.Println()
39 }
```

➤ Рис. 4. Код программы `randomPass.go`. Go предлагает преимущества языка C без его осложнений.

➤ Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.



## О природе случайных чисел

Случайные числа используются в различных сферах, в том числе в создании случайных строк. Поэтому на нашем уроке сначала создаются случайные числа, а затем они используются для создания случайных строк. В итоге эти случайные строки послужат надежными паролями.

Строго говоря, последовательность чисел считается статистически случайной, если в ней нельзя найти какие-либо узнаваемые закономерности или шаблоны. Любая последовательность

случайных чисел начинается со значения, называемого «затравкой» (начальным значением). Затравка является самым важным параметром генератора случайных чисел, ведь она инициализирует весь процесс. Сама затравка не обязательно должна быть случайной или трудно определяемой. Обычно она генерируется с помощью системного времени или какого-либо другого свойства системы.

Хотя совершенно случайных генераторов не существует, в большинстве случаев хватает, чтобы

числа были «достаточно случайными», поскольку вам не требуется бесконечно большое количество этих чисел. Другими словами, вам не нужна 1000 случайных чисел, чтобы создать строку с пятью символами.

Учтите, что для успешной работы требуется хорошее знание теории, и этим не следует пренебрегать ради экономии времени. Лучше потратить время на изучение теории, чем на решение не той проблемы.

При запуске программы **randomPass.go** будут получены следующие случайные пароли:

```
$ go run randomPass.go 10
```

```
E=Iz$)&r.u
```

```
$ go run randomPass.go 20
```

```
Gz4&OKcbL>.=mC=b]EYJ
```

```
$ go run randomPass.go 30
```

```
*?S0"Yg-5-pZ_kUrajJA/Z-bgEsp|z
```

На рис. 4 показан весь код программы **randomPass.go**, который можно сравнить с другими реализациями.

## Файл словаря

После работы с этими примерами неплохо поиграть еще и попытаться реализовать некоторые безумные идеи. В этом разделе мы употребим несколько иной подход и в качестве отправной точки для создания паролей воспользуемся файлом словаря, который входит в состав компьютера с Linux.

Если файл словаря отсутствует, может потребоваться установить его на компьютер с Linux через ваш любимый менеджер пакетов. Так как эта программа принимает в качестве аргумента командной строки текстовый файл словаря, вы сможете использовать в качестве аргумента собственные текстовые файлы с подходящим списком слов.

В данном разделе для создания паролей будет использоваться Python 3, поскольку этот язык позволяет решать многие задачи без необходимости писать слишком много кода. Имя скрипта Python 3 — **dictPass.py**, и первое, что он делает — подсчитывает количество слов, найденных в файле словаря. Затем он создает случайное число и выбирает случайное слово, основываясь на этом числе.

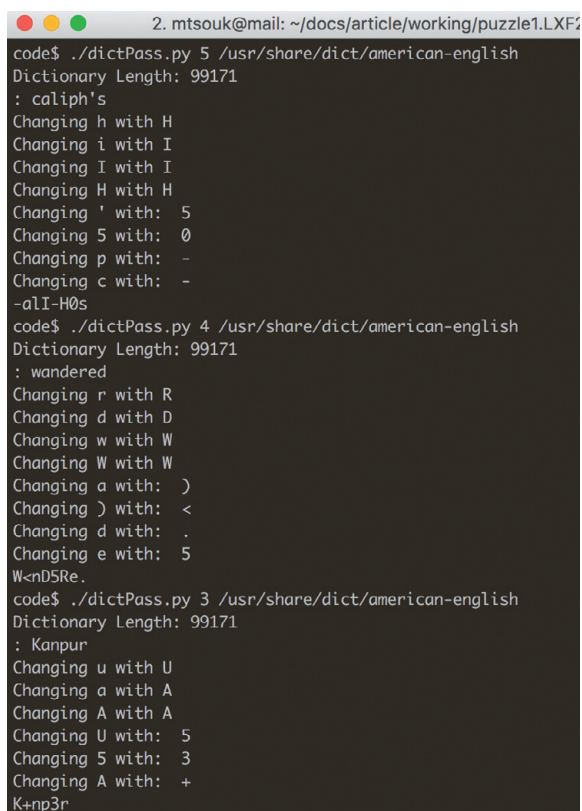
После выбора слова надо убедиться, что его длина превышает желаемое число символов в пароле. Если пароль меньше, программа создаст другое случайное число, выберет другое слово из словаря и будет делать это до тех пор, пока длина сгенерированного пароля не станет больше или равна желаемой. Поскольку текущий пароль основан на словарных словах, программа попытается изменить его, случайным образом заменив некоторые буквы цифрами, прописными буквами и другими символами. После этого пароль выведется на экран.

Все предыдущие шаги будут показаны в выводе скрипта **dictPass.py**, чтобы его работа была понятнее.

Основные функциональные возможности **dictPass.py** реализованы в следующем коде Python 3:

```
lineNumber = randint(1, length)
randomLine = file.getLine(lineNumber)
randomLine = randomLine.rstrip()
print(":", randomLine)
password = password + randomLine
```

Предыдущий код Python 3 создает первую версию пароля, выбирая слово из словарного файла.



```
2. mtsouk@mail: ~/docs/article/working/puzzle1.LXF
code$ ./dictPass.py 5 /usr/share/dict/american-english
Dictionary Length: 99171
: caliph's
Changing h with H
Changing i with I
Changing I with I
Changing H with H
Changing ' with: 5
Changing S with: 0
Changing p with: -
Changing c with: -
-aII-H0s
code$ ./dictPass.py 4 /usr/share/dict/american-english
Dictionary Length: 99171
: wandered
Changing r with R
Changing d with D
Changing w with W
Changing W with W
Changing a with: )
Changing ) with: <
Changing d with: .
Changing e with: 5
W<nD5Re.
code$ ./dictPass.py 3 /usr/share/dict/american-english
Dictionary Length: 99171
: Kanpur
Changing u with U
Changing a with A
Changing A with A
Changing U with: 5
Changing S with: 3
Changing A with: +
K+np3r
```

► Рис. 5. Вывод программы **dictPass.py**. Пароли получаются довольно изобретательными для столь небольшой программы.

Единственным недостатком файла **dictPass.py** является то, что он много раз читает файл словаря, и это может замедлить процесс. При запуске программы **dictPass.py** получится следующий результат:

```
$ ./dictPass.py 10 /usr/share/dict/american-english
voLu<43.1aCh)s
```

Как видите, скрипт **dictPass.py** генерирует довольно хорошие пароли — и это объясняется не тем, что используется Python 3, а умным и необычным способом генерации паролей. На рис. 5 показано несколько запусков программы **dictPass.py**, которые хорошо иллюстрируют, как далеко вас может завести ваше воображение.

Достаточно ли хороши созданные пароли? Можете ли вы улучшить программу? Можете ли вы предложить лучший метод создания более сложных паролей? Можете ли вы придумать более простой способ создания случайных строк с лучшими характеристиками? Напишите программу и отправьте ее нам, и, возможно, она будет опубликована в журнале.

Ограничивается ли создание случайных паролей этим уроком? Конечно, нет. Это только начало вашего путешествия по созданию более надежных паролей для вас, ваших друзей, вашей компании и членов вашей семьи. **LXF**

# π: Нарисуем в виде графика

Михалис Цукалос рассчитывает число π на разных языках программирования, а затем отображает его как набор случайных цветных пикселей.



Наш эксперт

### Михалис Цукалос

Михалис — администратор UNIX, программист-математик и администратор баз данных, который любит писать статьи. С ним можно связаться через Twitter: @mactsouk.



```
2_mtsouk@mail:~/docs/article/working/plotPiLXF226/code (ssh)
code$ time ./savePi.py 1000000 > /dev/null

real    1m18.183s
user    1m17.480s
sys     0m0.110s
code$ time ./savePi.py 10000000 > /dev/null

real    151m44.202s
user    149m41.532s
sys     0m7.400s
code$
code$ ./savePi.py 100
code$ ./savePi.py 1000
code$ ./savePi.py 10000
code$ ./savePi.py 100000
code$ ./savePi.py 2000
code$ ./savePi.py 1000000
code$ ./savePi.py 10000000
```

Рис. 1. Файл savePi.py в действии, а также вывод утилиты time(1) с оценкой времени выполнения файла savePi.py на сравнительно медленном компьютере с Linux.

необходимости прочитать значение числа π. Именно эти действия выполняет скрипт Python 3 с именем **savePi.py**.

Помните, что скрипт **savePi.py** использует всего один параметр — количество знаков после запятой, которые нужно получить. Скрипт выведет результат на экран, и чтобы этот результат сохранить, надо перенаправить стандартный вывод в текстовый файл.

В Python 3 масса пакетов для научных расчетов, и будет разумно использовать один из них для вычисления π. В данном случае мы выберем пакет **mpmath**. Если этот пакет не установлен в системе, его легко установить (от имени root):

```
# pip3 install mpmath
Установив пакет mpmath, мы запросто вычислим π — код Python 3 файла savePi.py приведен ниже:
```

```
#!/usr/bin/env python3
import sys
from mpmath import mp
if len(sys.argv) == 1:
    print("Не хватает аргументов!")
    sys.exit(0)
if sys.argv[1].isdigit():
    accuracy = sys.argv[1]
    mp.dps = accuracy
    print(mp.pi)
```

Желаемая точность задается с помощью свойства **mp.dps**. Также следует вывести на экран **mp.pi**. Чтобы запустить скрипт **savePi.py** и сохранить вывод, достаточно выполнить команду

```
$. /savePi.py 1000000 > pi1M.txt
```

Во всех реализациях созданный текстовый файл будет необходимо читать посимвольно — или, точнее, цифру за цифрой. Программам на Perl, Python 3 и Go потребуется четыре аргумента

В этом месяце мы продолжим исследовать Python и построим цифры числа π в виде изображения, на сей раз в формате PNG. Однако это сложнее, чем кажется, поскольку вам надо знать, как читать и записывать файлы, а также как указать цвет пикселя в изображении.

Чтобы всё стало еще интереснее, мы представим полные примеры кода на Python 3, Perl и Go. Кроме того, вы увидите код на Julia и R, посимвольно считывающий текстовый файл, и сможете почувствовать разницу между языками программирования общего назначения и языками программирования, созданными для научных вычислений.

π — иррациональное число. Это означает, что его цифры никогда не заканчиваются. Способов расчета числа π с заданной точностью существует масса, а на нашем уроке мы применим для этого модуль Python 3, что позволит избежать сложных вычислений. В следующем номере мы обсудим самостоятельное вычисление знаков числа π.

### Спасти рядового π

Самое удобное, общее и простое решение — сохранить число π в текстовом файле, который будет использоваться при каждой

### Скорая помощь

Хотя Image::Magick — популярный модуль Perl, его установка может оказаться непростой. Кроме того, функция ReadImage() из этого модуля может привести к «падению» ваших скриптов. Поэтому мы воспользовались модулем Image::Base::GD.

## Проблема точности числа π

Существуют различные алгоритмы, основанные на математических формулах и позволяющие рассчитывать число π с требуемой точностью и контролировать процесс. Основной недостаток такого подхода — то, что вам придется самостоятельно писать код. Зато, обладая собственной реализацией алгоритма, код будет проще оптимизировать и ускорить.

Список методов расчета числа π с требуемой точностью включает формулу Бэйли–Боруэйна–Плуффа [Bailey–Borwein–Plouffe], формулу Бельяра [Bellard], алгоритм Боруэйна, алгоритм Чудновского, метод быстрого вычисления E-функций, алгоритм Гаусса–Лежандра [Gauss–Legendre], формулу Лейбница [Leibniz], алгоритм Лю Хуэя [Liu Hui], формулу Виета [Viète], формулу Уоллиса

[Wallis] и алгоритм Чжао Юцина [Zhao Youqin]. Большинство из них считают сумму последовательности чисел — чем больше взять чисел, тем точнее результат.

Как видите, множество математиков из разных стран расценивают расчет числа π как интересную задачу, на которую стоит потратить время и силы. Надеюсь, вы согласитесь с ними.

командной строки: ширина и высота результирующего изображения, название выходного изображения и имя текстового файла с цифрами числа π. В программах на R и Julia параметры командной строки для простоты не используются.

На рис. 1 (стр. 88) показан вывод файла **savePi.py** при расчете числа π со сравнительно небольшой точностью, а также вывод команды `time()`, которая оценивает длительность выполнения **savePi.py** при вычислении числа π с 1 000 000 и 10 000 000 знаков после запятой на сравнительно медленном компьютере. Как видите, последнее вычисление заняло около 150 минут...

### Хватает ли цифр?

Количество цифр напрямую повлияет на размер графика. В общем-то, строить график для 10 или даже 100 знаков π нецелесообразно — для лучшего изображения нужно больше знаков.

Таким образом, для изображения с разрешением 800 × 600 пикселей потребуется 480 000 цифр π — недооценивать полезность пакета *mpmath* явно не следует. Наконец, в реализации на Python 3 и Perl будет использоваться массив с 10 элементами, и каждый элемент будет представлять собой один из цветов, используемых для построения графика. Если вы хотите изменить цвет в результирующем изображении, надо изменить этот массив.

Основная задача, которую надо решить в каждой реализации, заключается в том, чтобы определить изображение с желаемыми размерами и закрасить каждый пиксель изображения заданным цветом, а затем сохранить изображение на диске. В каждом языке программирования используется собственный подход и собственные пакеты для решения задачи, именно поэтому так интересно создать реализации на разных языках.

У каждого пикселя есть две координаты: x и y. Значение координаты x является частным от целочисленного деления количества обработанных на данный момент цифр на высоту изображения. Чтобы определить значение координаты x, нужно найти остаток от целочисленного деления количества обработанных на данный момент цифр на высоту изображения.

### Поработаем в Python 3

Программа на Python 3 содержится в файле **pPi.py**. Но сначала вы узнаете, как создавать PNG-изображения в Python 3 с помощью модуля PIL. В следующем коде Python 3, который сохранен в файле **tenten.py**, показано, как создать собственное изображение 10 × 10 с использованием заданных цветов. Тогда вы лучше поймете, как устроена программа в файле **pPi.py**:

```
#!/usr/bin/env python3
from PIL import Image
n = 10
m = 10
filename = "test.png"
image = Image.new('RGB', (n, m))
```

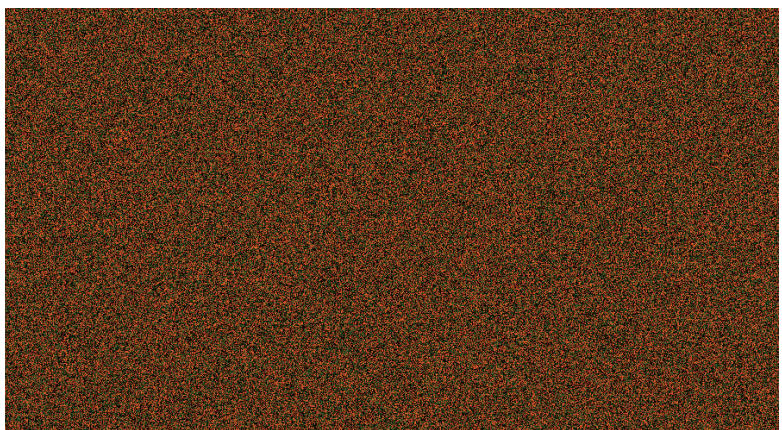
```
image.save(filename, "PNG")
img = Image.open(filename)
pix = img.load()
print(img.size)
print(pix[5,5])
for x in range(1,10):
    for y in range(1,10):
        pix[x, y] = x*y*x
img.save("test.png")
```

Оператор `pix[x, y]` определяет координаты интересующего вас пикселя, а значение `x*y*x` определяет цвет пикселя, который в этом случае является случайным. Теперь, когда у нас есть код файла **tenten.py**, остается решить еще две задачи. Первая заключается в чтении цифр π из текстового файла по одной, а вторая — назначает соответствующий цвет каждому пикселю на основе текущей цифры. Наиболее важным кодом в файле **pPi.py** является код для чтения числа π из файла по одной цифре:

```
f = open(textPi, 'r')
while True:
    ch = f.read(1)
    if not ch:
        break
```

Приведенный код на Python 3 продолжает чтение с помощью команды `f.read()` и сохраняет текущий символ в переменной `ch`. На рис. 2 показан вывод файла **pPi.py** при создании изображения шириной 900 пикселей и высотой 800 пикселей с использованием следующей команды:

```
$. /pPi.py 900 800 python.png pi1M.txt
900 800 named python.png - reading from pi1M.txt
python.png Image size: (900, 800)
Read: 720000 digits
```



» Рис. 2. Вывод скрипта **pPi.py** при создании файла PNG высотой 900 пикселей и шириной 800 пикселей.

### Скорая помощь

Не стоит недооценивать глубину небольших задач. На нашем уроке вы научились считывать текстовые файлы построчно на пяти разных языках программирования и создавать изображения по пикселю на трех разных языках программирования.

## π на Perl

Реализация на Perl сохранена в файле **pPi.pl**, но сначала вы узнаете, как создать изображение PNG размером 10×10 с помощью кода из файла **10x10.pl**:

```
#!/usr/bin/perl -w
use strict;
use warnings;
use Image::Base::GD;
my $x = 10;
my $y = 10;
my $outputfilename = "test.png";
my $im = Image::Base::GD->new(-width => $x, -height => $y);
my @color = ('#000000', '#118800', '#11BB00', '#1100FF',
'#1188FF', '#11BBFF', '#11FFFF', '#99FFFF', '#CCFFFF', '#FFFFFF');
for (my $i=0; $i<$x; $i++) {
    for (my $k=0; $k<$y; $k++) {
        my $index = ($i * $k) % 10;
        $im->xy($i, $k, $color[$index]);
    }
}
$im->save($outputfilename);
exit 0;
```

Таким образом, функция `xy()` позволяет добавить новый пиксел в изображение, созданное с помощью функции `Image::Base::GD->new()`.

С учетом этого кода легко реализовать недостающую функциональность и создать рабочий файл **pPi.pl**, потому что единственное, чего нам не хватает — это возможность считывания текстового файла посимвольно и возможность установки нужного цвета для каждого пикселя изображения на основе заранее определенных цветов.

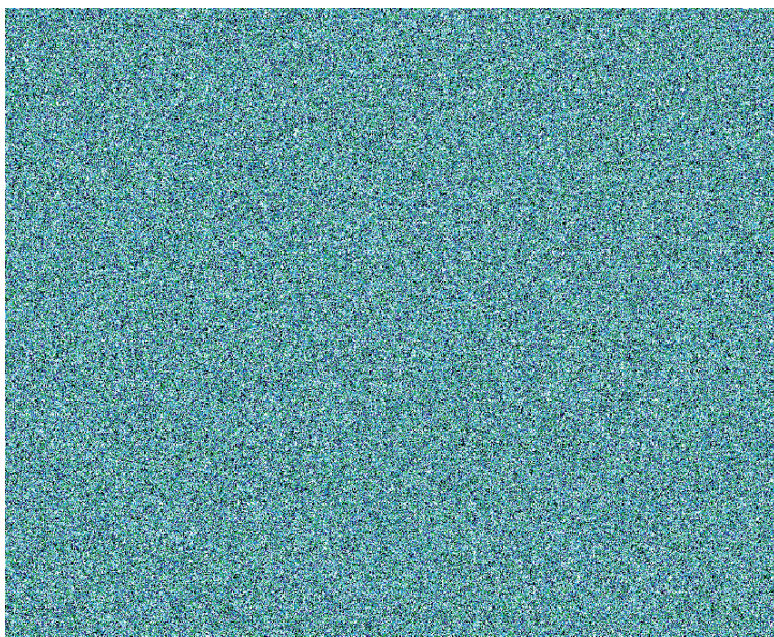
Следующий код Perl, сохраненный в файле **pPi.pl**, показывает, как считать содержимое текстового файла посимвольно:

```
my $read;
my $char;
while ($read = read PI, $char, 1) {
}
```



**Скорая помощь**

Начинать лучше с небольших примеров. Как вы видели на нашем уроке, окончательная версия программы создается не сразу. Желание реализовать сразу все функции программы сильно осложнит вашу жизнь.



➤ Рис. 3. Здесь показан вывод скрипта **pPi.pl** Perl для создания изображения с размером 800×900 пикселей.

Это происходит потому, что мы используем небольшой буфер для чтения (размером всего в один байт), что заставляет программу читать по одному байту за раз. На рис. 3 (внизу) показан вывод файла **pPi.pl** при создании изображения шириной 800 пикселей и высотой 900 пикселей, следующей командой:

```
$ ./pPi.pl 800 900 perl.png pi1M.txt
Image 800 x 900 - output: perl.png - input: pi1M.txt
Read 720001 digits.
$ ls -l perl.png
-rw-r--r-- 1 mtsouk staff 303668 Apr 25 21:41 perl.png
```

## Вычисляем π на Go

Хотя Go считается языком системного программирования, в составе его стандартной библиотеки есть пакеты, позволяющие создавать и обрабатывать PNG-файлы, а также другие популярные графические форматы файлов. На сей раз сначала покажем, как в Go посимвольно считываются текстовые файлы. Файл **cByC.go** содержит следующий код:

```
package main
import (
    "bufio"
    "fmt"
    "io/ioutil"
    "strings"
)
func main() {
    filename := "pi1M.txt"
    buf, _ := ioutil.ReadFile(filename)
    in := string(buf)
    s := bufio.NewScanner(strings.NewReader(in))
    s.Split(bufio.ScanRunes)
    for s.Scan() {
        fmt.Println(s.Text())
    }
}
```

Код для посимвольной обработки текстового файла на языке Go сложнее, чем ожидалось, особенно если сравнить его с кодом R, представленным в следующем разделе. Однако код Go для создания изображения в формате PNG довольно прост, так как использует пакеты стандартной библиотеки Go.

Чтобы создать изображение PNG в Go, надо выполнить следующую команду. Она создает изображение высотой `x` пикселей и шириной `y` пикселей:

```
m := image.NewRGBA(image.Rectangle{Min: image.Point{0, 0},
Max: image.Point{x, y}})
```

Затем, чтобы задать цвет заданного пикселя, добавим такую команду (где `x` и `y` — координаты пикселя, а `c` — название желаемого цвета):

```
m.Set(xx, yy, c)
```

Наконец, в файле **pPi.go** для получения желаемого цвета используется функция `getColor()`, вместо хранения значений цвета в массиве. На рис. 4 (см. стр. 91) показан вывод файла **pPi.go** при создании изображения шириной 1000 пикселей и высотой 1000 пикселей с использованием следующей команды:

```
$ go run pPi.go 1000 1000 go.png pi1M.txt
Количество знаков: 1000001
```

На рис. 5 (внизу стр. 91) показана эффективность каждой реализации. Это позволяет понять, насколько хорошо справляется с задачей каждый язык программирования. Программа на Python 3 является самой быстрой из всех, за ней следует программа на Go, а самая медленная программа — на Perl. Правда, на нашем компьютере Debian используется более старая версия Go. Возможно,

➤ Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.

## Вывод утилиты time()

Поскольку в этом руководстве используется утилита `time()`, было бы неплохо узнать, как интерпретировать ее вывод. Итак, утилита `time()` выполняет указанную команду как обычно, одновременно собирая статистику о ее выполнении.

Статистика выводится по завершении работы команды. Временная статистика представлена в виде трех значений: реального (`real`),

пользовательского (`user`) и системного (`sys`). Реальное значение (`real`) показывает фактическое время, которое потребовалось для выполнения команды от начала до конца. Пользовательское значение (`user`) — это время, затрачиваемое процессором на выполнение кода в пользовательском режиме, а системное значение (`sys`) — это время процесса, которое программа потратила на выполнение

кода в режиме ядра. Сложив значения `user` и `sys`, вы получите общее процессорное время. Однако для компьютеров с несколькими процессорами и ядрами процессорное время может превышать значение `real`.

Помните, что вывод утилиты `time()` может зависеть от текущей загрузки операционной системы Linux, в которой вы работаете.



► Рис. 4. Вывод программы `pPi.go` для создания изображения размером `1000 × 1000` пикселей.

из-за этого программа на Go уступила в производительности программе на Python 3.

## Julia ненавидит π

Julia — это язык программирования, более пригодный для научных и числовых расчетов. Программа на языке Julia будет сохранена в файле `pPi.julia`, и она иллюстрирует только посимвольное считывание текстового файла. Затем вы сможете построить график, используя свой любимый пакет Julia. Однако учтите, что это не всегда просто, так как Julia не умеет работать с файлами PNG.

Код Julia для посимвольного считывания текстового файла с цифрами числа π выглядит следующим образом:

```
open(input,"r") do f
  global digitCounter
  global x
  global y
  while !eof(f)
    if digitCounter > x * y
      break
    end
    c = read(f, Char)
    if c == '.'
      skip
    else
      xx = div(digitCounter, x)
      yy = digitCounter % x
      digitCounter += 1
    end
  end
end
```

Важнейшим кодом в этом файле является вызов функции `read()` с указанными параметрами. Чтобы просмотреть результат работы файла `pPi.julia`, запустите его.

## π на R

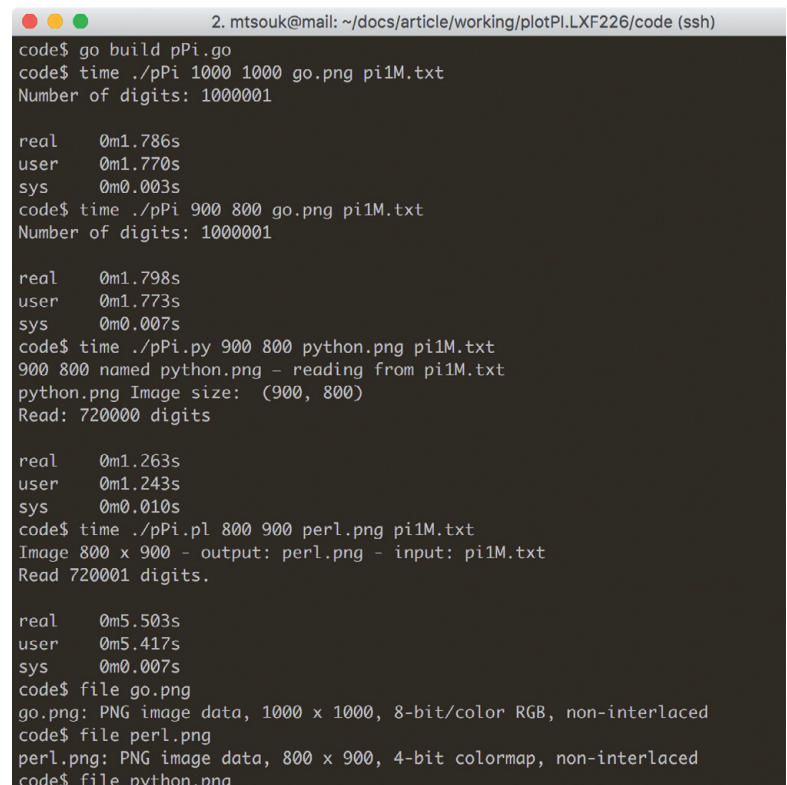
R — еще один язык программирования, разработанный для научных и статистических вычислений. В этом разделе мы покажем, как посимвольно прочесть и обработать текстовый файл на R (на чем и закончим). Имя скрипта на R — `pPi.R`, и в нем наиболее важен следующий код:

```
digits <- readLines("~/docs/article/working/plotPi.LXF226/code/pi1M.txt")
A <- strsplit(gsub("\\s+", "", digits), "")[[1]]

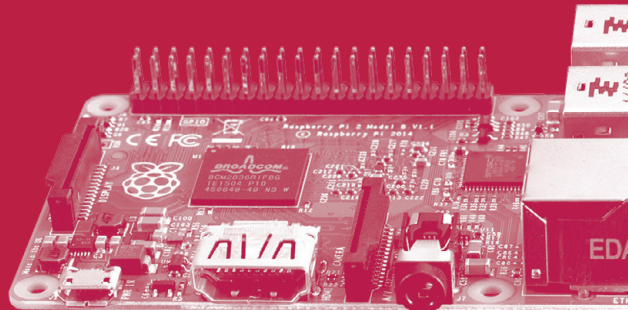
> A
[1] "3" "." "1" "4" "1" "5" "9" "2" "6" "5" ...
> length(A)
[1] 1000001
```

Итак, `A` — это большой объект с `1000001` элементом, которые являются десятичными знаками числа π! Чтобы увидеть результат работы файла `pPi.R`, запустите его.

Хотя языки R и Julia предоставляют отличные возможности для построения графиков, способ создания и обработки PNG-файлов в Go, Python 3 и Perl более простой и гибкий. [LXF](#)



► Рис. 5. Информация о производительности программ на языках Go, Perl и Python 3, используемых для визуализации числа π. Получена с помощью утилиты `time (1)`.



Ваша порция смачных новостей, обзоров и учебников от Raspberry Pi

**ЛЕС ПАУНДЕР**  
наш постоянный  
эксперт  
по рiоркам.  
Он также пишет  
о Raspberry Pi.



Привет...

**Н**едavno я написал программу, чтобы использовать датчик температуры DS18B20 с Python через какую-нибудь функцию. В основе программы был код для Cam Jam Edu Kits, который я использовал почти два года.

Программа была довольно громоздкая, и температура выводилась только в оболочке Python. Но из случайной беседы с Беном Наттолом [Ben Nuttall] из Raspberry Pi Foundation я узнал о библиотеке с именем *W1ThermSensor*, которая делает всю черную работу! Эту потрясающую библиотеку написал Тимо Фуррер [Timo Furrer], она абстрагирует шаги считывания однопроводного датчика, отображаемого как файл в файловой системе Linux. Посмотреть и попробовать ее можно на <https://github.com/timofurrer/w1thermsensor>. Далее, чтобы использовать этот датчик, надо контакт 3 В (VCC) датчика подключить через резистор 4,7 кОм к контакту вывода, подав на него высокое напряжение. Я использовал самодельную перемычку и употребил для соединения контактов колодку с клеммами. Но потом, поговорив с Дейвом Джонсом [Dave Jones], создателем библиотеки *Pisamera* Python, изменил свой подход. К GPIO надо подключать выходной контакт, и он должен быть под напряжением. Для активации и деактивации контактов по команде можно применить библиотеки Python, GPIO Zero и Rpi.GPIO. И внешний резистор не нужен.

Соль здесь в том, что даже если вы действуете правильно, есть и другие подходы. И они могут быть лучше, так что перенимайте их и учитесь на них. А не подойдет — отложите и вернитесь к своему проекту.

## Облачные вычисления делают погоду в доме

Безымянный мастер вызвал бурю на рабочем столе.

**Н**екий Modern Inventor со своим самодельным штормовым указателем вывел создание метеопрогнозов (или, на худой конец, метеосводок) в настольной среде на новый уровень. В основе его лежит Raspberry Pi Zero W, который с помощью сервиса Weather Underground получает прогноз погоды в любой точке мира. Затем эту погоду можно воссоздать внутри колбы. Дождь через центрифужный насос проходит к крышке и оттуда капает вниз.

Но нам больше всего понравилось создавать облака с помощью ультразвукового диффузора/увлажнителя. За счет высокой частоты вибрации диафрагмы водяной пар конденсируется в капельки, которые заполняют колбу (нечто вроде бутылки для воды) прозрачным туманом. В приборе также используется светодиодное кольцо Neopixel и Speaker Vonpet для визуальной и звуковой обратной связи,

и теоретически он может интегрироваться с Alexa Voice Service. И всё это с помощью лишь нескольких мазков горячего клея и строк Python. Суммарные затраты — где-то \$80, а инструкцию, как это делается, см. на сайте [www.instructables.com/id/Storm-Glass-the-Weather-Forecasting-Lamp-That-Simu](http://www.instructables.com/id/Storm-Glass-the-Weather-Forecasting-Lamp-That-Simu).



## Рi-золяция

Руками не трогать.

**К**орпуса для Pi — пятак за пучок, но этот — просто новое слово в изоляции: Raspberry Pi3 в параллелепипеде из эпоксидной смолы. Это творение Рональда Мак-Коллама [Ronald McCollum] из resin.io, создающего инструменты для использования в IoT-устройствах на базе Linux. О том, какие методы он использует, читайте здесь: <https://resin.io/blog/resin-in-resin-how-to-make-a-literal-embedded-device>.



## Вредители в Pi

Многие Pi — многия печали.

**Д**о Raspberry Pi добралось вредоносное ПО, в форме Linux.MulDrop14. Как утверждает российский сайт кибербезопасности Dr.WEB, ПО использует данные для входа в систему, устанавливает неизвестную систему добычи криптовалюты, *zmap* и *ssh*, и пытается передать ее другим Pi через SSH. Чтобы вылечить зараженные машины, надо очистить их SD-карты. Подробности на <http://bit.ly/2sAqMwC>.





# Ubuntu Mate 16.04

Да, запустить Ubuntu на Raspberry Pi можно, но будет ли это полноценной альтернативой дистрибутиву Raspbian? Лес Паундер попытался разобраться.

## Вкратце

» Ubuntu 16.04.2 с рабочим столом Mate, рекомендуем для менее мощных машин. Включает все те приложения и технологии Ubuntu, которые мы так любим, а также ряд опций и библиотек конфигурации Raspberry Pi, позволяющих умельцам использовать Ubuntu для своих проектов.

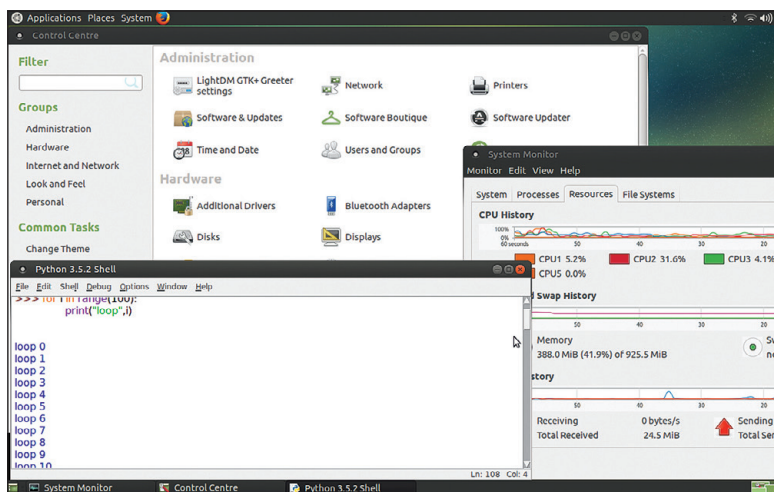
Ubuntu с нами уже немало лет, но только в 2015 г., с момента выхода Raspberry Pi 2, появилась версия и для этого маленького одноплатного компьютера. Столь долгое ожидание объясняется тем, что для этого популярного дистрибутива требовался процессор ARM v7 и увеличение памяти.

Теперь у нас есть Raspberry Pi 3 с более мощным процессором, и вот перед нами Ubuntu 16.04.2, с облегченным рабочим столом Mate — продолжением проекта Gnome 2, по-прежнему популярной настольной средой, подходящей для более старых и менее мощных машин.

Установка образа происходит аналогично Raspbian, то есть надо записать его на чистую SD-карту. Команда Ubuntu Mate рекомендует для достижения наилучших результатов использовать SD-карту класса 6/10.

Установив образ и вставив карту в Pi, вы производите первую загрузку и настраиваете себе свою учетную запись пользователя. По завершении вы попадаете на рабочий стол. Выглядит всё скромно, но очень профессионально; в левом верхнем углу экрана, как обычно, расположено меню с ярлыками приложений и доступа к нашей SD-карте и внешним дискам. В правом верхнем углу экрана — меню Bluetooth и Wi-Fi.

Ubuntu Mate совместим с Pi3-пакетами Wi-Fi и Bluetooth, но сразу после настройки учетной записи пользователя мы заметили ошибку: в меню Wi-Fi не указан SSID. Перезагрузка помогла, но такого, конечно, быть не должно.



» Выглядит и ощущается он точно так же, как и Ubuntu Mate на других машинах, но здесь есть еще и все специфические функции Pi.

Ubuntu Mate отстает от Raspbian в мелочах, например, в том, что Python требует доступа `sudo` для GPIO. Но для нас это был тест номер один, и мы рады сообщить, что проекты Python можно кодировать с помощью редактора *IDLE Python* так же, как и в Raspbian. Популярные библиотеки *GPIO Zero* и *Sense Hat Python* предустановлены и работают с Python 2 и 3; на самом деле, новый *Sense Hat Emulator* также установлен и полностью совместим с Ubuntu Mate, и можно попробовать виртуальную версию этой научной платформы для датчиков. Чтобы убедиться в полной работоспособности Python, мы также установили другую библиотеку, *GUI Zero*, используя инструмент пакетов Python, *pip3*, и всё прошло безупречно.

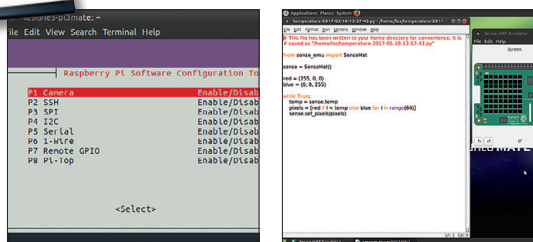
Затем мы протестировали *Sonic Pi* и *Minecraft*. Оба работают хорошо. С *Sonic Pi* возникла досадная проблемка — не работал регулятор громкости; но мы взамен воспользовались системными регуляторами, а *Minecraft* работал так же, как и на Raspbian. Код, чтобы внести изменения в мир *Minecraft*, мы написали без труда.

Следующим нашим тестом было использование официальной камеры Raspberry Pi с Ubuntu. В предыдущих релизах это было невозможно, но, активировав интерфейс с помощью инструмента *raspi-config* (еще одного отличного дополнения от Raspbian), мы смогли изготавливать снимки и видео с помощью терминала и через библиотеку Python. Как уже упоминалось, инструмент *raspi-config* предустановлен

и применяется для настройки множества разных протоколов и интерфейсов на Pi. Один из них — однопроводной интерфейс, используемый для датчиков, таких как датчик температуры DS18B20. Мы протестировали его с помощью библиотеки Python 3, и вновь всё получилось в два счета, не хуже, чем с Raspbian.

Ubuntu Mate наконец-то вышел из тени Raspbian. Это дистрибутив предназначен не только для технарей и мастеров-самоучек, каковых столь много в сообществе Raspberry Pi, но и на среднего пользователя: тех, кто использует этот дешевый компьютер в самых простых целях. Raspbian по-прежнему остается вариантом номер один, но Ubuntu Mate для Raspberry Pi уже обретает полноценность. **LXF**

## Свойства навскидку



### Настроится всё!

Включение *raspi-config* означает, что Pi можно легко настроить под любой ваш проект.

### Совместимость

Благодаря библиотекам для разных HAT, Ubuntu послужит прямой заменой Raspbian.

## LINUX FORMAT Вердикт

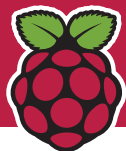
### Ubuntu Mate 16.04.2 LTS

Разработчик: Ubuntu Mate  
Сайт: [ubuntu-mate.org/raspberry-pi/](http://ubuntu-mate.org/raspberry-pi/)  
Лицензия: Open Source

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

» Наконец-то у Raspbian появился конкурент. И ориентирован он как на мастеров, так и на обычных пользователей.

## Рейтинг 9/10



# Pi Camera: Фото кадр за кадром

Лес Паундер расскажет, как создать несложный контроллер последовательной съёмки для камеры с аналоговой системой управления.



Наш эксперт

Лес Паундер — работающий на себя мастер-любитель и программист, который сотрудничает с Raspberry Pi Foundation и преподаёт в Pisacademy. Лес описывает свои приключения в блоге <http://bigl.es>.

Вам нужны

- » 40-контактный Raspberry Pi любой модели
- » Raspbian Pixel
- » Официальная камера Pi
- » 1 кнопка
- » 1 макетная плата
- » 1 АЦП MCP3008
- » 9-контактный кабель папа-папа
- » 8-контактный кабель папа-мама
- » 2 потенциометра 10к
- » Код проекта см. на <http://bit.ly/LXF226pitut>

Создавать серии снимков с помощью камеры Raspberry Pi легко, труднее добиться правильного освещения объекта. В этом проекте мы создадим контроллер для покадровой съёмки медленных процессов с аналоговым управлением яркостью и контрастностью. Кроме того, в сохраненных файлах также будет отметка времени, чтобы они правильно распознавались и их можно было просматривать в хронологическом порядке. Итак, начнем со сборки оборудования.

## Городим схему

Чтобы построить схему для этого проекта, надо разместить на макетной плате аналого-цифровой преобразователь MCP3008. Обратите особое внимание на выемку на одном из концов чипа: она указывает, где верх. Первый аналоговый вход Pin 1 расположен в левом верхнем углу микросхемы, если смотреть на чип при выемке в положении вверху.

Подключите MCP3008 к своему Raspberry Pi, как это показано на приведенной ниже схеме. Потенциометры подключаются к подводам 3 В и Земля [GND], и их выход запитывается на первые два вывода MCP3008. Кнопка подключается к GPIO и к GND-подводке.

Затем вставьте официальную камеру Raspberry Pi в порт камеры; он расположен либо между портом Ethernet и HDMI модели B, либо на конце платы Pi Zero — обратите внимание, что для этого нужен специальный адаптер. Ленту кабеля следует аккуратно вставить в синюю вкладку, находящуюся напротив порта Ethernet; не забудьте перед этим поднять пластиковый предохранитель, а затем вновь закрыть его, чтобы закрепить камеру на месте. Чтобы камера не касалась контактов GPIO, которые могут ее повредить, можно вмазать немного пластилина. Теперь подключите клавиатуру, мышь и т.д. и загрузитесь в рабочий стол своего Raspberry Pi.

## Начало работы

И на официальной камере Raspberry Pi, и на нашем MCP3008 надо настроить интерфейсы. Для этого перейдите в инструмент

настройки Raspberry Pi, который расположен в меню Настройки. На вкладке Интерфейсы включите Camera и SPI. Сохраните конфигурацию, после чего приложение попросит вас перезагрузить компьютер. Сделайте это и вернитесь на рабочий стол Raspbian.

Чтобы проверить камеру, откройте терминал и введите следующую команду, чтобы сделать снимок, который будет выводиться под именем **image.jpg**:

```
$ raspistill -o image.jpg
```

Если это не сработает, повторите настройку, выключите Pi и убедитесь, что кабель вставлен правильно.

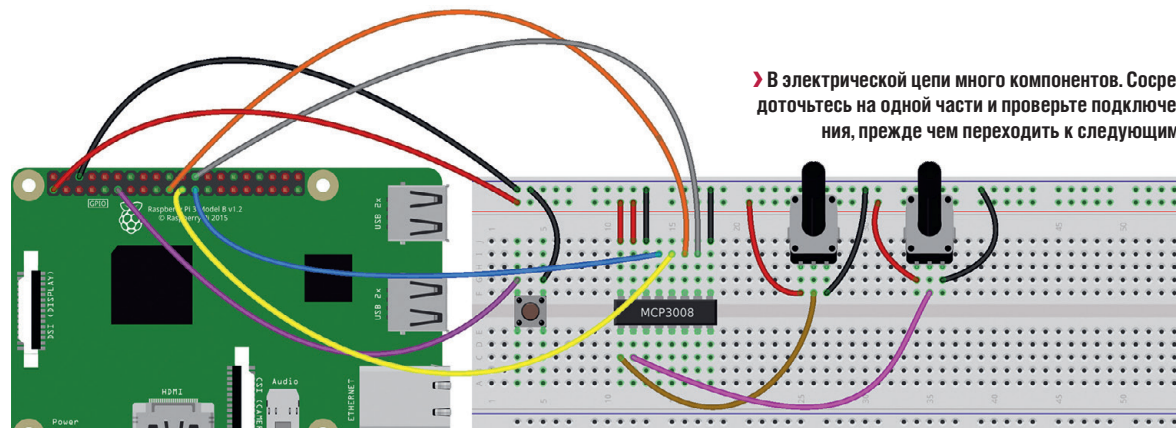
Теперь напишем код. Перейдите в меню Программирование и выберите Python 3. Когда откроется приложение, нажмите Файл > Создать, чтобы создать новый пустой файл. Сразу же сохраните файл (Файл > Сохранить) как **timelapse-controller.py**. Работу с кодом мы начнем с импорта библиотек, которые будем использовать. Это *picamera*, для управления камерой; *GPIO Zero* для нашего аналого-цифрового преобразователя и кнопки; из библиотеки *time* нам нужна функция *sleep*; и еще нам понадобится *datetime*, чтобы создать отметку времени.

```
from picamera import PiCamera
from gpiozero import MCP3008, Button
from time import sleep
from datetime import datetime
```

Теперь надо настроить соединения каналов с MCP3008, сообщить GPIO, где подключена наша кнопка, и создать объект из библиотеки *picamera*, чтобы с ним было проще работать.

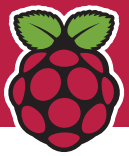
```
pot1 = MCP3008(channel=0)
pot2 = MCP3008(channel=1)
button = Button(17)
camera = PiCamera()
```

Для создания снимков мы введем функцию, которую позже сможем вызывать в нашем коде. Эта функция, называемая *capture*, будет начинаться с цикла *for*, повторяющийся столько раз, сколько укажет пользователь. Цикл *for* создает временную отметку,



» В электрической цепи много компонентов. Сосредоточьтесь на одной части и проверьте подключение, прежде чем переходить к следующим.





## Цифровой кинеограф

Недавно нам пришлось создавать видео из серии изображений, и первым препятствием, с которым мы столкнулись, было то, что отметки времени требовалось преобразовать в числовой порядок по возрастанию. К счастью, применив немного магии BASH, мы нашли однострочный сценарий оболочки, способный преобразовать метки в числа. Мы проделали это с копией всех изображений, чтобы иметь резервный вариант.

```
$ ls | cat -n | while read n f; do mv "$f" "${n}.jpg"; done
```

А зачем нам на изображениях отметка времени? Ну, этот проект можно использовать в научных

экспериментах, поэтому фиксация времени и даты каждого изображения позволяет увидеть, когда произошли изменения — например, при наблюдении, как постепенно подрастает кресс-салат или подсолнухи.

Само видео мы создали с помощью *avconv*. Он работает и на Raspberry Pi.

Чтобы установить *avconv*, откройте терминал и введите

```
$ sudo apt-get install libav-tools
```

Мы-то использовали его с ноутбуком Core i5, который намного лучше по производительности.

Теперь запустите команду в том же каталоге, где хранятся наши только что переименованные изображения. Она установит смену кадров с частотой 10 кадров в секунду, начиная с изображения под номером 1, битрейт 1000 К, и сохранит файл в *test.mp4*:

```
avconv -r 10 -start_number 1 -i %d.jpg -b:v 1000k test.mp4
```

Теперь вы можете открыть видео с помощью *omxplayer* и насладиться заслуженными плодами своих трудов:

```
omxplayer test.mp4
```

включающую время и дату снимка, а затем использует это значение как имя файла для захвата изображения. Затем *for* засыпает — опять же, на период времени, заданный пользователем — и потом выполняется снова.

```
def capture():
    for i in range(timer):
        timestamp = datetime.now().isoformat()
        camera.capture('/home/pi/%s.jpg' % timestamp)
        sleep(delay)
```

Чтобы увидеть то, что видит камера, мы создаем окно предпросмотра. В нашем случае это окно имеет фиксированное разрешение 640×480 и помещается в верхнем левом углу экрана (0,0). Но обычный предпросмотр является полноэкранным.

```
camera.start_preview(fullscreen=False, window = (0, 0, 640, 480))
```

### Попытка — не пытка

Используя конструкцию *try...except* и *finally* для обработки любых ошибок и для завершения кода пользователем, мы создаем две переменные: *timer* и *delay*. В переменной *timer* будет храниться время в минутах, в течение которого должна выполняться покадровая съемка. Переменная *delay* используется для хранения интервала в секундах между кадрами. Обе переменные фиксируют ввод пользователя с клавиатуры и преобразуют его в целочисленное значение.

```
try:
    timer = int(input("Сколько минут продолжается съемка?"))
    delay = int(input("Каков интервал, в секундах, между снимками?"))
```

Теперь внутри цикла *while true* мы берем значение потенциометра, обычно между 0.0 и 1.0, и умножаем его на 100, чтобы получить значение для управления яркостью и контрастностью изображения. Поворот потенциометров изменит это значение и выведет новое в оболочку.

```
while True:
    brightness = round(pot1.value * 100)
    print("Яркость", brightness)
    contrast = round(pot2.value * 100)
    print("Контраст", contrast)
```

Используйте переменные *brightness* и *contrast*, чтобы обновить окно предпросмотра и показать изменение конфигурации. Затем создадим переменную *settings*, куда войдут оба значения.

```
camera.brightness = brightness
camera.contrast = contrast
settings = "Яркость: "+str(brightness)+" Контраст: "+str(contrast)
```

Прежде чем появиться в окне предпросмотра, эти настройки отображаются в оболочке Python, как правило, в верхней части изображения. Перед повтором есть задержка в 0,1 секунды:

```
print(settings)
camera.annotate_text = settings
sleep(0.1)
```

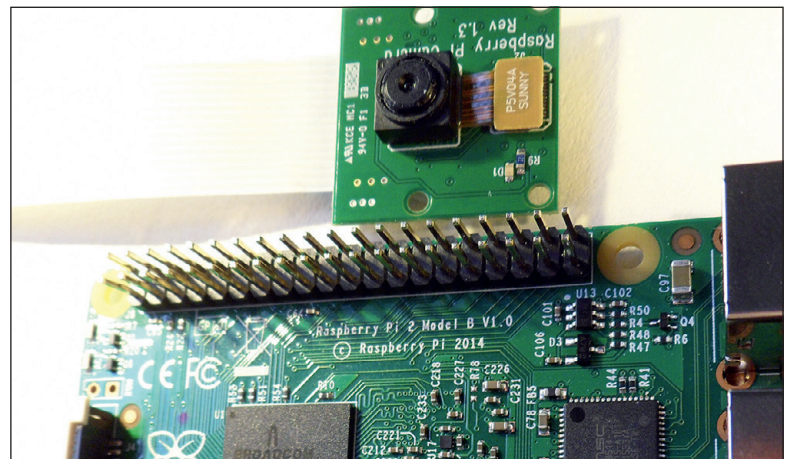
Теперь наша кнопка подготовлена к созданию серии снимков. При ее удержании в течение двух секунд она вызывает функцию *capture* (созданную ранее). Затем она делает снимки через тот интервал, который пользователь указал в переменных *timer* и *delay*.

```
button.when_held = capture
```

Переходим к заключительной части проекта. Здесь у нас есть часть *except*. С ее помощью пользователь сможет выйти из приложения, нажав Ctrl+C. Эта комбинация останавливает предпросмотр и затем переходит к части *finally*, где мы указываем, что приложение вышло в оболочку Python:

```
except KeyboardInterrupt:
    camera.stop_preview()
finally:
    print("ПРЕКРАШЕНИЕ СЪЕМКИ")
```

Сохраните код и нажмите Запустить > Запустить модуль. Откроется окно предпросмотра, и вы сможете настроить яркость и контрастность с помощью потенциометров, установить таймер и задержку; затем нажмите и удерживайте кнопку, чтобы начать съемку. Осталось только найти подходящий объект для создания серии фотографий. **LXF**

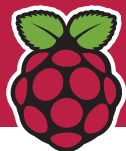


► Официальная камера Raspberry Pi создавалась специально для Pi и может снимать фото и видео с HD-разрешением 1080 пикселей.

### Скорая помощь

Официальная камера Raspberry Pi поставляется с до обидного коротким проводом. Как вариант, можно купить удлинитель от Tindie. Очень удобно!

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)



# Doom: Победим Силы Зла

Нейт Дрейк добавит шоколадную ноту вашему Raspberry Pi, установив на него версию культовой стрелялки от первого лица.



## Наш эксперт

**Нейт Дрейк** держится за свои честно заработанные охотой за сокровищами деньги и уклоняется от хитроумных вербовщиков. Сейчас он [Ред.: — По глупости] трудится исключительно внештатным журналистом по технологиям.

Несмотря на заезженный сюжет и неуклюжую графику, *Doom*, созданный id Software в 1993 г., снискал огромную популярность в мире ПК. Он помог жанру стрелялки от первого лица (FPS) укрепить свои позиции среди прочих видеоигр и породил ряд сиквелов, ответвлений и уровней, созданных фанатами.

Даже те, кто слишком юн, чтобы помнить появление *Doom*, скорее всего, знакомы с жанром FPS. Поскольку *Doom* является наиболее знаковым из них, нам хотелось, чтобы он ощущался как можно ближе к оригиналу из 1990-х годов, поэтому для запуска на нашем Raspberry Pi мы выбрали Chocolate Doom Engine.

Этот проект предполагает использование Raspberry Pi 2 или 3 с последней версией Raspbian. С целью убедиться, что это так, воспользуйтесь командой `sudo apt-get update`, а затем `sudo apt-get upgrade`. Если вы хотите играть в многопользовательский режим Deathmatch от *Doom*, вам также понадобится подключить Pi к вашей домашней сети и/или Интернету.

## Вся подноготная Doom

Жанр FPS предполагает, что вы видите всё глазами главного героя. Основной сюжет игры-оригинала в том, что вам надо исследовать луны Фобос и Деймос, а затем и самый Ад. Играя от лица космического путешественника, обладающего достойным выбором оружия, вы должны пробивать себе дорогу, сражаясь с демонами, слугами и силами тьмы [Ред.: — Привычная по тем временам публика], чтобы обрести хоть какой-то шанс на победу.

Демонические образы и чрезмерное насилие в игре пришлось не по душе ряду религиозных групп и политических активистов, которые после выхода игры безуспешно проводили кампании за то, чтобы подвергнуть *Doom* цензуре или сделать его более миролюбивым. В Германии игру запретили сразу, за «кровавое, садистское насилие», хотя в 2011 г. этот запрет был отменен.

Однако, похоже, нет ничего эффективнее черного пиара. В период с 1993 по 1999 г. было продано 1,1 миллионов копий *Doom*, что сделало его восьмой по популярности игрой того времени. К 1995 г. по всему миру насчитывалось около 10 миллионов компьютеров с установленной версией *Doom*.

В теории, игра проста — впрочем, как и на практике. На каждом уровне вы должны найти выход в следующий. Эти места обычно помечены знаком выхода. Однако прежде вам предстоит пройти через лабиринт уровней, одинаковых коридоров и запертых дверей. Доступна кое-какая помощь в форме бонусов, непонятно кем разбросанных по всей территории. К ним относятся основные карты, броня, оружие и аптечки первой помощи. Вы также можете использовать карту, нажав клавишу Tab, чтобы узнать, где вы уже были.

У многих монстров, населяющих игру, есть раздражающая привычка появляться тогда, когда их ждешь меньше всего. Они также охраняют ценные предметы и выходы, и, к несчастью, имеют привычку стрелять в вас или издали бросать огненные шары.



В начале игры вы выбираете из разных уровней с многообещающими названиями.

Хотя прислужников у Ада хватает, вражеский ИИ не особенно продвинутый, и они способны сдуру атаковать друг друга, если столкнутся в пути.

Еще одним преимуществом является арсенал внутриигрового оружия. Хотя вы начинаете с пистолета, по ходу игры у вас есть шанс подержать в руках дробовик, пулемет, плазменную винтовку и даже дьявольский дефрагментатор BFG (Big Fragging Gun) 9000.

## Двигатель на старт

Про движок *Doom* можно сказать, что он является исполняемым элементом игры, в отличие от файлов WAD (см. раздел *Где все данные?*, стр. 97), которые содержат необработанные данные, такие как графика, звук, оружие и т.д. Движок контролирует разные аспекты игры, такие как общая структура уровней и движения.

Официальный движок *Doom*, разработанный id, известен как Vanilla Doom, из-за его невыразительности. Так его называют энтузиасты *Doom*, чтобы разделять официальный игровой движок для запуска *Doom* и другие варианты. Различать их важно, поскольку движок *Doom* был модифицирован и повторно использован в ряде коммерческих игр, таких как *Heretic* и *Hexen*.

С тех пор как в 1997 г. id выпустил исходный код движка *Doom*, разработчики ряда сообществ также адаптировали его в своих целях. Chocolate Doom — один из таких проектов.

Хотя Chocolate Doom и затеял переделку игрового движка, он максимально воспроизводит опыт игры в оригинальной версии *Doom*. Любые изменения в исходном игровом движке не мешают этой аутентичности. Что еще важнее, Chocolate Doom стремится к совместимости всеми файлами расширения *Doom* (WAD), которые были разработаны для работы с Vanilla Doom.

Единственное, в чем Chocolate Doom значительно отличается от оригинала — это игра в многопользовательском режиме



## Скорая помощь

В мае 2016, id Software выпустили обновленный *Doom* для Microsoft Windows, PlayStation 4 и Xbox. Будем надеяться, что придет черед и для версии Pi.



## Жульничать нехорошо

В Chocolate Doom всё как в оригинале, в том числе встроенные коды-обманки. Они могут использоваться в любое время во время игры, просто при наборе их на клавиатуре. В большинстве случаев вы увидите сообщение, подтверждающее, что чит [cheat — *англ.* обман] был активирован. Чит-коды ниже предполагают, что вы используете QWERTY-клавиатуру. Коды могут работать в многопользовательских Deathmatches, в зависимости от настроек, предусмотренных создателем игры. Добавьте флаг `--nocheats`, чтобы другие не имели несправедливого преимущества.

Вот несколько читов, которые можно использовать по отдельности:

» `idchoppers` Этот чит вооружит вас бензопилой. Нажмите 1, чтобы выбрать его.

» `idcleve##` Используйте этот чит-код для перехода в другой эпизод или карту по вашему выбору — так, например, чит `idcleve02` переправляет вас на карту номер 2.

» `iddqd` С этим читом у вас всегда будет 100-процентное здоровье. Этот «режим Бога» сделает вас неуязвимым практически от любой напасти, хотя некоторые типы полов всё же смогут нанести вам вред. Вас также может убить телефрагментацией — если другие игроки или существа телепортируются туда, где вы находитесь.

Есть еще куча подобных строк для ленивого, отчаянного или, возможно, менее квалифицированного игрока. Полный список читов можно посмотреть в Doom wiki по адресу [http://doom.wikia.com/wiki/Doom\\_cheat\\_codes](http://doom.wikia.com/wiki/Doom_cheat_codes).



» Включив режим Бога, вы можете без проблем вести ближний бой, орудуя бензопилой.

Deathmatch. Здесь одна из машин должна выполнять функцию сервера, в отличие от Vanilla Doom, где игроки просто подключаются напрямую друг к другу. К счастью, это очень легко настроить (см. *Настройка Doom Deathmatches*, стр. 98).

## Где все данные?

Чтобы играть в *Doom*, вам понадобятся движок и файлы WAD (Where's All Data). Файлы WAD содержат данные игры: наполнение уровней, графики, звуковых эффектов и фоновую музыку. Идея WAD состоит в том, чтобы облегчить людям создание пользовательских уровней и модификаций игры. WAD существуют отдельно от игровых движков вроде Chocolate Doom и загружаются из других источников. Существует два основных типа WAD. В этом проекте мы сосредоточимся на iWAD (сокращение от internal, т.е. внутренних WAD), которые содержат данные для всех уровней игры.

Когда *Doom* только вышел, энтузиасты тут же принялись создавать новую графику, звуковые эффекты и т.д., но в отношении фундаментальных аспектов игры они многого сделать не могли. Например, степени урона, наносимого монстрами. После публикации исходного кода *Doom* и благодаря значительным усилиям сообщества в целом стало можно менять все аспекты игры, и есть тысячи WADов. Одни представляют собой просто небольшие вариации игрового процесса, тогда как другие значительно расширяют возможности исходной игры. Одним из самых известных подобных «MegaWAD» является *Eternal Doom*, который содержит 32 полных уровня, каждый из которых примерно вчетверо больше, чем уровни исходного Vanilla Doom.

Каким бы впечатляющим он ни был, *Eternal* — лишь расширение Вселенной *Doom*. Другие WAD считаются «полными конверсиями» — в том смысле, что ресурсы там совсем иные, чем в исходной игре. Первой из таковых была *Aliens TC* Джастина Фишера [Justin Fisher], основанная на франшизе фильма.

Тем, кто интересуется работой в данной отрасли, не стоит воспринимать проектирование игровых уровней как пустую трату времени — Фишеру предлагали работу DreamWorks, от которой он отказался ради того, чтобы завершить образование в университете. Однако другие дизайнеры популярных WAD, например, Йикка Керянен [Iikka Keränen], впоследствии сотрудничали с крупными производителями видеоигр в работе над официальными проектами.

По соображениям авторского права, в этом проекте мы сосредоточимся на официальной свободно распространяемой версии

WAD *Doom* (**DOOM1.WAD**), загруженной с фанатского сайта [www.doomworld.com](http://www.doomworld.com). Она содержит только первый эпизод игры, *Knee Deep in the Dead* [По колени в мертвецах], и в свое время была выпущена, чтобы пробудить интерес к игре. После прохождения первого уровня игрокам предлагалось заказать полную версию.

При этом надо сказать, что кроме нее для скачивания доступно несколько других официальных и неофициальных WAD, так что если в вашей стране это законно, загружайте и экспериментируйте на здоровье.

## Freedoom — не бесплатен

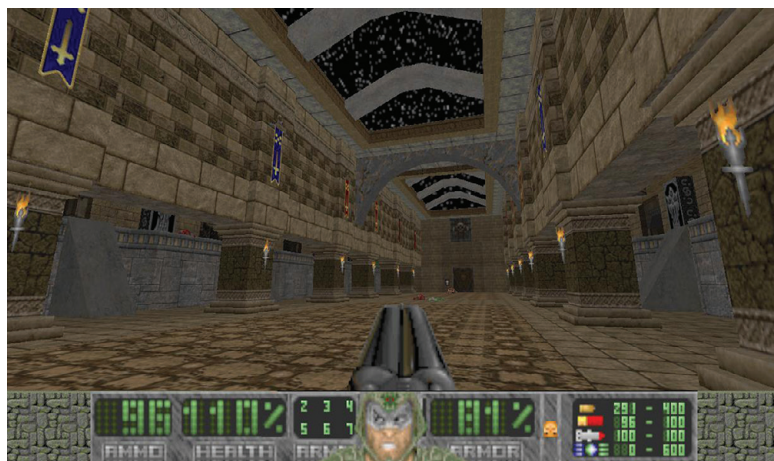
Freedoom Project — отличное подтверждение неугасающей популярности *Doom*. Заявленной целью проекта является создание WADов в стиле *Doom* целиком из свободного контента. Поскольку id Software опубликовала исходный код *Doom*, базовый код игры является бесплатным. Freedoom надстраивает на нем ряд уровней со своей графикой, звуковыми эффектами и музыкой.

В результате получилась бесплатная игра с открытым исходным кодом, без какого-либо коммерческого контента.

Freedoom предоставляет только файлы WAD, поэтому для игры требуется также совместимый игровой движок. На сайте Chocolate Doom сказано, что однопользовательские WAD-файлы Freedoom

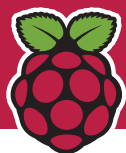
## Скорая помощь

Начинающие кодеры могут посмотреть исходный код оригинала *Doom* на <https://github.com/id-Software/DOOM>.



» *Eternal Doom* от TeamTNT славится богатой графикой, великолепной архитектурой и насыщенными уровнями.

» Подпишитесь на печатную или электронную версии на [www.linuxformat.ru/subscribe!](http://www.linuxformat.ru/subscribe!)



## Скорая помощь

Полный архив WADов Doomworld находится на [www.doomworld.com/](http://www.doomworld.com/) [idgames.com/](http://idgames.com/). Посмотреть самые популярные в сообществе Doom WAD можно на [www.doomworld.com/20years/best1.php](http://www.doomworld.com/20years/best1.php).

с ним несовместимы. Тем не менее, он рекомендует скачивать и использовать набор уровней Deathmatch от Freedoom, известных как "FreeDM". В отличие от обычных уровней Deathmatch, которые вы найдете в Vanilla и Chocolate Doom, здесь основное внимание уделяется игрокам — монстрам нет.

Если игра *Doom* уже раздражила ваш аппетит на предмет попробовать свои силы на уровне дизайнера, на сайте Freedoom также есть сотни оригинальных текстур и звуковых эффектов, которые можно использовать бесплатно. Посетите <https://freedoom.github.io>, чтобы найти их и узнать больше.

## Ваш момент «эврики»

Как человек, знающий толк в Raspberry Pi, вы, вероятно, не желаете играть с WAD, созданными другими, но захотите редактировать и создавать собственные уровни. *Doom* изначально был предназначен, чтобы его вскрывать, расширять и модифицировать — таким образом, ваше творчество полностью соответствует духу игры.

В Raspbian есть *Eureka*, единственный редактор уровней *Doom* в Linux. Перейдите в Приложения > Установки > Установка и удаление программ и найдите его, используя панель поиска в левом верхнем углу, или посетите сайт [www.sourceforge.net/p/eureka-editor](http://www.sourceforge.net/p/eureka-editor) для загрузки этого ПО.

С помощью *Eureka* можно редактировать или создавать WAD-файлы, чтобы писать новые уровни с нуля или изменять макет или объекты в существующих. Получить помощь по работе с редактором можно на форумах [Doomworld.com](http://Doomworld.com).

## Doom в деталях

Ради простоты этот проект построен на свободно распространяемом **DOOM1.WAD**, который загружается в каталог **chocolate-doom**. Если ваш WAD-файл располагается где-то в другом месте, то при запуске Chocolate Doom просто укажите путь к нему, например:

```
chocolate-doom -WAD /home/pi/Downloads/DOOM1.WAD -window 640x480
```

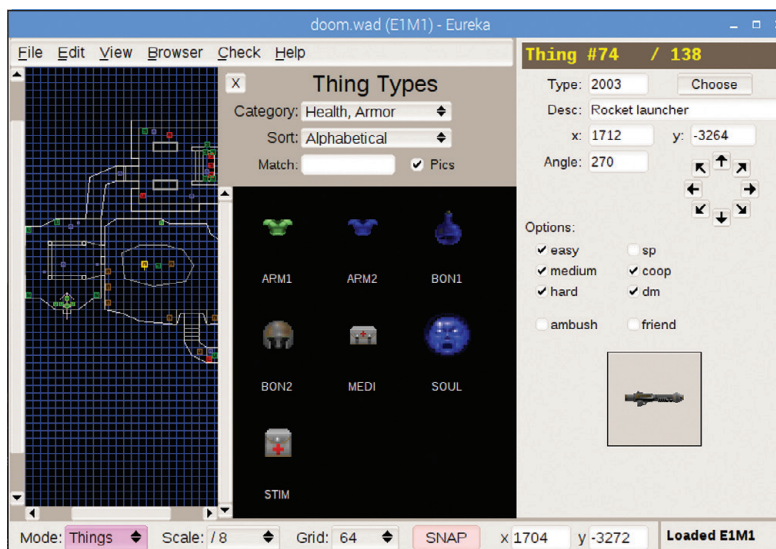
Chocolate Doom довольно педантичен по части имен файлов (заглавные буквы имеют значение), и если на экране установки вам нужно указать файл WAD, возможно, стоит переименовать ваши файлы в **.wad**, а не **.WAD**. То же касается редактора уровней *Eureka* — в ходе теста программа не смогла распознать файл **DOOM1.WAD**, пока он не был переименован в **doom.wad**.

Полный список всех аргументов командной строки и дальнейшая помощь с установкой доступны в руководстве пользователя Chocolate Doom, оно находится на сайте [www.chocolate-doom.org/wiki/index.php/User\\_guide](http://www.chocolate-doom.org/wiki/index.php/User_guide).

Чтобы играть в Deathmatches в *Doom*, вам надо открыть в своем брандмауэре и/или маршрутизаторе UDP-порт 2342. Что именно для этого делать, будет зависеть от вашего маршрутизатора. Обратитесь к производителю или посетите сайт [www.portforward.com](http://www.portforward.com), там есть соответствующая информация.

К сожалению, Chocolate Doom не совместим с режимом одного игрока WAD Freedoom. Возможно, вы обнаружите, что другие файлы WAD работают неправильно или создают проблемы для вашего Pi. Но в идеале ваш Pi должен быть отведен только для игр, поэтому риска потери данных быть не должно.

Примите нашу самую горячую благодарность за ваши усилия по борьбе с силами Ада. Удачи, солдат! **LXF**



➤ Чтобы переместить отдельные элементы в редакторе уровней *Eureka*, просто щелкните по ним. Изменить их свойства можно в панели справа.

## Настройка Doom Deathmatches

Chocolate Doom без особого труда поддерживает многопользовательский режим. В отличие от Vanilla Doom, в Chocolate Doom можно играть через Интернет или через LAN (локальная сеть). Однако для простоты, прежде чем начать, убедитесь, что все игроки, которые хотят принять участие, запускают Chocolate Doom и используют тот же файл WAD.

Независимо от того, играете ли вы через Интернет или через локальную сеть, один из участвующих игроков должен выступать в роли сервера и раздавать игру. Будьте уверены, что игрок с хоста тоже сможет в ней участвовать. UDP-порт 2342 должен быть открыт на каждом маршрутизаторе и / или межсетевом экране каждого игрока. (См. «*Doom в деталях*», вверху.)

Чтобы разместить игру Deathmatch на своем Pi, откройте терминал или подключитесь через SSH и выполните команду

```
chocolate-doom-server-privateserver-deathmatch
```

Чтобы присоединиться к вашей игре, интернет-игроки должны скопировать

```
chocolate-doom -connect xxx.xxx.xxx.xxx
```

Замените xxx.xxx.xxx.xxx общедоступным IP-адресом машины хоста. Свой публичный IP-адрес вы можете узнать, посетив такие сайты, как [www.whatismyip.com](http://www.whatismyip.com).

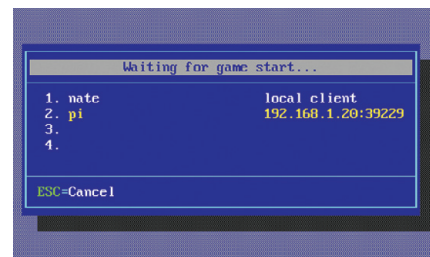
Для игроков, подключенных к локальной сети, всё еще проще: просто запустите игру обычным способом, но добавьте аргумент `-autojoin`, например:

```
chocolate-doom -WAD DOOM1.WAD -window 640x480 -autojoin
```

Появится окно, в котором будут показаны подключенные клиенты.

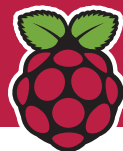
Чтобы начать игру, просто нажмите клавишу пробела.

Если вы хотите задействовать многопользовательский режим по максимуму, ознакомьтесь с дополнительной информацией в вики Wiki Chocolate Doom. Просто заходите на [www.chocolate-doom.org/wiki/index.php/Multiplayer](http://www.chocolate-doom.org/wiki/index.php/Multiplayer) и вникайте.



➤ Когда все игроки присоединились, нажмите пробел, чтобы начать игру.

➤ Пропустили номер? Узнайте на с. 108, как получить его прямо сейчас.



## Рецепт Chocolate Doom

```
fish /home/pi
File Edit Tabs Help
pi@raspberrypi -> sudo apt-get install libSDL-mixer1.2-dev libSDL-net1.2-dev
```

### 1 Установите зависимости

Для компиляции Chocolate Doom вашему Pi потребуются определенные программы. Откройте терминал из Приложения > Дополнения > Terminal или подключитесь к Pi через SSH и установите их командой

```
sudo apt-get install libSDL-mixer1.2-dev libSDL-net1.2-dev pythonimaging
```

Нажмите Y, чтобы при запросе подтвердить, что вы хотите продолжить установку.

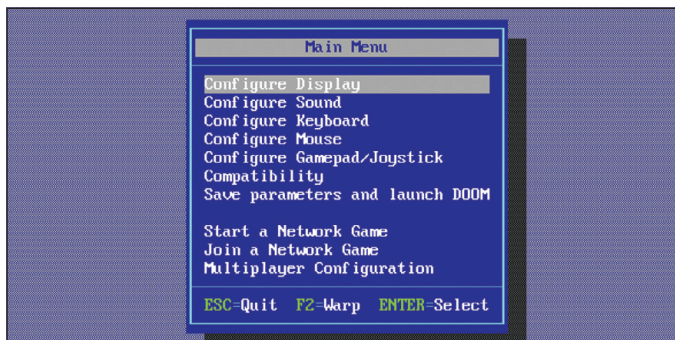
```
fish /home/pi/chocolate-doom-2.2.1
File Edit Tabs Help
6
../man/docgen -g hexen -m ../man/default.cfg.template \
-c default ../src/m_config.c > hexen.cfg.5
../man/docgen -g hexen -m ../man/extra.cfg.template \
-c extended ../src/m_config.c > chocolate-hexen.cfg.5
../man/docgen -g strife -m ../man/default.cfg.template ../src ../src/strife > chocolate-strife.6
../man/docgen -g strife -m ../man/default.cfg.template \
-c default ../src/m_config.c > strife.cfg.5
../man/docgen -g strife -m ../man/extra.cfg.template \
-c extended ../src/m_config.c > chocolate-strife.cfg.5
../man/simplecpp -DDOOM -DPRECOMPILED < ../man/INSTALL.template > INSTALL.doom
../man/docgen -p ../man/CMDLINE.template ../src/ ../src/doom/ > CMDLINE.doom
../man/simplecpp -DHERETIC -DPRECOMPILED < ../man/INSTALL.template > INSTALL.heretic
../man/docgen -p ../man/CMDLINE.template ../src/ ../src/heretic/ > CMDLINE.heretic
../man/simplecpp -DHEXEN -DPRECOMPILED < ../man/INSTALL.template > INSTALL.hexen
../man/docgen -p ../man/CMDLINE.template ../src/ ../src/hexen/ > CMDLINE.hexen
../man/simplecpp -DSTRIFE -DPRECOMPILED < ../man/INSTALL.template > INSTALL.strife
../man/docgen -p ../man/CMDLINE.template ../src/ ../src/strife/ > CMDLINE.strife
make[2]: Leaving directory '/home/pi/chocolate-doom-2.2.1/man'
```

### 3 Скомпилируйте Chocolate Doom

Используйте команду `cd` для перехода во вновь созданный каталог:

```
cd chocolate-doom-2.2.1
```

Затем выполните `configure`, чтобы подготовить необходимые файлы, и `make`, чтобы начать компиляцию. Это требует некоторого времени, так что можете спокойно сделать перерыв. По завершении выполните следующую команду, чтобы установить саму программу: `sudo make install`.



### 5 Запуск процедуры настройки Doom

Используйте следующую команду для запуска установки:

```
chocolate-doom-setup -WAD DOOM1.wad
```

Этим инструментом можно настроить изображение, звук, раскладку клавиатуры, мышь и даже геймпад, если он у вас есть. Технически можно использовать этот экран для размещения и соединения многопользовательских deathmatches, но проще всего сделать это из командной строки. См. врезку внизу стр. 98.

```
fish /home/pi
File Edit Tabs Help
pi@raspberrypi ->
pi@raspberrypi -> wget https://www.chocolate-doom.org/downloads/2.2.1/chocolate-doom-2.2.1.tar.gz
--2016-12-17 13:53:52-- https://www.chocolate-doom.org/downloads/2.2.1/chocolate-doom-2.2.1.tar.gz
Resolving www.chocolate-doom.org (www.chocolate-doom.org)... 178.79.130.180, 2a01:7e00::f03c:91ff:fe50:c56
Connecting to www.chocolate-doom.org (www.chocolate-doom.org)|178.79.130.180|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2127538 (2.0M) [application/gzip]
Saving to: 'chocolate-doom-2.2.1.tar.gz.1'

chocolate-doom-2.2.1. 100%[=====] 2.03M 843KB/s in 2.5s
2016-12-17 13:53:54 (843 KB/s) - 'chocolate-doom-2.2.1.tar.gz.1' saved [2127538/2127538]

pi@raspberrypi -> tar xzf chocolate-doom-2.2.1.tar.gz
pi@raspberrypi ->
```

### 2 Скачайте и распакуйте

Перейдите на сайт [www.chocolate-doom.org/wiki/index.php/Downloads](http://www.chocolate-doom.org/wiki/index.php/Downloads) и введите `wget` в терминале Pi, чтобы загрузить игру. На момент написания, это делалось так:

```
wget https://www.chocolate-doom.org/downloads/2.2.1/chocolatedoom-2.2.1.tar.gz2.2.1.tar.gz
```

Когда файл загрузится, извлеките его с помощью `tar` — например, `tar xzf chocolate-doom-2.2.1.tar.gz`.

```
fish /home/pi/chocolate-doom-2.2.1
File Edit Tabs Help
pi@raspberrypi ~/chocolate-doom-2.2.1>
pi@raspberrypi ~/chocolate-doom-2.2.1> wget http://www.doomworld.com/3downloads/ports/shareware_doom_iwad.zip
--2016-12-17 14:12:09-- http://www.doomworld.com/3downloads/ports/shareware_doom_iwad.zip
Resolving www.doomworld.com (www.doomworld.com)... 69.164.207.121, 2600:3c00:f03c:91ff:fe26:a249
Connecting to www.doomworld.com (www.doomworld.com)|69.164.207.121|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1782482 (1.7M) [application/zip]
Saving to: 'shareware_doom_iwad.zip'

shareware_doom_iwad.z 100%[=====] 1.70M 515KB/s in 3.4s
2016-12-17 14:12:13 (515 KB/s) - 'shareware_doom_iwad.zip' saved [1782482/1782482]

pi@raspberrypi ~/chocolate-doom-2.2.1> unzip shareware_doom_iwad.zip
Archive: shareware_doom_iwad.zip
  inflating: DOOM1.WAD
pi@raspberrypi ~/chocolate-doom-2.2.1>
```

### 4 Загрузите файл (или файлы) WAD

Загрузите условно-бесплатную версию *Doom*, используя `wget`. Убедитесь, что вы по-прежнему находитесь в папке Chocolate Doom, и выполните следующие действия:

```
wget http://www.doomworld.com/3downloads/ports/shareware_doom_WAD.zip
```

Когда файл загрузится, извлеките его, выполнив следующее:

```
unzip shareware_doom_WAD.zip
```

Именем полученного файла будет `DOOM1.WAD`.



### 6 Запуск Chocolate Doom

Чтобы запустить *Doom*, выполните следующую команду:

```
chocolate-doom -WAD DOOM1.WAD
```

*Doom* запустится в полноэкранном режиме, что для Pi может быть напряженно, поэтому лучше открыть его в окне с помощью такой команды:

```
chocolate-doom -WAD DOOM1.WAD -window 640x480
```

Если у вас есть другой файл WAD, подставьте его имя вместо `DOOM1.WAD`.



# LXF Hot Picks



**Александр Толстой**

считает, что использовать Linux с приложениями с закрытым кодом так же ненатурально, как смешивать масло с водой.

Flowblade » Kmail » Screencloud » ANGRYsearch » tVitunes » Min » Curlew » Konv » Feedreader » SpeedLazer » Extreme Tux Racer

## Видеоредактор

# Flowblade

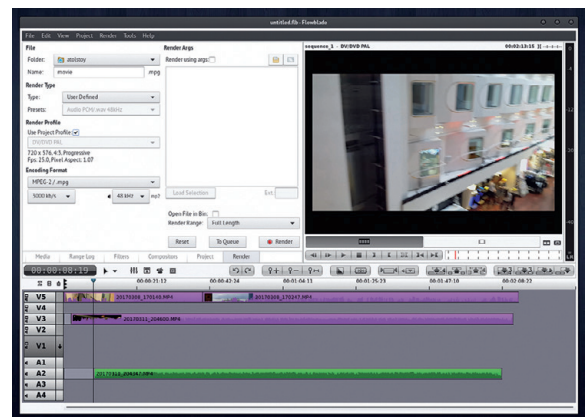
Версия: 1.12.2 Сайт: <http://bit.ly/2rM2Ui3>

Даже в эру после видеокamer люди продолжают создавать бесчисленные гигабайты видеоконтента. Большинство этого контента идет со смартфонов и по большей части остается неотредактированным и нелюбимым. Команда *Linux Format* полагает, что эти файлы не должны быть забыты, став мертвым грузом на жестком диске, поэтому давайте найдем для них лучшее применение и создадим крутое домашнее видео.

В сопоставлении с самыми солидными нелинейными видеоредакторами в Linux, такими как *Kdenlive* и *OpenShot*, *Flowblade* будет менее прославленным соперником. Но пусть его безвестность вас не пугает.

Это программа-видеоредактор, с которым вам захочется познакомиться, и, вероятно, вы и не подозревали о его существовании до того момента, когда взялись за эту страницу (если только вы не читали о нем в LXF175).

Основными преимуществами *Flowblade* являются дружелюбная и логичная структура GUI вкупе с общей хорошей производительностью. Нижняя часть окна занята



» *Flowblade* демонстрирует хороший баланс между функциями и простотой редактирования видеоклипов.

## Это видеоредактор, с которым вам захочется познакомиться.

временной шкалой, а зона предпросмотра с кнопками управления находится в правом верхнем углу. Если вы когда-либо редактировали видео в любом другом видеоредакторе, ознакомление с *Flowblade* не займет у вас много времени благодаря его традиционной структуре интерфейса.

Вы можете начать с импорта исходного клипа, нажав на кнопку **Add** или просто перетаскивая файлы в верхнюю левую часть окна. Затем можете перетаскивать клипы на временную шкалу и нарезать их с разными эффектами и переходами. На временной шкале имеется несколько треков (по умолчанию активный трек установлен на **Track 1**, самый нижний), что позволяет разместить несколько клипов и поиграть с наложением эффектов. В верхней левой части имеются вкладки внизу для переключения на разные виды, исследования фильтров и композиторов, управления проектами и рендеринга вашей работы в целевой файл.

*Flowblade* использует тот же самый движок **MLT/FFmpeg**, что и большинство других нелинейных редакторов. Приложение требует установки длинного списка модулей **Rythm**, но ему не нужно никакой компиляции из исходника. Выполнив все его требования, просто запустите скрипт `./flowblade` для начала работы. Если вы ничего не пропустили в своей системе, *Flowblade* должен оказаться стабильным и полностью функциональным.

## Исследуем интерфейс Flowblade

### Микшер, блок ввода титров и другое

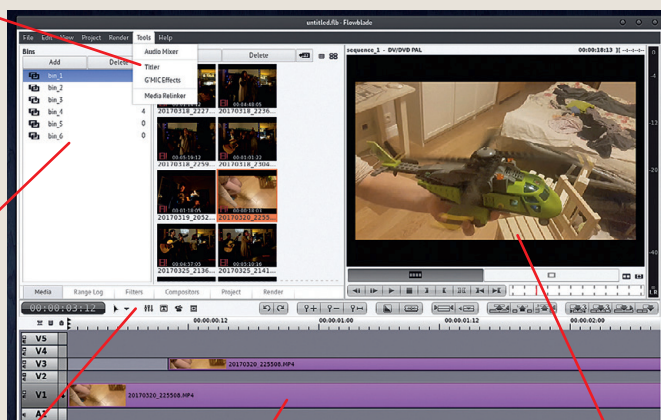
Отдельные инструменты для добавления титров, мастеринга саундтреков и дополнительных эффектов.

### Клипы и контейнеры

Работайте со своими клипами внутри одного или более контейнеров. Удобно при работе с большим количеством входящих клипов.

### Фильтры и композиторы

*Flowblade* имеет ряд дополнительных функций, скрытых за похожими на вкладки кнопками.



### Временная шкала

У вас есть пять видеотреков и четыре аудиотрека, которые можно заблокировать или убрать звук в любое время.

### Панель предпросмотра

Вы можете управлять происходящим с помощью удобной области предпросмотра с управлением воспроизведением.

## Клиент электронной почты

## KMail

Версия: 17.04 Сайт: <https://github.com/KDE/kmail>

Способ чтения сообщений электронной почты в последнее время изменился, и не надо быть гением, чтобы понять тенденцию. Все больше людей привыкает к получению доступа к входящим сообщениям через браузеры, а не клиенты рабочего стола. Однако традиционные почтовые приложения вроде *Thunderbird* и *Evolution* по-прежнему имеют большую пользовательскую базу.

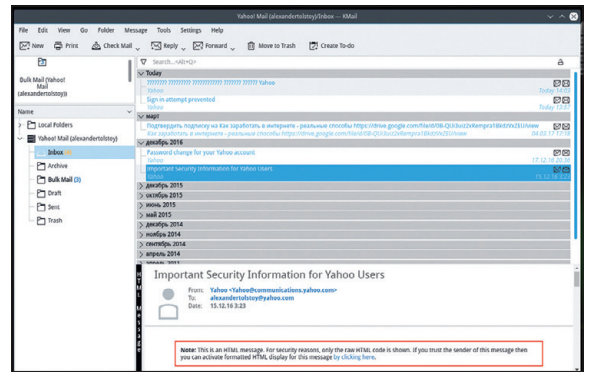
Более того, имеется хороший выбор подобных почтовых приложений с не столь известными именами, находящиеся в активной разработке. В **LXF214** мы исследовали *Trojita*, основанный на *Qt5* клиент IMAP, а на сей раз рассмотрим почтовое решение KDE по умолчанию. Неудивительно, что называется оно *KMail*.

Недавние обновления пакета KDE Applications, к которому и принадлежит *KMail*, сделали приложение еще лучше. Среди улучшений — переход на более быстрый движок *QtWebEngine* для рендеринга HTML-почты и возможность обратиться к Google

Safe Browsing и проверить, не является ли вредоносной ссылка, на которую вы собираетесь нажать.

Мы устроили *KMail* тест с нашей взятой для примера учетной записью Gmail, потому что множество пользователей уже подсели на Google — или, по крайней мере, имеют там хоть одну учетную запись. Важно отметить, что присоединить входящий ящик Google IMAP к настольному приложению с офлайн-хранилищем сложнее, чем кажется. Причина в том, что Google поощряет свой web-интерфейс и не поощряет сторонние приложения. Например, многие современные почтовые приложения, такие как *Trojita* или *Geary*, имеют проблемы или с офлайн-хранением, или с сохранением настроек, или с тем, и с другим.

**KMail зарекомендовал себя как солидное и удобное ПО.**



► Понятный интерфейс *KMail* поощряет вас и ваших друзей к развитию эпистолярного жанра.

Впрочем, *KMail* зарекомендовал себя как солидное и исключительно удобное в работе ПО. Поверх стабильной и не вызывающей проблем поддержки IMAP *KMail* предлагает достойный выбор полезных вспомогательных функций, таких как фильтры, средства обеспечения безопасности исходящих сообщений, интеграция сторонних инструментов анти-спам, соединитель базы данных Mozilla для плавной регистрации учетной записи и многое другое.

Итак, если вам требуется многофункциональное почтовое приложение со множеством дополнительных функций, способное работать с вашей учетной записью без указания серверов и портов, предоставьте *KMail* шанс на своем рабочем столе.

## Инструмент создания скриншотов

## Screencloud

Версия: 1.3.1 Сайт: <http://bit.ly/2swwLaP>

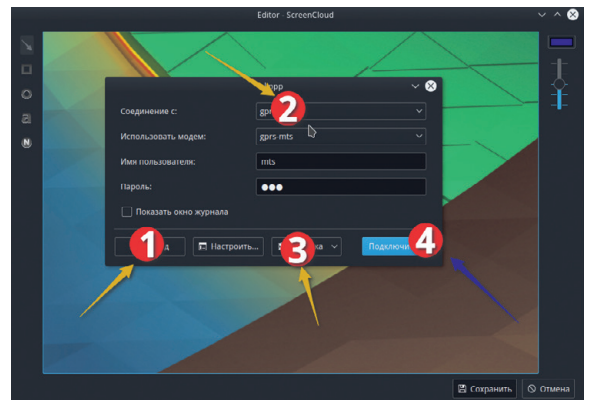
Не так уж давно (в **LXF223**) мы восхищались супер-удобным *Ksnip* и упоминали, что он взял лучшие функции Snipping Tools из Windows, добавив поверх них дополнительные функции. Однако *Ksnip* всё же не был идеальным решением, и мы продолжили поиски более богатого функциями приложения для создания экранных снимков (скриншотов).

Если вы работаете с сохраненными изображениями, которые надо аннотировать, хороший инструмент для создания скриншотов трудно переоценить. Он должен быть в состоянии делать больше, чем просто снимок окна или некой его части. Скажем, загружать изображения на специализированные web-сервисы (чтобы делиться всякими прелестями рабочего стола), рисовать стрелки и круги, нумеровать объекты, выделять элементы экрана и выполнять прочие задачи.

*Screencloud* — кросс-платформенный клиент web-сервиса [Screencloud.net](http://Screencloud.net).

Он выглядит, как коммерческое предприятие, но таковым отнюдь не является. Сайт вместе с клиентом рабочего стола — плод труда одного человека и проект с открытым кодом. Приложение позволяет делать снимок полного экрана, окна или выбранной области. Но основное преимущество *Screencloud* — его способности пост-обработки снимков во встроенном редакторе. Можно добавлять ровные стрелки, текстовые пометки, овалы или прямоугольники, и есть — что нам понравилось больше всего — автоматическая нумерация вставляемых кружочков. Здесь только пять инструментов рисования, но они очень кстати, потому что нет другого инструмента на *Qt5*, способного делать нечто подобное. Опции

**Есть автоматическая нумерация вставляемых кружочков.**



► Выделите ключевые элементы вашего изображения с помощью многообещающего инструмента с открытым кодом.

экспорта, однако, ограничены. *Screencloud* может отправить изображение в официальное web-хранилище или скопировать в буфер. Еще программа умеет делать загрузку на Dropbox, Imgur, индивидуальный (S)FTP-сервер или в локальный файл: просто нажмите на кнопку *More services* [Еще сервисы] в окне предпочтений приложения и отметьте требуемые окна. А на Github десятки плагинов для *Screencloud*.

*Screencloud* создан на Python и Qt, но оказался капризным в компиляции и установке, так как допускает использование Python 2 и 3 вместе с Qt4 или 5 в любой комбинации. Но пользователи популярных версий Linux мигром заставят его работать.

Поисковая машина

# ANGRYsearch

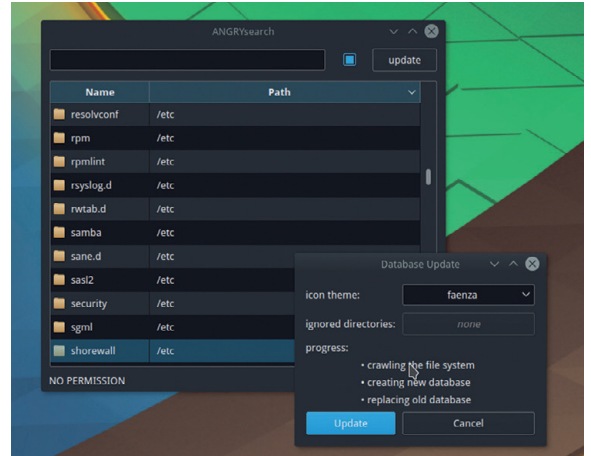
Версия: 1.0.0 Сайт: <http://bit.ly/2riPed7>

Было уже довольно поздно читать журналы, и солнце уже скрылось за горизонтом теплым летним вечером, когда мы наткнулись на древний выпуск LXF. И там была статья о *Beagle*, новаторском индексаторе файлов из страны корпоративного дистрибутива Linux от Novell, с массой крутых инструментов и функций. *Beagle* черпал вдохновение в поисковой машине Spotlight от Mac OS X. Удивительно, но даже спустя все эти годы более продвинутые индексаторы типа *Baloo* и *Tracker* по-прежнему не полностью отвечают запросам многих пользователей Linux из-за своей низкой скорости и заметному замедлению отклика системы.

*ANGRYsearch* — это отдельный индексатор и поисковая машина, очень похожая на *FSearch*. Обе они являются эквивалентами с открытым кодом для *Everything Search Engine* от Voidtools, исключительно для Windows. В будущем нам бы хотелось сравнить *FSearch* с *ANGRYsearch*, но раз уж

на тестовой машине есть файлы разработки PythonQt после компиляции *Screencloud*, естественно начать с *ANGRYsearch*, которая в свою основу кладет ту же технологию. Приложение появляется как чистое окно с панелью поиска и кнопкой Update [Обновить] рядом с ней. Не ждите здесь опций настройки помимо возможности выбрать тему значков и добавить определенные директории в список игнорирования.

*ANGRYsearch* индексирует всё, что есть в /, и помещает это в базу данных SQLite. На полный прогон для среднего дистрибутива Linux ушло относительно мало времени. Авторы похваляются, что их приложение справляется с миллионом файлов минуты за две, что после наших тестов



► Минимальный набор опций вкупе с очень быстрым поиском.

показалось нам весьма правдоподобным. Конечно, такая скорость возможна только если индексатор проходит только по именам файлов, а не их содержанию; и тем не менее, очень впечатляет, когда ваш файл обнаруживается, едва вы введете его имя.

После *ANGRYsearch* большинство других поисковиков выглядят тормозами. Отсутствие встроенных опций настройки компенсируется удобными подсказками, записанными автором *ANGRYsearch* в официальный документ README. Здесь вас научат автоматизировать обновление базы данных через *Cron*, и много чему еще.

Индексирует всё, что есть в /, и помещает в базу данных SQLite.

Музыкальный плеер

# Vitunes

Версия: 2.3 Сайт: <http://vitunes.org>

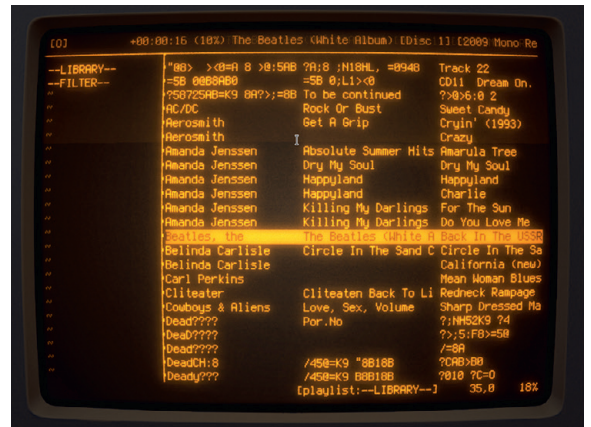
Мы всегда норовим рассказы-вать о музыкальных плеерах, имеющих новаторскую структуру GUI, но на этот раз выбрали инструмент, способный дополнить рабочее пространство командных строк технаря. Есть много программ для Linux, которые используют популярную библиотеку *Ncurses* — ту, которая рисует псевдографические интерфейсы в терминалах — а теперь мы наткнулись и на похожий музыкальный плеер.

Название *Vitunes* намекает на музыкальный магазин Apple и легендарный текстовый редактор *Vi*, и при ближайшем рассмотрении у него найдется нечто общее с обоими. *Vitunes* предлагает минималистичный интерфейс командной строки для вашей музыкальной коллекции и позволяет управлять вашей музыкальной библиотекой и наслаждаться своими треками. Приложение зависит от *Taglib*, и для воспроизведения звука ему нужен *Mplayer* или *Gstreamer*. Процесс компиляции прост:

```
$ mv Makefile.linux Makefile && make
```

Возможно, вы захотите сразу запустить плеер как `$. /vitunes`, но это не сработает. Вместо этого вам надо сначала создать базу данных (`$. /vitunes -e init`) и добавить как минимум одну директорию с музыкой (`$. /vitunes -e add ~/Music`). После этого вы готовы запустить минимальную версию исполняемого файла.

Навигация по разделам плеера осуществляется посредством Tab и клавиш со стрелками. Путешествие по *Vitunes* показало, что у плеера есть расширенные функции, например, фильтры (при поиске определенного трека), поддержка плейлиста редактором наподобие *Vi*, а также и ожидаемая глубокая настраиваемость. Порезвитесь с файлом `~/vitunes/vitunes.conf` и создайте уникальную, а, возможно,



► Вот еще одна причина отказаться от *X11* и *Wayland* в пользу интерфейса командной строки.

и цветную настройку *Vitunes*. Загляните на страницу официальной документации сайта *Vitunes*, чтобы найти индивидуальные шаблоны с приятными цветовыми схемами.

Нам очень понравился *Vitunes* благодаря его разумному и четкому подходу, который мягко заставляет вас консолидировать свою музыкальную библиотеку (если она в некотором беспорядке) и наслаждаться читаемыми метаданными для альбомов и треков, пока плеер с большой точностью извлекает все имена и названия благодаря интеграции с *Taglib*.

Нам понравился Vitunes благодаря его разумному подходу.



## Web-браузер

## Min

Версия: 1.5.1 Сайт: <https://github.com/minbrowser>

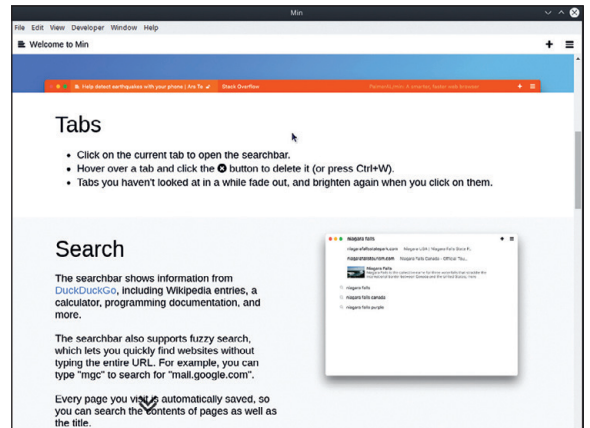
Есть много подходов к созданию альтернативных браузеров, и число проектов постоянно растет. Мы нагнулись на еще одну попытку добиться «наипростейшего» подхода к реализации браузера.

*Min* — это недавнее приложение, основанное на *Electron* и *NodeJS*, и, следовательно, использующее встроенный движок *Chromium* для рендеринга страниц. Мало кого удивит наличие еще одного браузера на базе *Chromium*, однако *Min* отличается внешне. Вкратце, *Min* относительно *Chromium* — то же, что *FocusWriter* относительно *LibreOffice Writer*. У него целеустремленный, минималистский и понятный интерфейс, в соответствии с основными принципами Материального дизайна [Material Design] от Google. Большая часть вида браузера по умолчанию пуста, но нажав на верхнюю строку, вы можете ввести адрес или искать что-то. Пока вы блуждаете по Интернету, *Min* окрашивает верхнюю

строку согласно дизайну текущей страницы — нечто подобное уже наблюдалось в проприетарном браузере *Vivaldi*.

В верхней строке также размещается список открытых вкладок: просто нажмите на удобное сочетание `Ctrl+T`, чтобы создать новую открытую вкладку. Помимо ожидаемого «крестика», который закрывает вкладку, есть также значок с выровненной стопкой на противоположной стороне заголовка вкладки. Нажав на него, вы попадете в режим «чтения», который убирает все лишние элементы дизайна на странице и форматирует ее для удобства чтения, чтобы сосредоточиться чисто на тексте.

*Min* предлагает и другие инструменты для работы с контентом: например, *Focus*



Советуем предпринять ознакомительный тур для изучения самых выдающихся функций этого браузера.

Mode [Режим Фокуса] (под меню View [Вид]), который скрывает почти всё, кроме текущей страницы, а также Tasks [Задачи] — способ группировать вкладки и организовать рабочий процесс. Когда вы дадите задаче имя, все страницы, которые вы посетите после, будут принадлежать этой задаче. *Min* позволяет перетаскивать вкладки между задачами — очень удобно!

Двоичные пакеты на сайте *Min* — только для macOS и Ubuntu, но у нас не было проблем с запуском браузера на менее популярных ОС. Нужны только *NodeJS* с менеджером пакетов NPM и пакет *Grunt*, и *Min* запустится как клиент в среде *Electron*.

Окрашивает верхнюю строку согласно дизайну текущей страницы.

## Медиа-конвертор

## Curlew

Версия: 0.2.4 Сайт: <http://bit.ly/2qK259K>

Мы понимаем, что воспроизведение видео 4K — это непосильная задача для маленького экрана карманного устройства. Конвертирование видео под определенную платформу — ход, как правило, хороший: вы экономите место на диске и ресурсы (и, следовательно, заряд батареи) на своем смартфоне или планшете.

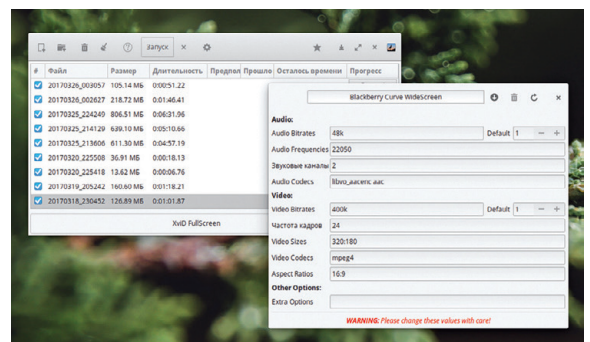
В Linux уже есть достойный выбор медиа-конверторов. Большинство из них основаны на широко используемом пакете *FFmpeg*. Его можно найти практически повсеместно, и если вы решите от него избавиться, отдельный экземпляр *FFmpeg* всё равно будет по-прежнему существовать в вашем браузере на базе *Chromium*.

*Curlew* — хорошее дополнение к многофункциональному рабочему столу на *GTK3*. В нем имеется достойный выбор команд *FFmpeg* в блестящем и приглядном интерфейсе со строкой заглавия и стильными значками. Приложение делает то же, что и *Handbrake*, еще один конвертер файлов,

так что выбор между ними — дело исключительно личного вкуса и визуальных предпочтений.

Из этих двух у *Curlew* более компактный интерфейс и он по умолчанию скрывает свои расширенные настройки. Вам надо сначала открыть файл или папку, или с помощью кнопок на строке заглавия, или просто перетаскив контент по окну *Curlew*.

Большая кнопка в нижней части окна показывает текущий целевой формат. Нажмите на нее, чтобы выбрать из внушительного списка поддерживаемых медиа-форматов. Здесь перечислены опции видео и аудио, и вы можете использовать *Curlew* для извлечения аудиотрека из видео. Нажав на кнопку настройки, вы можете точно настроить каждый целевой формат



Мы не знаем, умны ли серые киты, но они определенно быстрые и легкие!

и погрузиться в его параметры для установки опций битрейта, профилей кодеков, определенной частоты кадров и т.д. Программа умеет обрабатывать файлы в очереди, поэтому она идеальна для пакетного конвертирования. Естественно, предлагается не так много дополнений, помимо стандартных опций *FFmpeg*, но вряд ли большинству это нужно.

Менее 150 КБ, *Curlew* просто крошечный, и зависит от *Python3*, *GTK3* и, как и следовало ожидать, от *FFmpeg* и *Libavconv*. Проект любезно предоставляет Deb-пакет для Ubuntu и его производных.

Интерфейс Curlew имеет строку заглавия и стильные значки.

## HotGames Развлекательные приложения

Стрелялка с боковой прокруткой

# SpeedLazer

Версия: GIT Сайт: <http://bit.ly/2qOtl1Z>

Существует общее предубеждение против браузерных игр. Считается, что большинство из них просты и не могут в полной мере задействовать возможности вашего оборудования. Однако многие демо-версии WebGL онлайн опровергают этот аргумент, а стало быть, всё дело в том, чтобы найти достойную игру на OpenGL, которая запускается в браузере.

*SpeedLazer* — сторонняя стрелялка с многоцветной графикой, достойным звуком и быстрым движением. На вас накатывают волны вражеских кораблей — и вам надо просто их отстрелять. Уничтожив достаточно врагов, вы сможете улучшить свое оружие и функции самолета, ловя разные летающие подарки.

Интересная функция *SpeedLazer* в том, что сессия игры по умолчанию позволяет вам поиграть с другом, используя два

геймпада, или же клавиатуру и геймпад. У вас должен быть браузер на базе *Chromium* или браузер *Firefox*. Можете проверить, распознает ли он ваше оборудование, зайдя на <http://html5gamepad.com>.

В режиме одного игрока второй игрок управляется ИИ, и вскоре его убивают, оставляя вас бороться за себя в небе в одиночестве. Хотя вы получаете приказы от командира, вам всё равно надо сопротивляться бесконечным атакам вражеских дронов в небе, да еще и подводным угрозам, например, минам и морским дронам, когда вы летите над морем. Уровень сложности игры довольно высок, поэтому вам



► Не отвлекайтесь на чудесный закат — там злобные дроны, которых надо уничтожить!

придется позаботиться о своих летных навыках, чтобы продержаться дольше, чем несколько минут.

*SpeedLazer* работает на *Crafty.js* и доступен онлайн на <http://speedlazer.net>. В качестве альтернативы можете запустить его на локальной машине с помощью *npm*, *bower* и *Grunt* — подробные инструкции представлены в официальном документе README.md.

**Вам надо сопротивляться атакам вражеских дронов в небе.**

## Гонки

# Tux Racer

Версия: 0.7.4 Сайт: <http://bit.ly/2qOGms6>

Согласно исследованиям, есть селедку намного полезнее, когда вы скатываетесь на пузе с горы со скоростью 200 км/ч. Примерно так вкратце описывается игра *Extreme Tux Racer*, целью которой является набрать очки за время и количество собранных рыб.

*Extreme Tux Racer* — одна из звезд среди игр с открытым кодом. Она существует примерно с 2000 г. и стала классической презентацией Тукса [Tux], талисмана Linux. Просто прибавили приставку "Extreme".

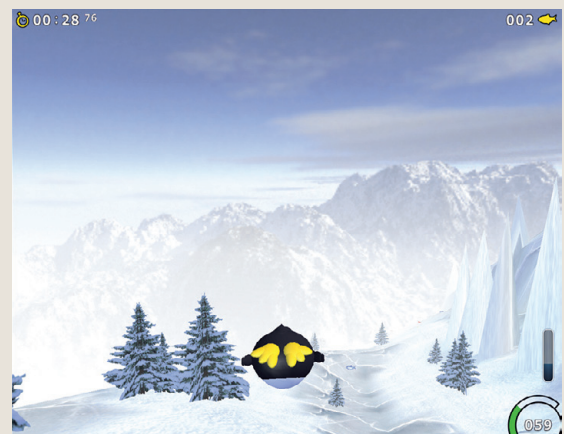
Качественная графика показывает отражения от ледяных поверхностей и отлично прорисованные елки, что, в сочетании с веселой музыкой, делает игру увлекательной и забавной. Последние версии *Extreme Tux Racer* предлагают больше опций, начиная с персонажа: вместо Тукса вы можете играть за пингвина-девушку Трикси [Trixie], демоненка Бисти

[Beastie] от FreeBSD или тюленя Самуэля [Samuel].

Игра понравится и детям, и взрослым, и содержит также и тактический элемент. Вы можете разогнаться до 60 км/ч, после чего надо постараться удержать своего персонажа на ледяных частях карты; и изо всех сил помогайте Туксу разогнаться еще больше. Вы можете заставить его кувыркаться в полете, удерживая Т и любую клавишу со стрелками. Выиграть это вам не поможет, но выглядит великолепно!

Можно менять время суток или добавлять элементы, и попытаться закончить свой уровень, скажем, при сильном боковом ветре, соревнуясь ночью.

**Изо всех сил помогайте Туксу разогнаться еще больше.**



► Тукс не особо следит за своей формой, но при этом летает, как реактивный самолет!

*Extreme Tux Racer* позволяет пройти подготовительные уровни, а в режиме кампании вы можете попытаться выиграть все кубки в выбранной кампании. В каждом кубке вы обязаны пройти маршрут за установленное время, собрав достаточно рыб, что совсем не так просто, как может показаться!

Лучше всего начать с тренировочной горки, а уж затем переходить ко множеству уровней, предоставленных сообществом Linux.

## Приложение для обмена сообщениями

## Konv

Версия: 0.1 Сайт: <http://bit.ly/2rt0NNj>

Каждые несколько лет люди меняют предпочтения в способе обмена быстрыми сообщениями, обычно из-за брешей и уязвимостей в системе безопасности, о которых становится известно через СМИ. Конфиденциальность — это, вероятно, самая большая забота людей вообще, а для пользователей Linux она играет еще более важную роль.

Сеть Tox появилась как следствие сообщений Эдварда Сноудена об утечках и о шпионской деятельности АНБ несколько лет назад. Даже если вы чужды сплетням и заговорам, в использовании более безопасного канала коммуникации в повседневной жизни нет ничего дурного. Шаг вперед, *Konv*! В основном он делает практически то же самое, что и *qTox*, который мы рассматривали в **LXF211** и **217**, но использует интерфейс *GTK3* вместо *Qt5*.

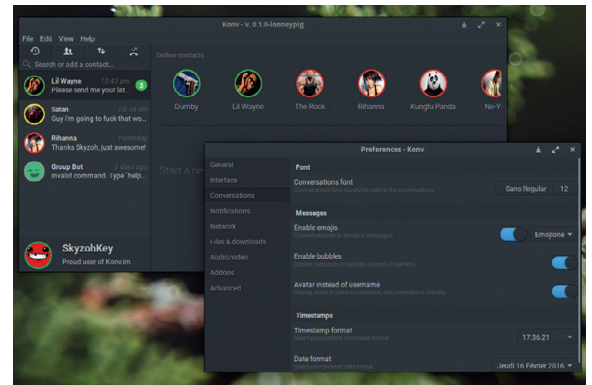
Итак, *Konv* появился, чтобы заменить *Skype*, *WhatsApp* и *Telegram* более прозрачным подходом, не требующим проприетарного web-сервера, чтобы вы могли

связаться с друзьями. Безопасные разговоры в небезопасном открытом Интернете больше не проблема, если вы выберете *Konv*, потому что у вас будет преимущество однорангового соединения.

Интерфейс *Konv* аккуратный и удобный, хотя предусмотрен только на английском и французском языках. Каждый раз, когда вы вызовете *Konv*, не загружая своего профиля, приложение в ответ поприветствует вас интересным профилем, заполненным знаменитостями и вымышленными персонажами.

*Konv* отлично подходит для повседневного использования, аудио- и видеозвонков, передачи файлов и т.д. Он интегрируется с такими рабочими столами, как *GNOME*, *Unity* и *Pantheon*, и выглядит в них,

## Поприветствует вас профилем, заполненным знаменитостями.



► Похоже, пора поболтать с Кунг-Фу Пандой...

как родное приложение. Диалоговое окно Preferences [Предпочтения] предлагает ряд настроек безопасности, например, использование списка индивидуальных узлов загрузки, что может помешать отслеживанию.

Вам понадобится некоторое время на привыкание к обмену сообщениями в стиле Tox, где нет центрального сервера аутентификации (например, вам надо прихватить свой профиль с собой, чтобы выйти онлайн на другом ПК), но мы полагаем, что усилия будут вознаграждены.

## Читалка новостей

## Feedreader

Версия: 2.0.2 Сайт: <http://bit.ly/2rk0gyK>

Чтение новостей в течение длительного времени — это деятельность, которая в определенных случаях может привести к жесткой прокрастинации. Нет ничего плохого в осведомленности о том, что происходит вокруг, однако мы должны выделить важность того, как именно люди получают новости.

Большинство онлайн-медиа поддерживают формат RSS, что позволяет подписаться на ленту сайта и читать новостные заголовки и краткие новости, не используя браузера. *Feedreader* — прекрасный клиент RSS, который выводит чтение на новый уровень комфорта. Приложение основано на *GTK3* и имеет приятный и лаконичный внешний вид.

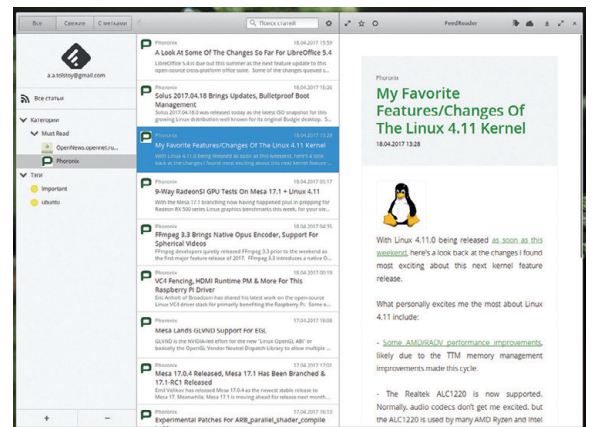
После первого запуска вам надо будет предоставить информацию об учетной записи для того сервиса агрегации RSS, по которому вы страдали. Да, *Feedreader* интегрируется с таким сервисом, вместо того, чтобы импортировать RSS напрямую,

так что вам действительно надо зарегистрироваться. Приложение поддерживает Tiny Tiny RSS, Feedly, Owncloud и InoReader. Мы выбрали Feedly, который, в свою очередь, позволяет вам войти, используя аутентификационную информацию Google.

Окно *Feedreader* сразу же заполнится подписками, которые ранее были зарегистрированы с помощью Gmail. Структура приложения похожа на то, что мы многократно видели в почтовых клиентах: список лент слева, текущая лента новостей в середине и выбранный новостной контент справа.

Для *Feedreader* нужно некоторое время, чтобы найти ваши подписки, но во всем остальном он предлагает приятное и сосредоточенное чтение. Когда его используют

## Feedreader предлагает приятное и сосредоточенное чтение.



► Сортируйте, присваивайте теги и изучайте новости в этом быстром приложении.

с рабочими столами, такими как *Unity* или *Pantheon*, *Feedreader* отображает ряд свежих непрочитанных новостей рядом со своим значком в доке или панели приложений.

Внутри программы вы можете отмечать новости как прочитанные или непрочитанные, добавлять теги, осуществлять поиск по лентам и новостям, делиться новостями и т.д. Она напомнила нам работу электронной почты, но по сравнению с чтением новостей в браузере использование лент RSS во *Feedreader* во многом было удобнее. Никакой рекламы, никаких посторонних материалов — вы получаете именно то, на что подписались! **LXF**

# На диске

Дистрибутивы, приложения, игры, книги и всякое-разное...

Лучшее из Интернета, упакованное в 8 ГБ качественного DVD.



## Дистрибутивы

Когда-то говорили: «Linux дружелюбен к пользователю, просто он придирчиво выбирает друзей». Ныне это уже не актуально, потому что Linux стал намного проще для новичков.

На DVD этого месяца — три отличных примера того, как настроить дистрибутив, чтобы в нем легко освоились начинающие пользователи, но не проиграли и опытные пользователи тоже. Особенность статуса новичка (в любом деле) такова, что он преходящий: со временем вы или наберетесь опыта и освоитесь, или просто бросите.

Наши три дистрибутива предназначены в первую очередь тем, кто перешел с Windows, но это всё же дистрибутивы Linux. Linux — не просто бесплатная операционка для тех, кому жалко денег на Windows. Это отдельная ОС, и как таковая, она делает всё иначе. Если вы впервые используете Linux, следует помнить именно об этом. Неважно, насколько рабочий стол похож на Windows — Linux другой... так что будьте готовы научиться делать всё по-другому.

Классическим примером этого является установка программ, которая использует центральный менеджер пакетов — систему, принятую в мобильных устройствах — вместо скачивания программ с разных сайтов.

*Neil*

## » Важно ВНИМАНИЕ!

### Порченые 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.

## Дружелюбный к Windows дистрибутив Linux

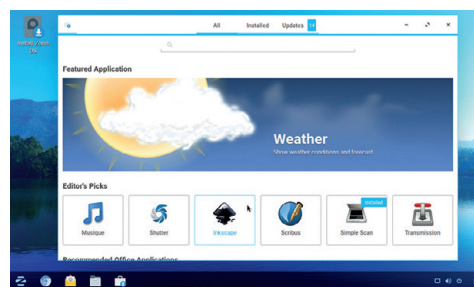
# Zorin OS 12.1

Хотя Zorin OS уже достиг версии 12.1, *Linux Format* уже несколько лет не включал его на свои диски. Так давайте наверстаем упущенное с последней версией этого дистрибутива для начинающих линуксоидов. У него простой интерфейс, который должен быть достаточно знаком пользователям Windows, впервые пытающимся окунуться в воды Linux, и при этом Zorin OS содержит всё, что делает Linux особенным.

Легкость в работе не обязательно должна означать простоватость или ограничения: в глубине души это всё же дистрибутив Linux, подобный всем остальным. Из него ничего не убрали; просто интерфейс пользователя и выбор программ по умолчанию настроены для тех, кому привычнее Windows. Zorin практикует такой же подход, как elementary OS, предлагая для выполнения основных задач, для которых и служат компьютеры, заранее отобранные программы; разве что его интерфейс более традиционный.

Дистрибутив взял за основу Ubuntu, так что у вас есть доступ ко всему ассортименту программ, предлагаемых Ubuntu, а также и Debian — просто через выбор опции Software в главном меню. Это стандартная настольная версия Zorin OS, но есть также версии, предназначенные специально для бизнеса или образования. Однако все программы доступны в репозиториях, и вы можете установить и запустить любую из версий.

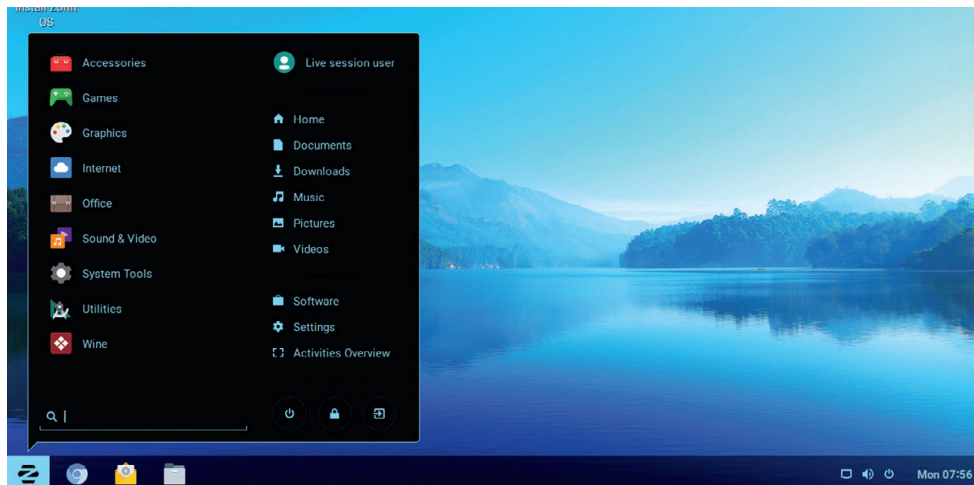
Если у кого-то и были сомнения в том, что Zorin OS предназначен для перешедших с Windows, они должны рассеяться благодаря присутствию раздела *Wine* в главном меню. *Wine* позволяет запускать программы



» Начальный выбор программ в Zorin OS тщательно продуман, но их доступно намного больше, и они выполняют практически любую задачу.

Windows в Linux, однако принудить эти программы к совместной работе бывает непросто. Чтобы справиться с этим, Zorin включает не только сам *Wine*, но также и *PlayOnLinux*. Это GUI для *Wine*, который значительно облегчает процесс запуска программ Windows в Linux, особенно с базовым *Wine*. Очень хорошо, когда есть *Wine* и *PlayOnLinux*, но они нечасто вам понадобятся, поскольку Zorin OS предлагает большой выбор программ, способных предоставить всё, что вы делали в Windows, а использовать родные программы всегда лучше!

DVD должен загружаться прямо в рабочий стол Zorin OS, и вам не понадобятся имя пользователя или пароль; но на всякий случай, пользователь — «zorin» с пустым паролем (если вам предложат ввести пароль, просто нажмите Enter).



**ЛУЧШИЙ ДИСТРИБУТИВ  
ДЛЯ НОВИЧКА**

**Elementary OS 0.4.1**

Простой и понятный дистрибутив, который облегчит вам переход с MacOS или Windows

Быстрый Открытый Многофункциональный

**Manjaro 17.0** **Zorin OS 12.1**

Выглядит профессионально, работает дружелюбно

Индивидуально настраиваемый, крутой и очень стильный

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

# Новичок в Linux? Начните отсюда!

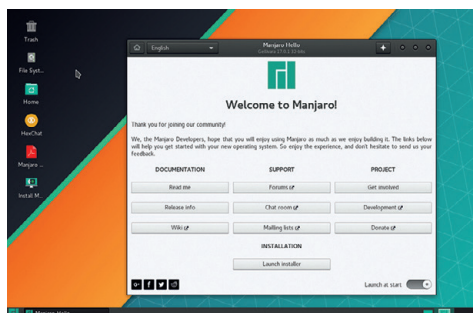
- » Что такое Linux? Как его установить?
- » Есть ли в нем эквивалент MS Office?
- » Зачем нужна командная строка?
- » Как устанавливать программы?

Ответы приводятся в [Index.html](#) на диске.

Мощный потомок Arch Linux

## Manjaro 17.0.1

Другие два дистрибутива на первой стороне нашего DVD основаны на Ubuntu, а вот Manjaro произошел от Arch Linux. Arch — это в какой-то степени дистрибутив для опытных пользователей, и вас может удивить, что Manjaro считают подходящим для новичков — и тем не менее это так! Manjaro предлагает выбор рабочих столов, и в этом месяце вам предлагается версия Xfce. Xfce — это быстрый, достаточно легковесный рабочий стол, который следует более традиционной структуре рабочего стола. И это делает его идеальным выбором для начинающих пользователей, если только их единственным опытом не был Windows 8! Логин: manjaro, пароль: manjaro.



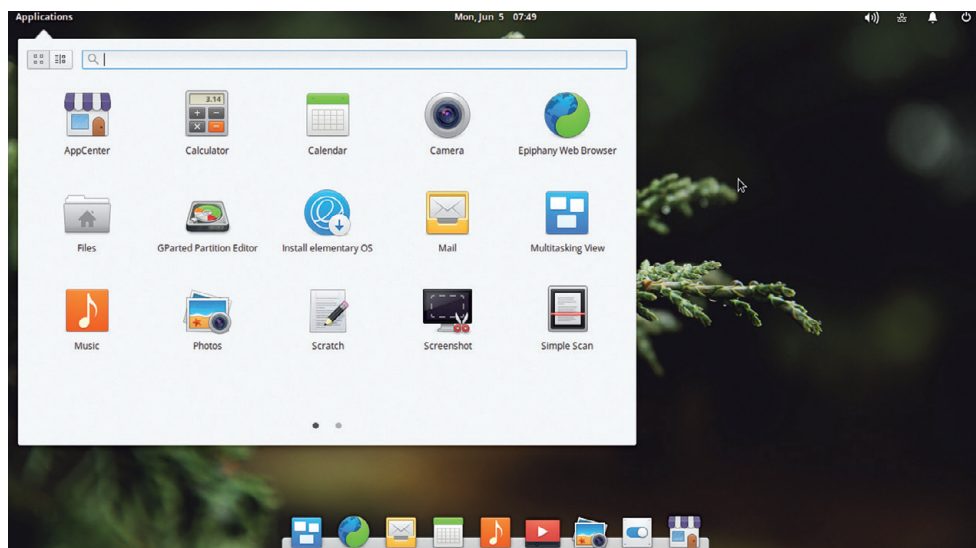
» Если вы раньше работали с Windows, в Manjaro вы почувствуете себя как дома.

Простой дистрибутив на базе Ubuntu

## Elementary OS 0.4.1

С таким названием elementary OS вряд ли учредит приз за угадывание его цели: быть простым в работе дистрибутивом, годным для новичков в Linux. Но он отнюдь не упрощен до абсурда — это скорее дистрибутив Linux, сделавший приоритетом простоту. Часто говорят, что в Linux главное — выбор; многие дистрибутивы восприняли это на-ура и предлагают кучу программ для выполнения одной задачи, обескураживая

начинающих. В elementary OS выбор более узкий — по программе на каждую задачу: один офисный пакет, один графический редактор, один клиент электронной почты, и т.д. Это облегчает жизнь новичкам, но может стать ограничением по мере вашего освоения Linux. К счастью, elementary OS основан на Ubuntu, и вы получаете доступ к изобилию программ в репозиториях Ubuntu и Debian. Логин: elementary, пароль: пустой.



## 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.

**Memtest86+** 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.

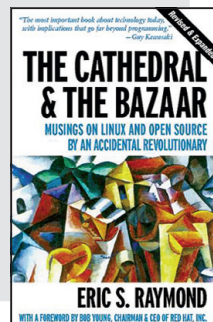
**Введение в Linux** Удобное руководство со множеством подсказок для начинающих пользователей Linux.

**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 Грегом 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!

## LXF225 Июль 2017



- » Притворимся хакерами Но не корысти ради
- » Мультки и не только Программы анимации
- » Параллельный Интернет Сеть для избранных
- » Лучше перебдеть Двухфакторная аутентификация

**LXFDVD:** Ubuntu, Solus, Android-x86, 12 книг о Linux (на английском), горячие новинки, архив LXF за 2005–2016 гг. и прочее...

Печатная версия: [shop.linuxformat.ru/lxf\\_225/](http://shop.linuxformat.ru/lxf_225/)  
PDF-версия: [shop.linuxformat.ru/elxf\\_225/](http://shop.linuxformat.ru/elxf_225/)

## 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

**LXFDVD:** 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

**LXFDVD:** 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

**LXFDVD:** 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

**LXFDVD:** 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

**LXFDVD:** 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/)

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

**Elementary OS 0.4.1** (64-битный)  
Настольный дистрибутив на базе Ubuntu, сделавший своим приоритетом простоту. Предлагает множество готовых пользовательских приложений, включая Photos, Music, Videos, Calendar, Terminal, Files и т. д., состав которых можно пополнить из репозитория Ubuntu и Debian.

**Manjaro 17.0** (32-битный)  
Быстрая и удобная настольная ОС на базе Arch Linux. Ключевые свойства — интуитивно понятный процесс установки, автоматическое обновление оборудования, стабильная возобновляемая версия, возможность установки нескольких ядер, специальные скрипты *Basit* для управления графическими драйверами и исключительная настраиваемость.

**Zorin OS 12.1** (64-битный)  
Дистрибутив на базе Ubuntu предоставляет знакомый интерфейс с целью облегчить пользователям переход с Windows на Linux.

**УЧЕБНИКИ**  
Зарадка LXF  
Рисовки

#### НОПРИСКС

**AMGryseach 1.0.0** Поликовая машина  
<https://bit.ly/2ZtRedz>  
**Slime 0.2.4** Медиа-конвертор  
<https://bit.ly/2Kc259K>  
**Extreme Tux Racer 0.14** Гонки  
<https://bit.ly/2R0Dn8s>  
**Feedreader 2.0.2** Читалка новостей  
<https://bit.ly/2K0KqK>  
**Formalade 1.12.2** Видеоредактор  
<https://bit.ly/2DmZU13>

**KMail 17.04** Клиент электронной почты  
<https://bit.ly/2stKJCE>  
**Копи 0.1** Приложение для обмена сообщениями  
<https://bit.ly/2Z10U11>  
**Мип 1.5.1** Web-браузер  
<https://bit.ly/2cm1p0a>  
**Screenoid 1.31** Инструмент создания экранных снимков  
<https://bit.ly/2wM1dE>

**Speed-azer (G1T)** Стрелялка с боковой прокруткой  
<https://bit.ly/2D011Z>  
**Vinues 2.3** Музыкальный плеер  
<https://bit.ly/2X1v50d>

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)

# ЛУЧШИЙ ДИСТРИБУТИВ ДЛЯ НОВИЧКА

## Elementary OS 0.4.1

Простой и понятный дистрибутив,  
который облегчит вам переход  
с MacOS или Windows

» Быстрый » Открытый » Многофункциональный



Manjaro 17.0

Выглядит профессионально,  
работает дружелюбно



Zorin OS 12.1

Индивидуально настраиваемый,  
крутой и очень стильный



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

Внимательно прочтите это перед тем, как использовать LXF DVD!

# Содержание



## Сторона 2

### ДИСТРИБУТИВЫ

#### BackBox Linux 5 (64-битный)

Простой и быстрый дистрибутив на базе Ubuntu, предназначенный для тестирования на втроежения и оценку безопасности. Предлагает минималистичную, но исчерпывающую настольную среду и обширную коллекцию утилит безопасности.

#### Tails 3.1 (64-битный)

Live-дистрибутив на CD/USB, на базе Debian, предназначенный для обеспечения полной анонимности пользователя в Интернете. Поставляется с несколькими интернет-приложениями, включая веб-браузер, почтовый и IRC-клиент и клиент обмена мгновенными сообщениями, преднастроенные с учетом безопасности. Использует сеть Tor, чтобы затруднить слежение за интернет-трафиком.

### АРХИВ НОМЕРОВ ЖУРНАЛА LINUX FORMAT ЗА 2016 Г.

### ИНДЕКС СТАТЕЙ В LINUX FORMAT ЗА 2005–2016 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 установка на жесткий диск.

### Пожалуйста, перестаньте использовать именем Адного Диска озанкомьтесь 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 этапах производства, однако, как и в случае с любым другим ПО, мы рекомендуем вам использовать аппаратный сканер. Мы также рекомендуем всегда иметь под рукой актуальную резервную копию данных вашего жесткого диска. К сожалению, редакция Linux Format не в состоянии принимать на себя ответственность за любые повреждения, разрушения или иные убытки, которые могут повлечь за собой использование этого DVD, представленных на нем программ или данных. Прежде чем устанавливать какое-либо ПО на компьютер, пожалуйста, убедитесь, что вы получили разрешение от автора ПО. Поддержка Linux Format осуществляется с помощью добровольцев. Этот диск изготовлен ООО «Марком», 188652, Россия, Ленинградская область, Всеволожский р-н, дер. Юрки, Школьная ул., 7-а. Лицензия ИМТР ВАО № 77-03.

## Создание установочных дисков при помощи cdrecord

Самый быстрый способ записать ISO-образ на чистую матрицу — это обратиться к программе *cdrecord*. Для всех перечисленных ниже действий потребуются права суперпользователя-root. Сначала определите путь к вашему устройству для записи дисков. Наберите следующую команду:

```
cdrecord -scanbus
```

После этого на экране терминала должен отобразиться список устройств, подключенных к вашей системе. SCSI-адрес каждого устройства представляет собой три числа в левой колонке — например, 0,3,0. Теперь вы можете с легкостью записать образ на диск:

```
cdrecord dev=0,3,0 -v /путь к образу/image.iso
```

Чтобы упростить дальнейшее использование *cdrecord*, сохраните некоторые настройки в файле `/etc/default/cdrecord`. Добавьте по одной строке для каждого устройства записи (скорее всего, в вашей системе присутствует только одно такое устройство):

```
Plextor= 0,3,0 12 16M
```

Первое слово в этой строке — метка; затем после адреса SCSI-устройства вы должны указать скорость и размер буфера. Теперь можете заменить SCSI-адрес в командной строке на выбранную вами метку. Все будет еще проще, если вы добавите следующее:

```
CDR_DEVICE=Plextor
```

Для записи ISO-образа вам осталось набрать команду

```
cdrecord -v /path/to/image.iso
```

Если вы не принадлежите к любителям командной строки, в таком случае вам придет на помощь утилита *gcombust*. Запустите ее от имени root и выберите вкладку Burn и ISO 9660 Image в верхней части окна. Введите путь к образу, который вы хотите записать на диск, и смело нажимайте на Combust! Пока ваш образ пишется на диск, можете выпить чашечку кофе.

## Другая ОС?

Использовать Linux для записи компакт-диска не обязательно. Программы вроде *cdrecord* просто переносят двоичные данные на чистую матрицу. Все необходимые файлы уже включены в ISO-образ, который распознается любой операционной системой, будь то Linux, Windows, Mac OS X или AmigaOS.

## Нет устройства для записи дисков?

Если у вас нет устройства, с помощью которого можно было бы записать образ на диск, можно найти какого-нибудь друга или организацию, у кого есть компьютер с дисководом, и прожечь диск у них. Опять-таки, вам подойдет любая операционная система, способная распознать пишущий привод (см. выше).

Некоторые дистрибутивы умеют монтировать образы дисков и выполнять сетевую установку или даже установку с раздела жесткого диска. Конкретные методы, конечно, зависят от дистрибутива. За дополнительной информацией обращайтесь на web-сайт разработчика дистрибутива.



# 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





В сентябрьском номере

# Как это — виртуализовать?

Распатроним гипервизор, развернем контейнеры Docker и всё превратим в виртуальность: этак проще будет жить.

## Файловая система Zeta

Пристально рассмотрим, в чем ваши нужды и как непринужденно выкатить ZFS на все ваши компьютеры.

## Linux Multimedia Studio

Вместе сочиняем прекрасную музыку на профессиональном уровне при помощи комплекса LMMS для композиторов.

## BSD против Linux

Лучшие ядра ОС с открытым кодом идут голова к голове; решим, наконец, которое больше подойдет вашему серверу.

## Серверы, серверы, серверы

Какой сервер лучше развернуть для бизнеса? Стабильность, скорость, функции и прочее — протестируем буквально всё.

Содержание будущих выпусков может меняться — вдруг мы ударимся в увеселения...

# LINUX FORMAT

Главное в мире Linux

Журнал зарегистрирован Федеральной службой по надзору за соблюдением законодательства в сфере массовых коммуникаций и охране культурного наследия ПИ № Ф077-21973 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, Даниил Кривошеин, Светлана Кривошеина, Валентин Развозжаев, Валерий Смирнов, Елена Толстякова, Ирина Шулакова

#### 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 «Линукс 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 16313

### 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 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-Теге Аксельссон [Mats Tage Axelsson], Джонни Бидвелл [Jonni Bidwell], Нейл Ботвик [Neil Botwick], Марк Чизхолм [Mark Chisholm], Нейт Дрейк [Nate Drake], Эрмин Крепоник [Ermin Kreponik], Джереми Лэйд [Jeremy Laird], Ник Пирс [Nick Peers], Лес Паундер [Les Pounder], Шашанк Шарма [Shashank Sharma], Зак Стори [Zak Storey], Александр Толстой [Alexander Tolstoy], Михалис Цукалос [Mihalis Tsoukalos], Филиппа Уорр [Philippa Warr], Майк Вильямс [Mike Williams], Евгений Балдин, Андрей Гондаренков, Сергей Ерёменко, Дмитрий Кузнецов, Алексей Федорчук, Максим Черепанов

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 ответственность за повреждение или ущерб, нанесенные вашему компьютеру и периферии вследствие использования тех или иных советов.

Linux — зарегистрированный товарный знак Линуса Торвальдса [Linus Torvalds].

“GNU/Linux” заменяется на “Linux” в целях сокращения. Все остальные товарные знаки являются собственностью их законных владельцев. Весь код, опубликованный в журнале, лицензирован на условиях GPL v3. См. [www.gnu.org/copyleft/gpl.html](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).



© 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

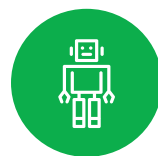
- Праздничные мастер-классы
- Городской лагерь на время каникул
- Курсы повышения квалификации для педагогов



Инновационная программа обучения

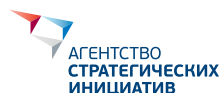


Собственная методическая база



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**

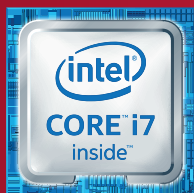
**ОСТРЫЙ КАК ЛЕЗВИЕ**

**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.